Paralinear Solutions

Team 7

Deliverable 04: Technical Specification

Our Team:



Vincent Yu	Team Leader Michael Vosloo	Tshepang Mashao	Amore Rossouw	Matthew Veltman
u15195059	u19031174	u19293675	u19052864	u19050608
084 073 7834	082 387 0072	074 447 0511	073 567 3529	081 341 1664
u15195059@tuks.co.za	u19031174@tuks.co.za	u19293675@tuks.co.za	u19052864@tuks.co.za	u19050608@tuks.co.za

Client Information:





System Name: Gohvan Construction system
Type of System: Complete Business Solution

This deliverable represents the technical flow and representation of the system that Team 7 is building for Gohvan Construction. The documents included in this deliverable will contain a final descriptive layout of each functionality that will be presented in the system. Each functionality aspect in this deliverable will have the relevant diagrams for further explanation and a more visual representation for each data flow in the system. The diagrams present will include use cases for the entire system, context diagrams for the system, High, Middle and primitive level data flow diagrams. Finally, this document will also include basic illustrations of the solution for the system as well as test scenarios of the system we have designed for Gohvan construction.

Contents

Deliverable Introduction 15
System Architecture Design 16
1. Use Cases 16
Introduction 16
1.1 Use case diagrams 17
1.1.1 Establishment Subsystem 17
1.1.2 Quotation Subsystem 18
1.1.3 Administration Subsystem 19
1.1.4 Construction Subsystem 20
1.1.5 Invoicing Subsystem 23
1.1.6 Human Resource Subsystem 24
1.1.7 Teams Subsystem 25
1.1.8 Supplier Order Subsystem 26
1.1.9 Supplier Subsystem 27
1.1.10 Inventory Management Subsystem 28
1.1.11 Report Subsystem 29
1.1.12 User Management Subsystem 30
1.1.13 Client Subsystem 31
1.2 Use case Narratives 32
1.2.1. Establishment Subsystem 32
1.2.1.1 Create TSS 32
1.2.1.2 Edit TSS 60
1.2.1.3 Submit TSS Report 87
1.2.2. Quotation Subsystem 90
1.2.2.1 Create Quote 90
1.2.2.2 Search Quote 110
1.2.2.3 Capture Purchase Order 114
1.2.3. Administration Subsystem 121
1.2.3.1 Create Health and Safety file 121
1.2.3.2 Search Health and Safety file 124
1.2.3.3 Create Health and Safety Type 127
1.2.3.4 Search Health and Safety Type 129
1.2.3.5 Create Drawing Type 131
1.2.3.6 Search Drawing Type 133

1.2.3.7 Upload Formal Drawing 135
1.2.3.8 Upload Second cut Drawing 138
1.2.3.9 Search As-built document 141
1.2.3.10 Generate As-Built Document 144
1.2.4. Construction Subsystem 145
1.2.4.1 Capture Site-Go-Ahead 145
1.2.4.2 Search construction 156
1.2.4.3 Edit construction 164
1.2.4.4 Allocate Team Site 174
1.2.4.5 Search Allocated Team 182
1.2.4.6 Update Allocated Team 188
1.2.4.7 Upload Compliance Document 195
1.2.4.8 Search Compliance Document 202
1.2.4.9 Search Project 208
1.2.4.10 Create Site Progress description 214
1.2.4.11 View Site Progress description 219
1.2.4.12 Upload complete site photos 223
1.2.4.13 Search complete site photos 229
1.2.4.14 Upload Radio Frequency report 232
1.2.4.15 Search Radio Frequency report 239
1.2.4.16 Request Site Visit 246
1.2.4.17 Create Snag 250
1.2.4.18 Search Snag 255
1.2.4.19 Edit Snag 260
1.2.4.20 Edit Acceptance Document 264
1.2.4.21 Search Acceptance document 272
1.2.4.22 Confirm acceptance document 278
1.2.4.23 Capture Site Sign-off 284
1.2.4.24 Search Site Sign-off 290
1.2.4.25 Capture Commission report 295
1.2.4.26 Search Commission report 302
1.2.4.27 Create Site Type 308
1.2.4.28 Search Site type 313
1.2.4.29 Edit Site Type 318
1.2.5. Invoicing Subsystem 322

1.2.5.1 Search Invoice 322
1.2.5.2 Send Invoice 326
1.2.5.3 Capture Proof of Payment 331
1.2.5.4 Capture Claim 338
1.2.5.5 Search Proof of Payment 344
1.2.5.6 Search Invoice Type 349
1.2.5.7 Create Invoice Type 352
1.2.6. Human Resource Subsystem 358
1.2.6.1 Add employee 358
1.2.6.2 Edit employee 366
1.2.6.3 Search employee 374
1.2.6.4 Upload Employee Document 382
1.2.6.5 Edit Employee Document 390
1.2.6.6 Search Employee document 397
1.2.6.7 Create Account 402
1.2.6.8 Edit Account 409
1.2.6.9 Search Account 415
1.2.6.10 Create Employee Type 420
1.2.6.11 Search Employee Type 427
1.2.6.12 Create Employee Skill 432
1.2.6.13 Search Employee Skill 439
1.2.7. Team Subsystem 445
1.2.7.1 Create Team 445
1.2.7.2 Search Team 452
1.2.7.3 Edit Team 457
1.2.8. Supplier Order Subsystem 465
1.2.8.1 Search Project Line-Item List 465
1.2.8.2 Edit Project Line-Item List 470
1.2.8.3 Create Supplier Purchase Order 478
1.2.8.4 Capture Supplier Quote 484
1.2.8.5 Search Supplier Quote 490
1.2.8.6 Search Supplier Purchase Order 495
1.2.9. Supplier Subsystem 500
1.2.9.1 Add Supplier 500
1.2.9.2 Search Supplier 508

1.2.9.3 Edit Supplier 514
1.2.10. Inventory Management Subsystem 523
1.2.10.1 Add-Line Item 523
1.2.10.2 Search-Line Item 529
1.2.10.3 Edit line-item 534
1.2.11. Reporting Subsystem 541
1.2.11.1 Generate Site Progress Report 541
1.2.11.2 Generate Site Type Report 548
1.2.11.3 Generate Team Site Completed Report 553
1.2.11.4 Generate Employee Document Expiry Report 560
1.2.11.5 Generate Site Acceptance Document 565
1.2.12. User Management Subsystem 571
1.2.12.1 Login 571
1.2.12.2 Logout 573
1.2.12.3 Reset Password 575
1.2.13. Client Subsystem 579
1.2.13.1 Add Client 579
1.2.13.2 Search Client 585
1.2.13.3 Edit Client 591
Conclusion 598
2. Process - Oriented Design 599
Introduction 599
2.1 Context Diagram 599
2.1.1 Establishment Subsystem 599
2.1.2 Quotation Subsystem 600
2.1.3 Administration Subsystem 601
2.1.4 Construction Subsystem 602
2.1.5 Invoicing Subsystem 607
2.1.6 Human Resource Subsystem 608
2.1.7 Teams Subsystem 610
2.1.8 Supplier Order Subsystem 611
2.1.9 Supplier Subsystem 612
2.1.10 Inventory Management Subsystem 613
2.1.11 Report Subsystem 614
2.1.12 User Management Subsystem 615

2.1.13 Client Subsystem 616
2.2 Decomposition 617
2.2.1 Establishment Subsystem 617
2.2.2 Quotation Subsystem 617
2.2.3 Administration Subsystem 618
2.2.4 Construction Subsystem 619
2.2.5 Invoicing Subsystem 622
2.2.6 Human Resource Subsystem 623
2.2.7 Teams Subsystem 624
2.2.8 Supplier Order Subsystem 624
2.2.9 Supplier Subsystem 625
2.2.10 Inventory Management Subsystem 625
2.2.11 Report Subsystem 626
2.2.12 User Management Subsystem 626
2.2.13 Client Subsystem 626
2.3 Physical Data Flow Diagrams 627
2.3.1 High Level Date flow diagrams 627
2.3.1.1 Establishment Subsystem 627
2.3.1.2 Quotation Subsystem 628
2.3.1.3 Administration Subsystem 629
2.3.1.4 Construction Subsystem 630
2.3.1.5 Invoicing Subsystem 635
2.3.1.6 Human Resource Subsystem 636
2.3.1.7 Teams Subsystem 638
2.3.1.8 Supplier Order Subsystem 639
2.3.1.9 Supplier Subsystem 640
2.3.1.10 Inventory Management Subsystem 641
2.3.1.11 Report Subsystem 642
2.3.1.12 User Management Subsystem 643
2.3.1.13 Client Subsystem 644
2.3.2 Mid-Level Data flow diagrams 645
2.3.2.1. Establishment Subsystem 645
2.3.2.1.1 Create TSS 645
2.3.2.1.2 Edit TSS 645
2.3.2.1.3 Submit TSS Report 646

2.3.2.2. Quotation Subsystem 647
2.3.2.2.1 Create Quote 647
2.3.2.2.2 Search Quote 647
2.3.2.2.3 Capture Purchase Order 648
2.3.2.3. Administration Subsystem 649
2.3.2.3.1 Create Health and Safety file 649
2.3.2.3.2 Search Health and Safety file 649
2.3.2.3.3 Create Health and Safety Type 650
2.3.2.3.4 Search Health and Safety Type 650
2.3.2.3.5 Create Drawing Type 650
2.3.2.3.6 Search Drawing Type 650
2.3.2.3.7 Upload Formal Drawing 651
2.3.2.3.8 Upload Second cut Drawing 651
2.3.2.3.9 Search As-built document 651
2.3.2.3.10 Generate As-Built Document 652
2.3.2.4. Construction Subsystem 653
2.3.2.4.1 Capture Site-Go-Ahead 653
2.3.2.4.2 Search construction 654
2.3.2.4.3 Edit construction 654
2.3.2.4.4 Allocate Team Site 655
2.3.2.4.5 Search Allocated Team 655
2.3.2.4.6 Update Allocated Team 656
2.3.2.4.7 Upload Compliance Document 656
2.3.2.4.8 Search Compliance Document 657
2.3.2.4.9 Search Project 657
2.3.2.4.10 Create Site Progress description 658
2.3.2.4.11 View Site Progress description 658
2.3.2.4.12 Upload complete site photos 659
2.3.2.4.13 Search complete site photos 659
2.3.2.4.14 Upload Radio Frequency report 660
2.3.2.4.15 Search Radio Frequency report 660
2.3.2.4.16 Request Site Visit 661
2.3.2.4.18 Search Snag 662
2.3.2.4.19 Edit Snag 662
2.3.2.4.20 Edit Acceptance Document 663

2.3.2.4.21 Search Acceptance document 663
2.3.2.4.22 Confirm acceptance document 664
2.3.2.4.23 Capture Site Sign-off 664
2.3.2.4.24 Search Site Sign-off 665
2.3.2.4.25 Capture Commission report 665
2.3.2.4.26 Search Commission report 666
2.3.2.4.27 Create Site Type 666
2.3.2.4.28 Search Site type 667
2.3.2.4.29 Edit Site Type 667
2.3.2.5. Invoicing Subsystem 668
2.3.2.5.1 Search Invoice 668
2.3.2.5.2 Send Invoice 668
2.3.2.5.3 Capture Proof of Payment 668
2.3.2.5.4 Capture Claim 669
2.3.2.5.5 Search Proof of Payment 669
2.3.2.5.6 Search Invoice Type 669
2.3.2.5.7 Create Invoice Type 669
2.3.2.6. Human Resource Subsystem 670
2.3.2.6.1 Add employee 670
2.3.2.6.2 Edit employee 670
2.3.2.6.3 Search employee 671
2.3.2.6.4 Upload Employee Document 671
2.3.2.6.5 Edit Employee Document 672
2.3.2.6.6 Search Employee document 672
2.3.2.6.7 Create Account 673
2.3.2.6.8 Edit Account 673
2.3.2.6.9 Search Account 674
2.3.2.6.10 Create Employee Type 674
2.3.2.6.11 Search Employee Type 675
2.3.2.6.12 Create Employee Skill 675
2.3.2.6.13 Search Employee Skill 675
2.3.2.7. Team Subsystem 676
2.3.2.7.1 Create Team 676
2.3.2.7.2 Search Team 676
2.3.2.7.3 Edit Team 677

2.3.2.8. Supplier Order Subsystem 678
2.3.2.8.1 Search Project Line-Item List 678
2.3.2.8.2 Edit Project Line-Item List 678
2.3.2.8.3 Create Supplier Purchase Order 679
2.3.2.8.4 Capture Supplier Quote 679
2.3.2.8.5 Search Supplier Quote 680
2.3.2.8.6 Search Supplier Purchase Order 680
2.3.2.9. Supplier Subsystem 681
2.3.2.9.1 Add Supplier 681
2.3.2.9.2 Search Supplier 681
2.3.2.9.3 Edit Supplier 682
2.3.2.10. Inventory Management Subsystem 683
2.3.2.10.1 Add-Line Item 683
2.3.2.10.2 Search-Line Item 683
2.3.2.10.3 Edit line-item 684
2.3.2.11. Reporting Subsystem 685
2.3.2.11.1 Generate Site Progress Report 685
2.3.2.11.2 Generate Site Type Report 685
2.3.2.11.3 Generate Team Site Completed Report 686
2.3.2.11.4 Generate Employee Document Expiry Report 686
2.3.2.11.5 Generate Site Acceptance Document 686
2.3.2.12. User Management Subsystem 687
2.3.2.12.1 Login 687
2.3.2.12.2 Logout 687
2.3.2.12.3 Reset Password 688
2.3.2.13. Client Subsystem 689
2.3.2.13.1 Add Client 689
2.3.2.13.2 Search Client 689
2.3.2.13.3 Edit Client 690
2.3.3 Primitive Level Date flow diagrams 691
2.3.3.1. Establishment Subsystem 691
2.3.3.1.1 Create TSS 691
2.3.3.1.2 Edit TSS 692
2.3.3.1.3 Submit TSS Report 693
2.3.3.2. Quotation Subsystem 694

2.3.3.2.1 Create Quote 694
2.3.3.2.2 Search Quote 695
2.3.3.2.3 Capture Purchase Order 696
2.3.3.3. Administration Subsystem 697
2.3.3.3.1 Create Health and Safety file 697
2.3.3.3.2 Search Health and Safety file 698
2.3.3.3.3 Create Health and Safety Type 699
2.3.3.3.4 Search Health and Safety Type 700
2.3.3.3.5 Create Drawing Type 701
2.3.3.3.6 Search Drawing Type 702
2.3.3.3.7 Upload Formal Drawing 703
2.3.3.3.8 Upload Second cut Drawing 704
2.3.3.3.9 Search As-built document 705
2.3.3.3.10 Generate As-Built Document 706
2.3.3.4. Construction Subsystem 707
2.3.3.4.1 Capture Site-Go-Ahead 707
2.3.3.4.2 Search construction 708
2.3.3.4.3 Edit construction 709
2.3.3.4.4 Allocate Team Site 710
2.3.3.4.5 Search Allocated Team 711
2.3.3.4.6 Update Allocated Team 712
2.3.3.4.7 Upload Compliance Document 713
2.3.3.4.8 Search Compliance Document 714
2.3.3.4.9 Search Project 715
2.3.3.4.10 Create Site Progress description 716
2.3.3.4.11 View Site Progress description 717
2.3.3.4.12 Upload complete site photos 718
2.3.3.4.13 Search complete site photos 719
2.3.3.4.14 Upload Radio Frequency report 720
2.3.3.4.15 Search Radio Frequency report 721
2.3.3.4.16 Request Site Visit 722
2.3.3.4.17 Create Snag 723
2.3.3.4.18 Search Snag 724
2.3.3.4.19 Edit Snag 725
2.3.3.4.20 Edit Acceptance Document 726

2.3.3.4.21 Search Acceptance document 727
2.3.3.4.22 Confirm acceptance document 728
2.3.3.4.23 Capture Site Sign-off 729
2.3.3.4.24 Search Site Sign-off 730
2.3.3.4.25 Capture Commission report 731
2.3.3.4.26 Search Commission report 732
2.3.3.4.27 Create Site Type 733
2.3.3.4.28 Search Site type 734
2.3.3.4.29 Edit Site Type 735
2.3.3.5. Invoicing Subsystem 736
2.3.3.5.1 Search Invoice 736
2.3.3.5.2 Send Invoice 737
2.3.3.5.3 Capture Proof of Payment 738
2.3.3.5.4 Capture Claim 739
2.3.3.5.5 Search Proof of Payment 740
2.3.3.5.6 Search Invoice Type 741
2.3.3.5.7 Create Invoice Type 742
2.3.3.6. Human Resource Subsystem 743
2.3.3.6.1 Add employee 743
2.3.3.6.2 Edit employee 744
2.3.3.6.3 Search employee 745
2.3.3.6.4 Upload Employee Document 746
2.3.3.6.5 Edit Employee Document 747
2.3.3.6.6 Search Employee document 748
2.3.3.6.7 Create Account 749
2.3.3.6.8 Edit Account 750
2.3.3.6.9 Search Account 751
2.3.3.6.10 Create Employee Type 752
2.3.3.6.11 Search Employee Type 753
2.3.3.6.12 Create Employee Skill 754
2.3.3.6.13 Search Employee Skill 755
2.3.3.7. Team Subsystem 756
2.3.3.7.1 Create Team 756
2.3.3.7.2 Search Team 757
2.3.3.7.3 Edit Team 758

2.3.3.8. Invoicing Subsystem 759
2.3.3.8.1 Search Project Line-Item List 759
2.3.3.8.2 Edit Project Line-Item List 760
2.3.3.8.3 Create Supplier Purchase Order 762
2.3.3.8.4 Capture Supplier Quote 763
2.3.3.8.5 Search Supplier Quote 764
2.3.3.8.6 Search Supplier Purchase Order 765
2.3.3.9. Supplier Subsystem 766
2.3.3.9.1 Add Supplier 766
2.3.3.9.2 Search Supplier 767
2.3.3.9.3 Edit Supplier 768
2.3.3.10. Inventory Management Subsystem 769
2.3.3.10.1 Add-Line Item 769
2.3.3.10.2 Search-Line Item 770
2.3.3.10.3 Edit line-item 771
2.3.3.11. Reporting Subsystem 772
2.3.3.11.1 Generate Site Progress Report 772
2.3.3.11.2 Generate Site Type Report 773
2.3.3.11.3 Generate Team Site Completed Report 774
2.3.3.11.4 Generate Employee Document Expiry Report 775
2.3.3.11.5 Generate Site Acceptance Document 776
2.3.3.12. User Management Subsystem 777
2.3.3.12.1 Login 777
2.3.3.12.2 Logout 777
2.3.3.12.3 Reset Password 778
2.3.3.13. Client Subsystem 779
2.3.3.13.1 Add Client 779
2.3.3.13.2 Search Client 780
2.3.3.13.3 Edit Client 781
Conclusion 781
2.4 Pseudo Code 782
System Database Design 783
3. Entity Relationship Model 783
Introduction 783
3.1 Entity Relationship Model 784

Conclusion 785
4. Size Estimation of proposed database 786
Introduction 786
4.1 Size Estimation of proposed Database 787
4.2 Five Year Projection 794
Conclusion 794
System Interface Design 795
5. Input Design 795
Introduction 795
5.1 Complete Input design description 795
Conclusion 795
6. Output Design 796
Validate the details of the functional specification 796
Introduction 796
6.1 Complete Output design 797
6.1.1.11 Report Subsystem 797
6.1.1.11.1 Generate Site Progress Report 797
6.1.1.11.2 Generate Site Type Report 800
6.1.1.11.2 Generate Site Type Report 801
6.2 Message Outputs 804
6.2.1 Establishment Subsystem 804
6.2.2 Quotation Subsystem 806
6.2.3 Administration Subsystem 814
6.2.4 Construction Subsystem 818
6.2.5 Invoicing Subsystem 825
6.2.6 Human Resource Subsystem 827
6.2.7 Teams Subsystem 840
6.2.8 Supplier Order Subsystem 842
6.2.9 Supplier Subsystem 843
6.2.10 Inventory Management Subsystem 844
6.2.11 Report Subsystem 846
6.2.12 User Management Subsystem 847
6.2.13 Client Subsystem 847
Conclusion 849
Package Design Specification 850

7. Test Specification 850
Introduction ..... **Error! Bookmark not defined.**
Introduction 850
7.1 Test Plan 850
7.2 Test scenarios 852
7.3 Test Data 863
Conclusion 863
8. Hardware and Software Specification 864
Introduction 864
8.1 Hardware and software 864
8.1.1 Software requirements 864
8.1.2 Hardware requirements 864
Conclusion 865
9. Network/ Web Lay out Specification 866
Introduction 866
9.1 Network/Web Layout 866
9.1.1 Network Specifications 866
9.1.2 Network Diagram 866
Conclusion 866
10. Validation 867
Introduction 867
10.1 Validate the technical specifications against requirements 867
10.1.1 Establishment Subsystem 867
10.1.2 Quotation Subsystem 869
10.1.3 Administration Subsystem 871
10.1.4 Construction Subsystem 876
10.1.5 Invoicing Subsystem 898
10.1.6 Human Resource Subsystem 904
10.1.7 Teams Subsystem 912
10.1.8 Supplier Order Subsystem 914
10.1.9 Supplier Subsystem 921
10.1.10 Inventory Management Subsystem 923
10.1.11 Report Subsystem 925
10.1.12 User Management Subsystem 930
10.1.13 Client Subsystem 934

Conclusion....936
Project Related Documents....937
11. Sign-off by The Client....937
Introduction....937
11.1 Client Sign off....937
Conclusion....937
12. Complexity Matrix....938
Introduction....938
12.1 Complexity Matrix....938
Conclusion....940
13. Sign-off by The Team....941
Introduction....941
11.1 Team Sign off....941
Conclusion....941
Deliverable Conclusion....942

Deliverable Introduction

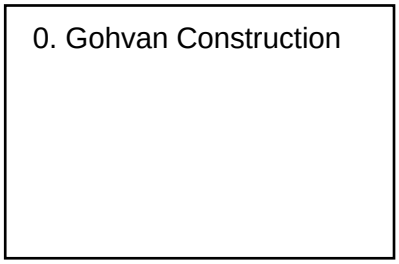
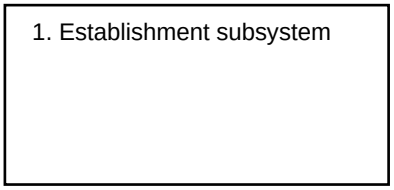
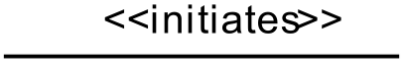
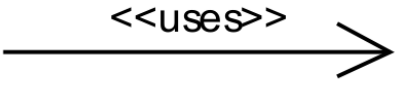
This document follows on the signed-off functional requirements listed in the previous deliverable by expanding and explaining them through use case diagrams, use case narratives, and technical data flow diagrams. The document also contains an entity relationship model which explains how all the relevant entities in the system connect with one another and what attributes they contain. This deliverable also explains the hardware, software and network layout to help implement the proposed system. Finally, the document contains inputs, outputs and validation of the functional requirements diagrams that help explain the flows and describe the overall layout and presentation of the system.

System Architecture Design

1. Use Cases

Introduction

This section contains the complete set of use case diagrams, as well as the complete set of use case narratives for the Gohvan Construction System. These will provide the logical view of the system.

Use Case Diagram Key		
Element	Description	Preview
System	Entity element, filled with white and identified by "**0. Gohvan Construction**".	
Subsystem	Use Case rectangle element, filled with white & black text and outline.	
Use Case	Use Case use case element, with black text, filled with a light blue	
Actor	Element, with relevant relationship types, with light blue fill and outline.	
Association Relationship	Use Case association element, identified by the <<initiates>> tag	
Uses Association Relationship	Use Case dependency element, identified by the <<uses>> tag	

1.1 Use case diagrams

1.1.1 Establishment Subsystem

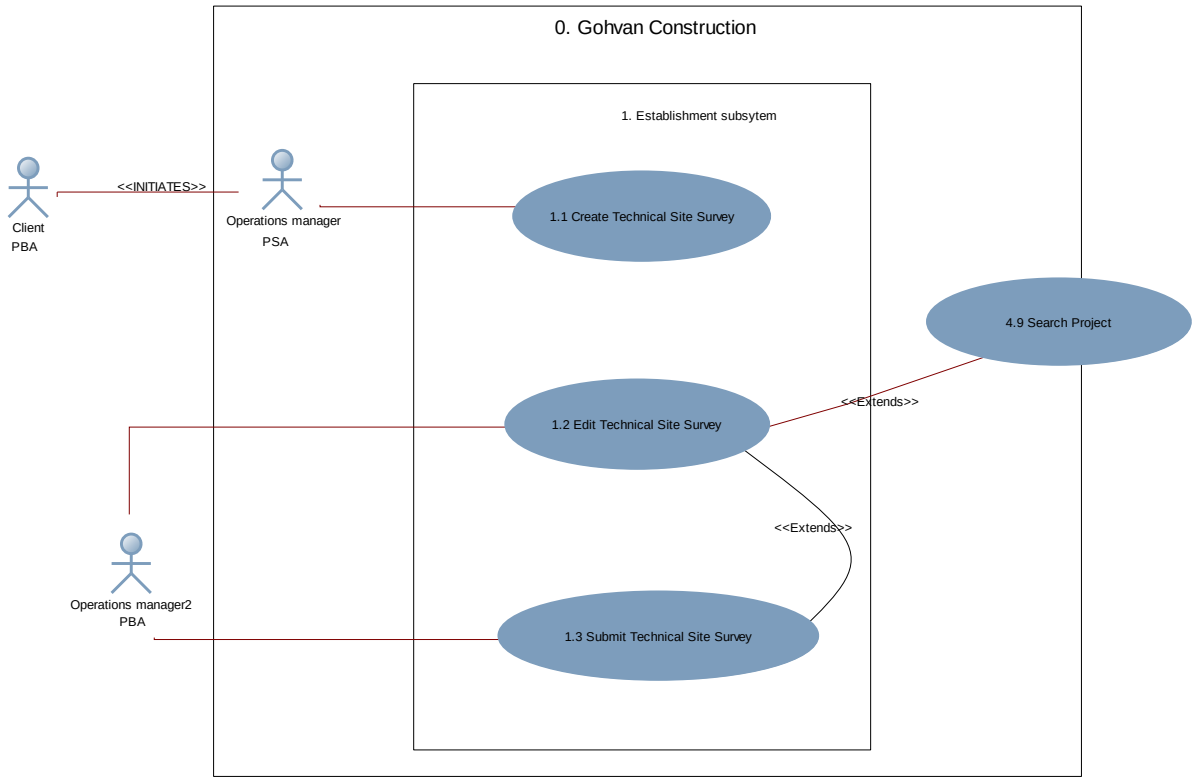


Figure 1 Use Case Diagram Establishment Subsystem

1.1.2 Quotation Subsystem

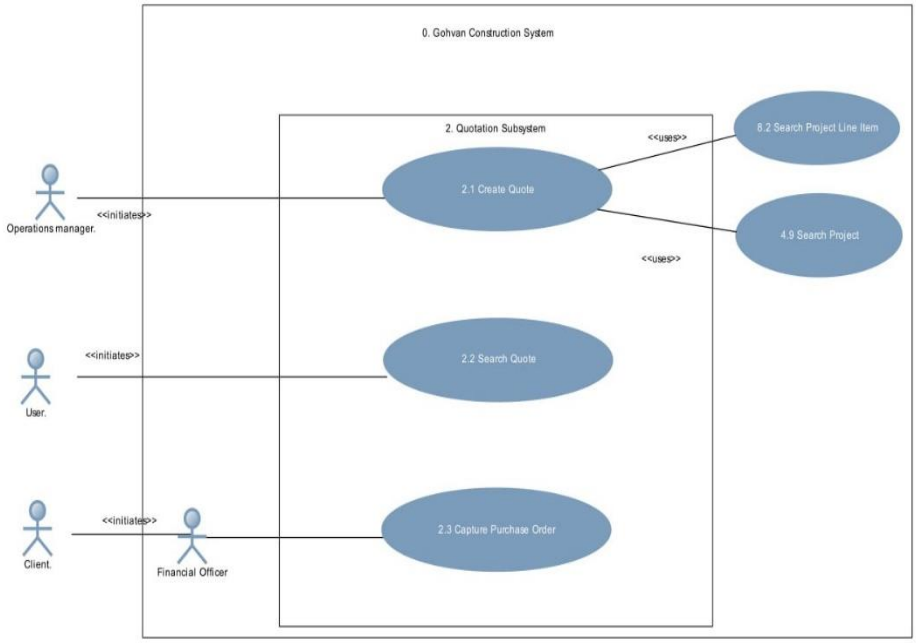


Figure 2: Use Case Diagram Quotation Subsystem

1.1.3 Administration Subsystem

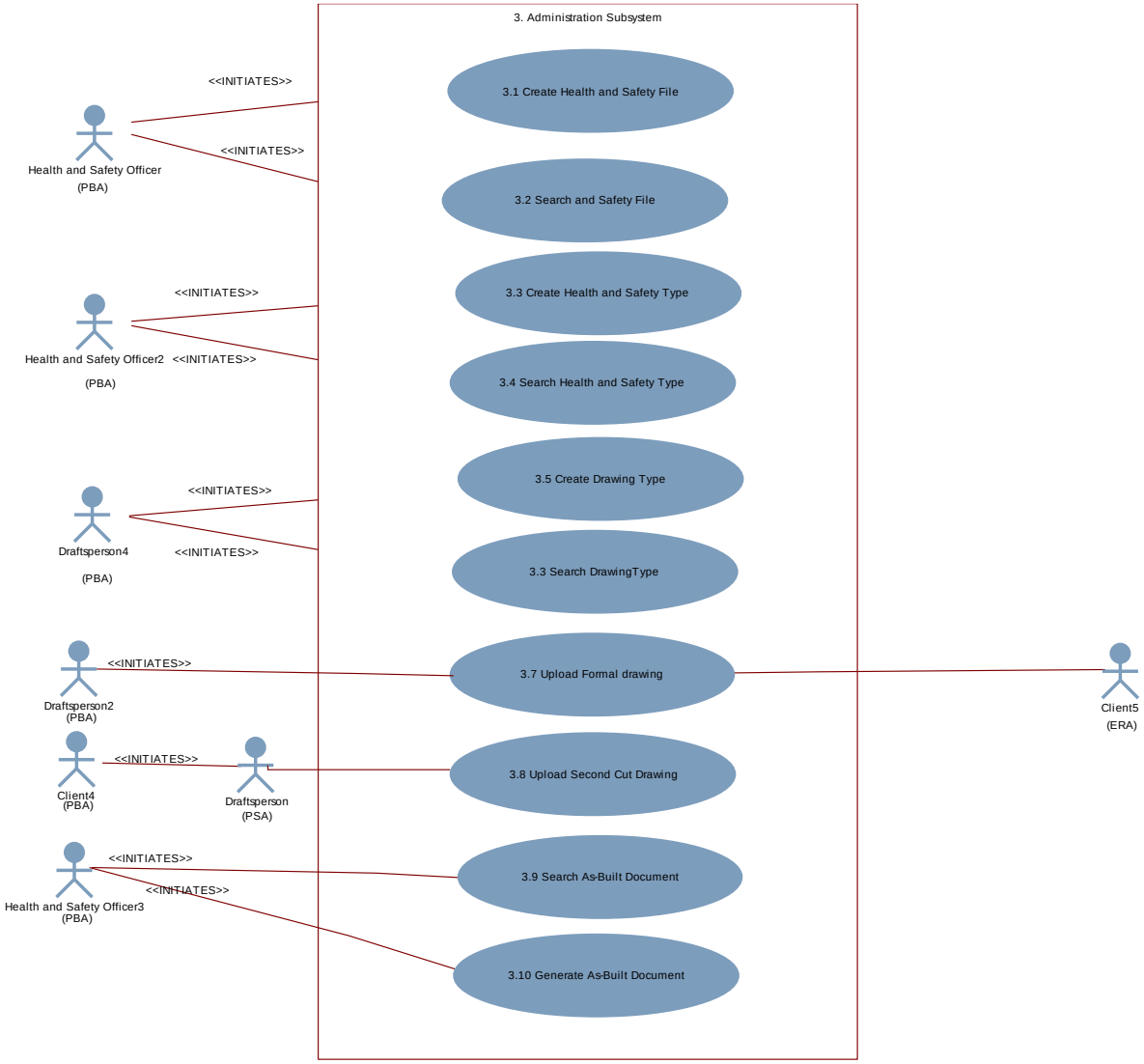


Figure 3: Use Case Diagram Administration Subsystem

1.1.4 Construction Subsystem

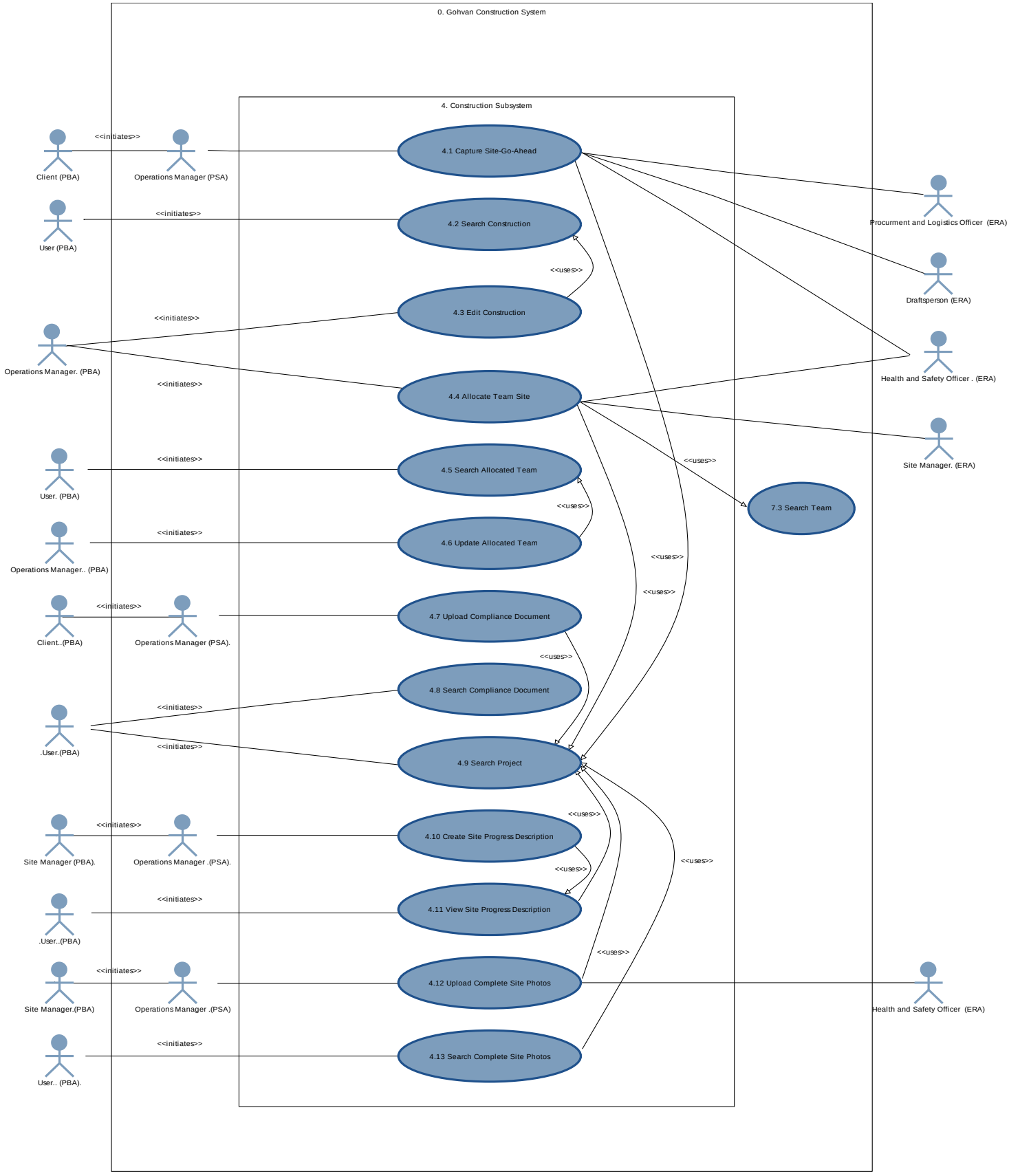


Figure 4: Use Case Diagram Construction Subsystem 4.1-4.13

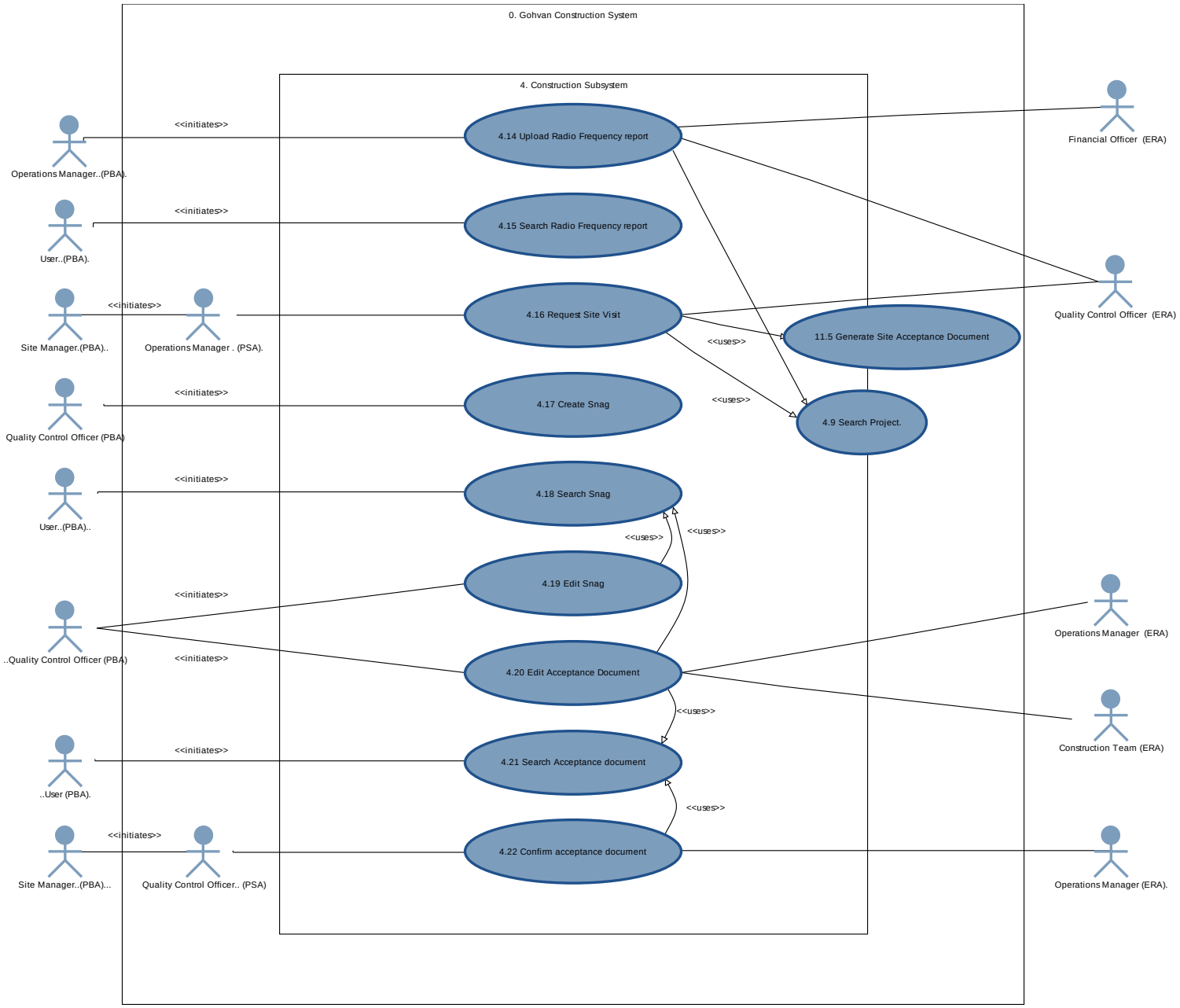


Figure 5: Use Case Diagram Construction Subsystem 4.14 - 4.22

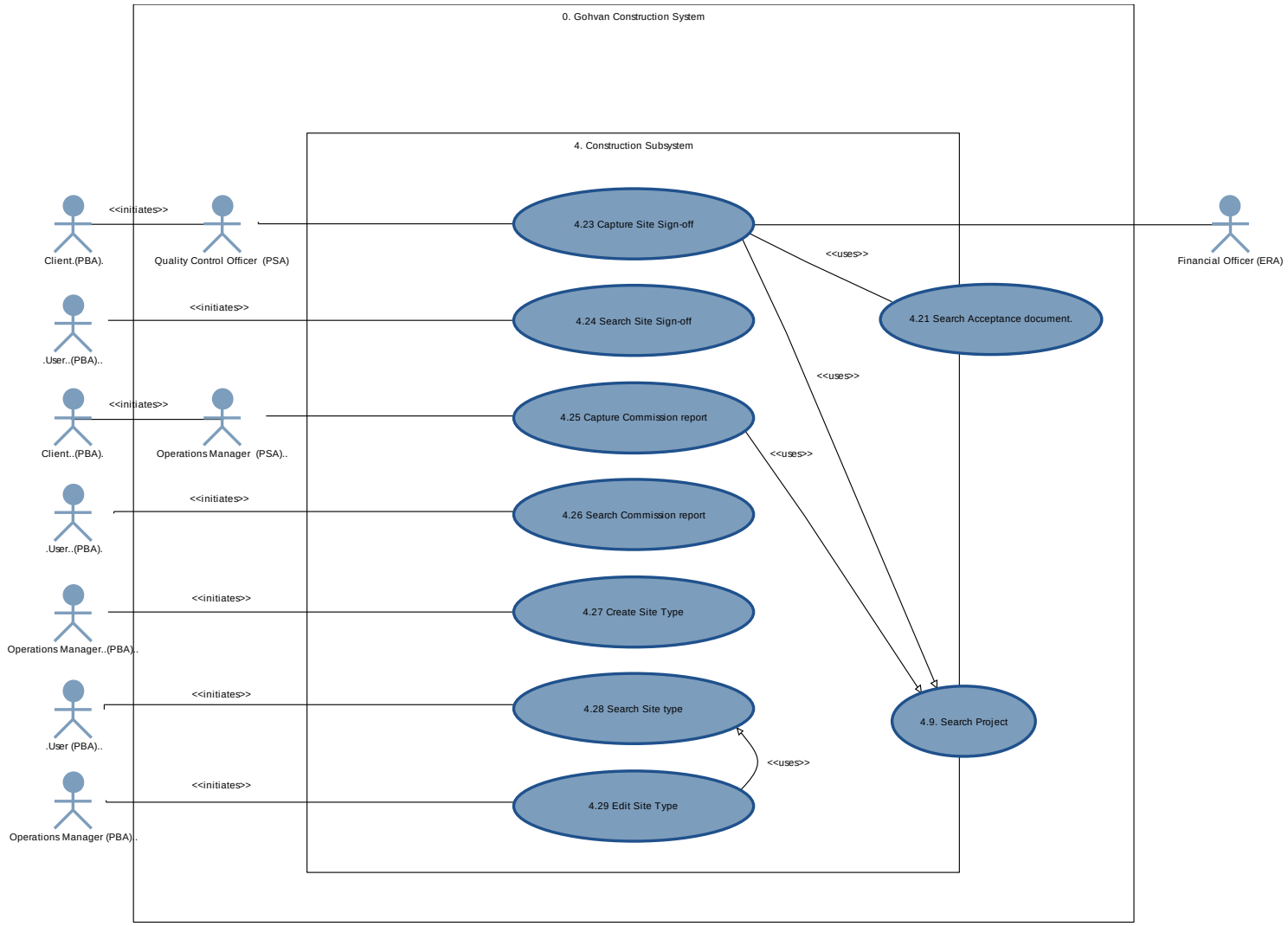


Figure 6 Use Case Diagram Construction Subsystem 4.23-4.29

1.1.5 Invoicing Subsystem

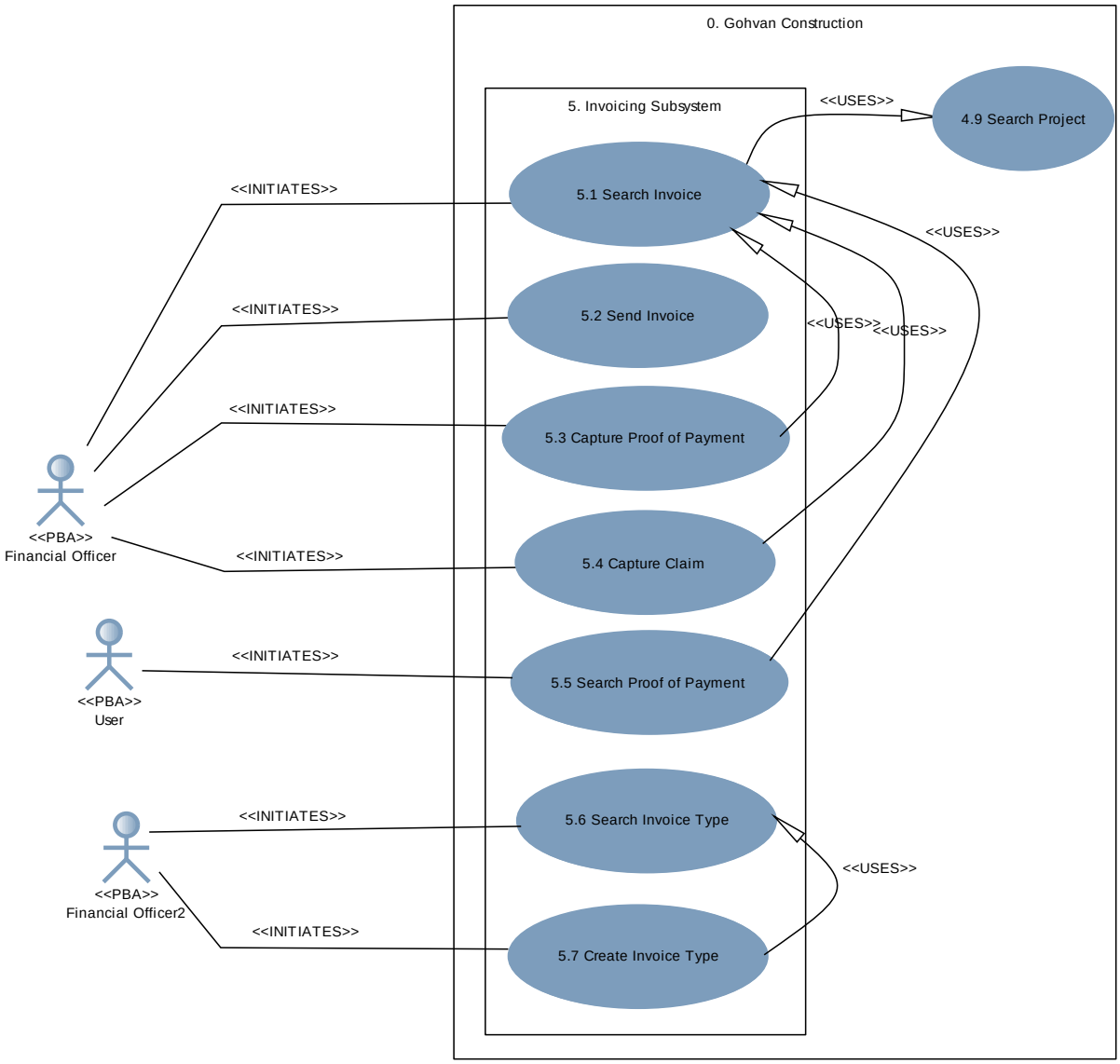


Figure 7: Use Case Diagram Invoicing Subsystem

1.1.6 Human Resource Subsystem

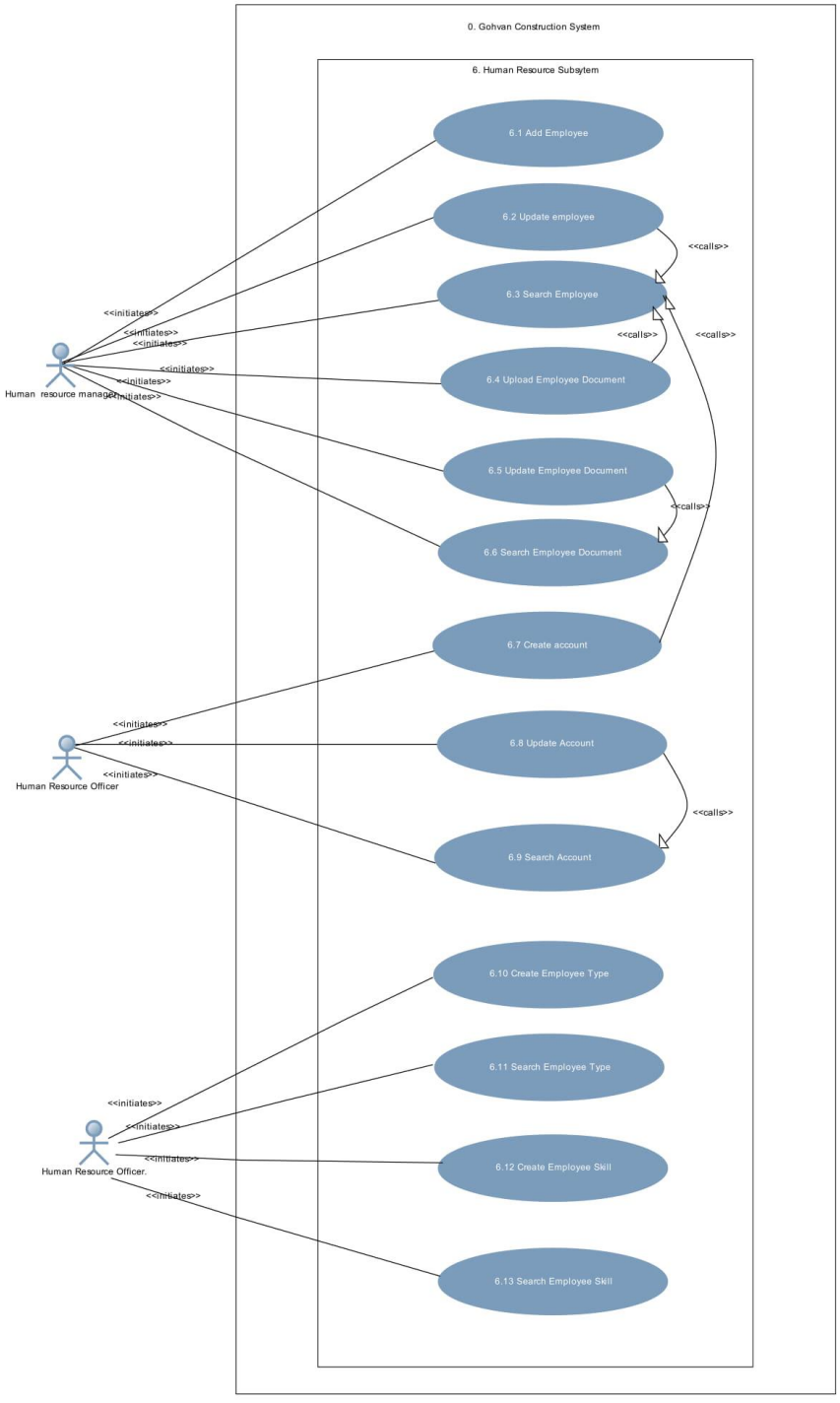


Figure 8: Use Case Diagram Human Resource Subsystem

1.1.7 Teams Subsystem

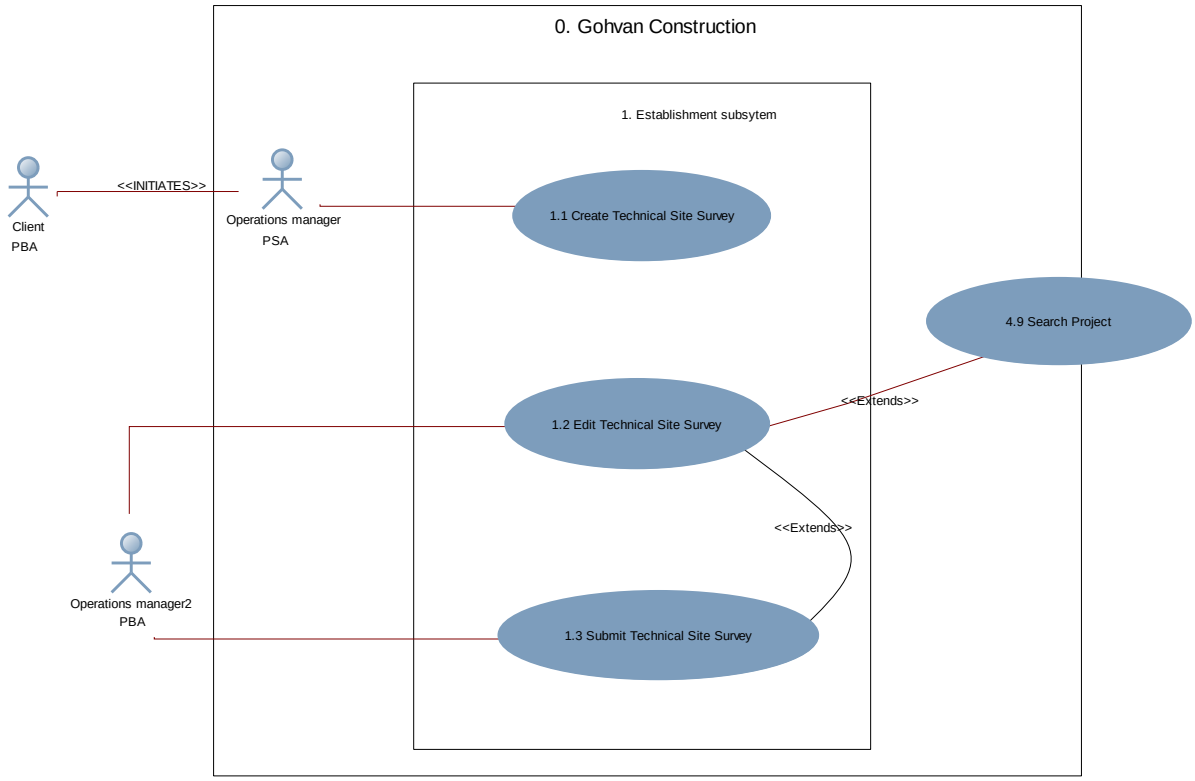


Figure 9: Use Case Diagram Team Subsystem

1.1.8 Supplier Order Subsystem

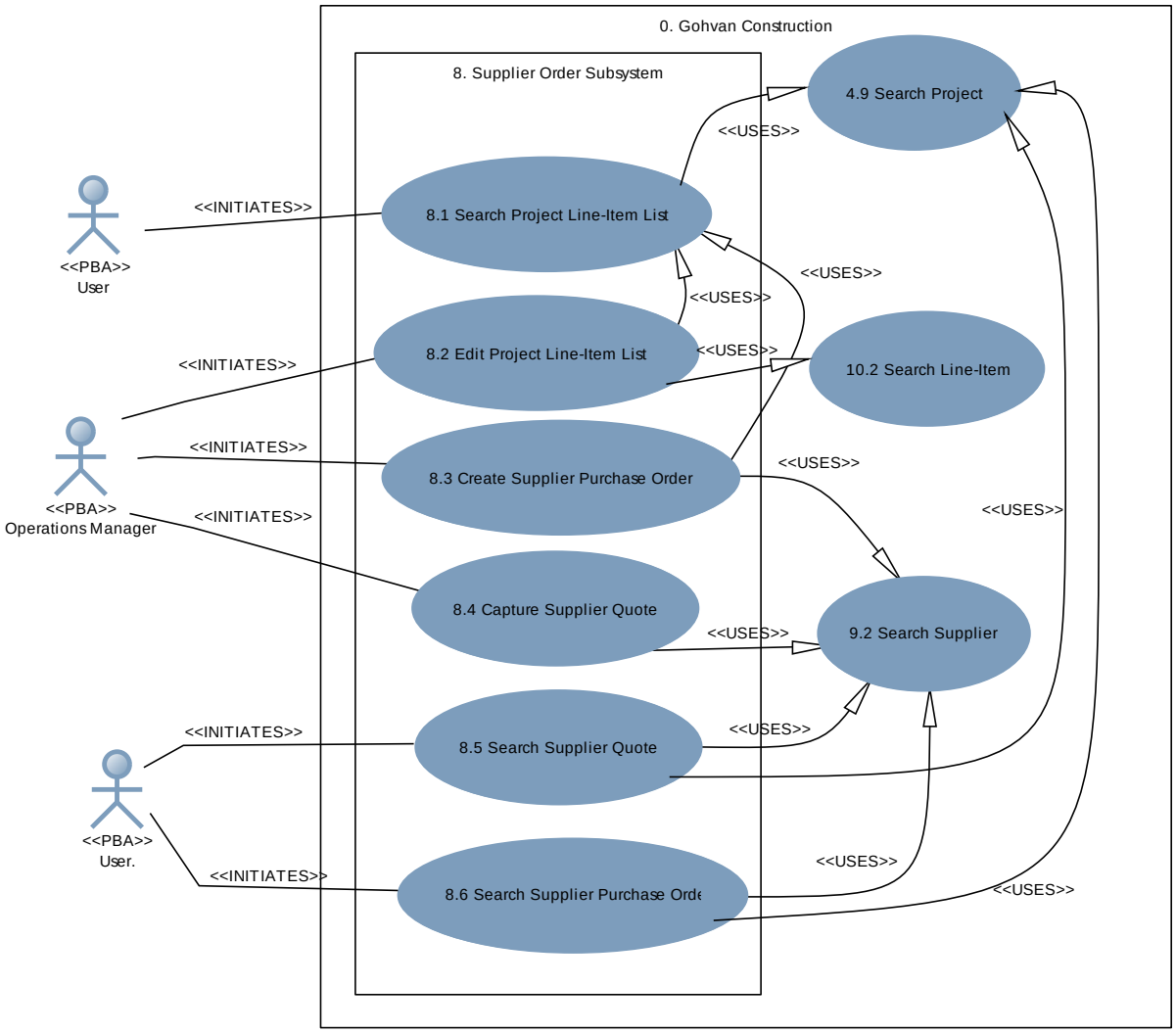


Figure 10: Use Case Diagram Supplier Order Subsystem

1.1.9 Supplier Subsystem

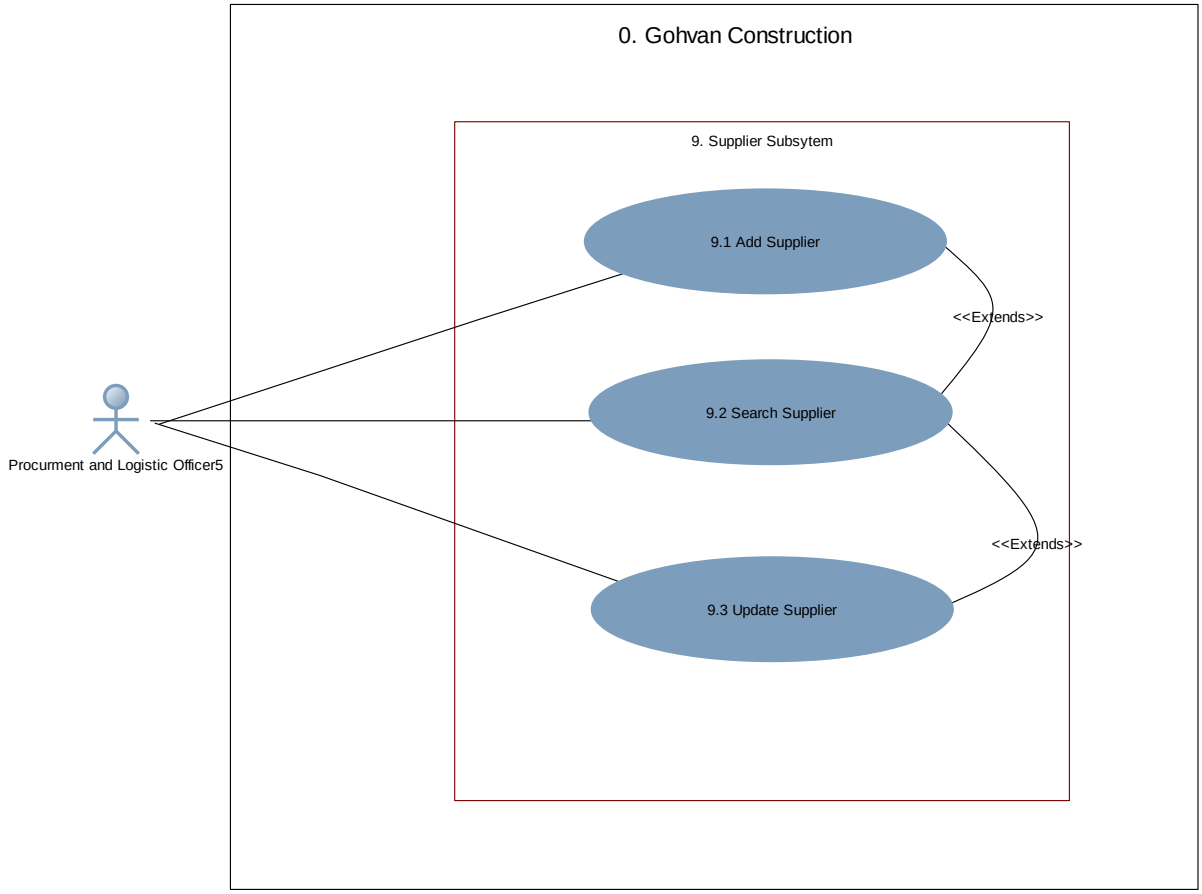


Figure 11: Use Case Diagram Supplier Subsystem

1.1.10 Inventory Management Subsystem

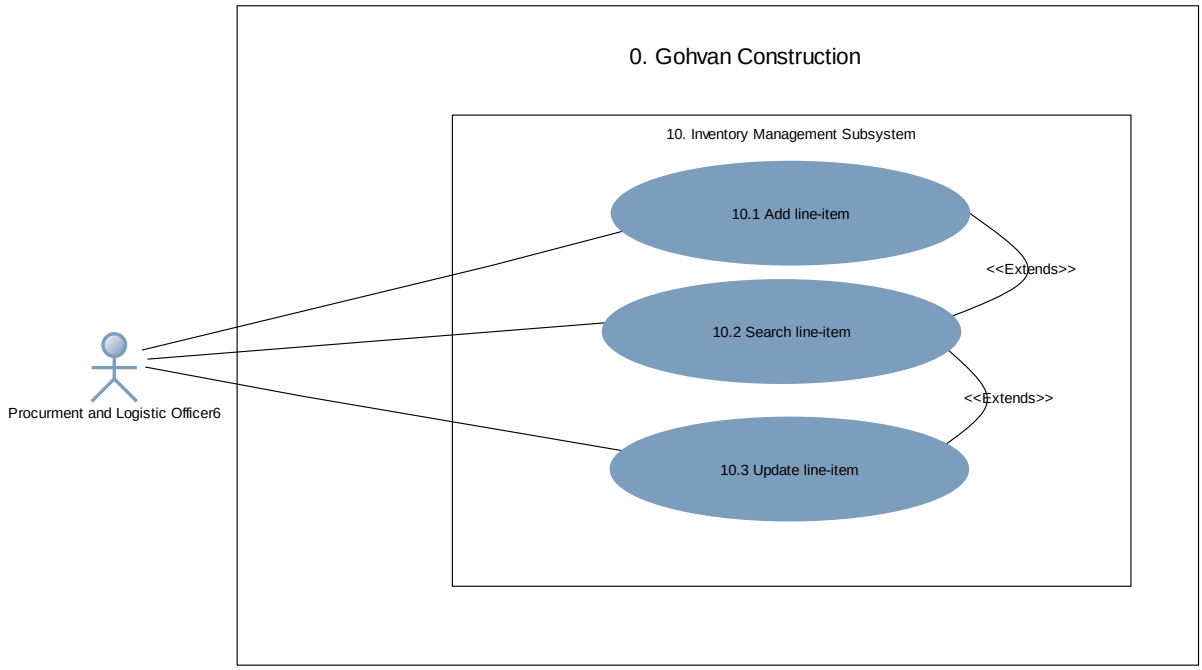


Figure 12: Use Case Diagram Inventory Management Subsystem

1.1.11 Report Subsystem

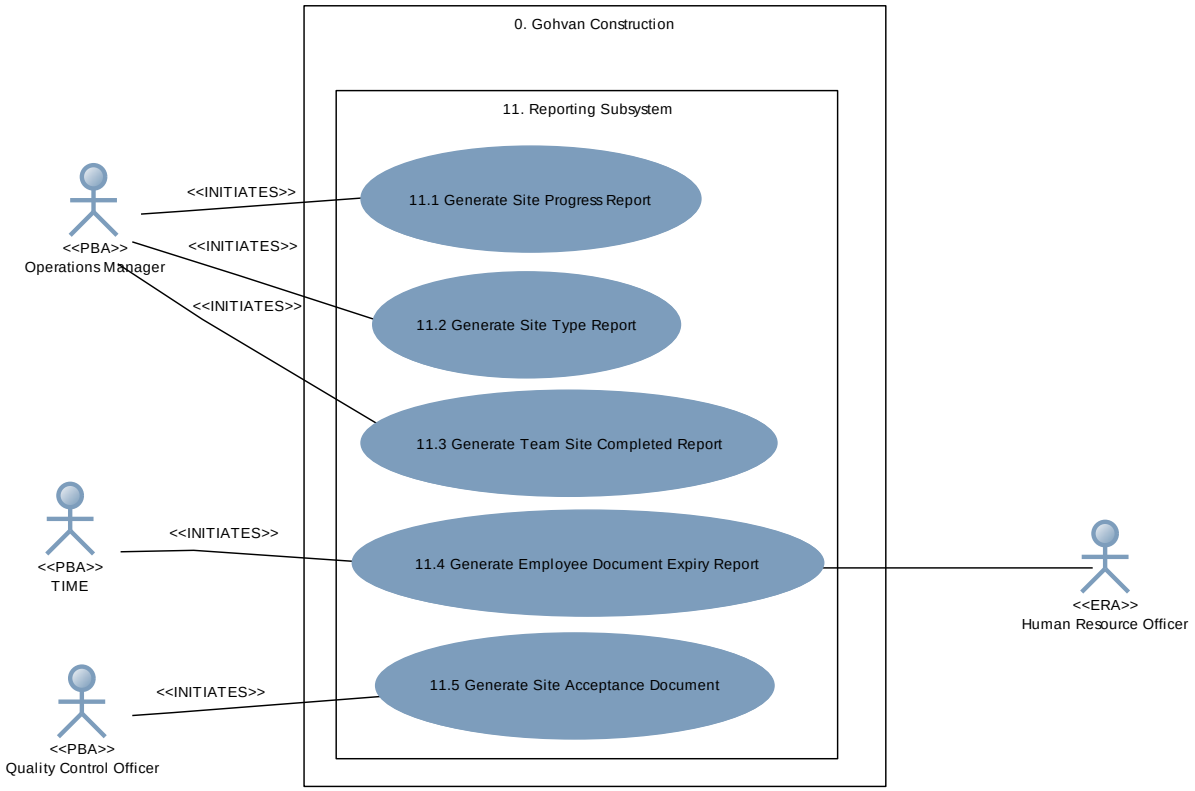


Figure 13: Use Case Diagram Report Subsystem

1.1.12 User Management Subsystem

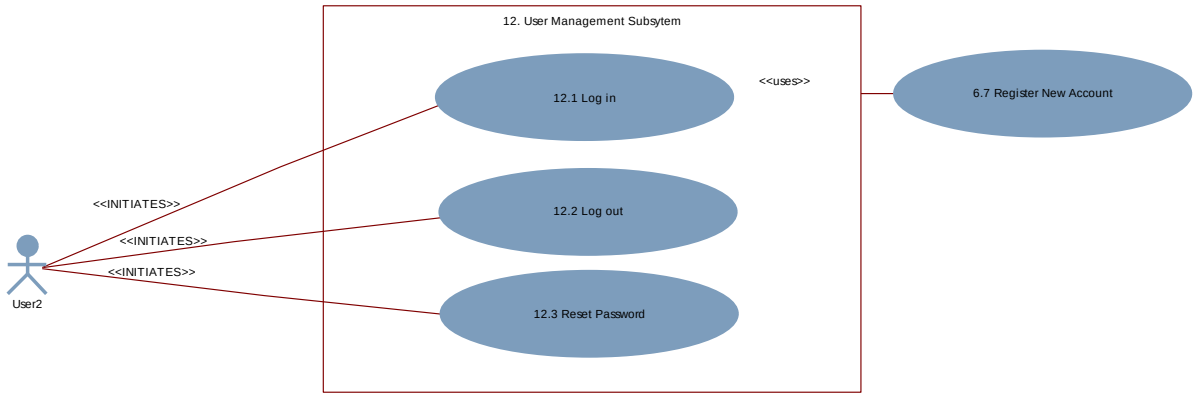


Figure 14: Use Case Diagram User management- Subsystem

1.1.13 Client Subsystem

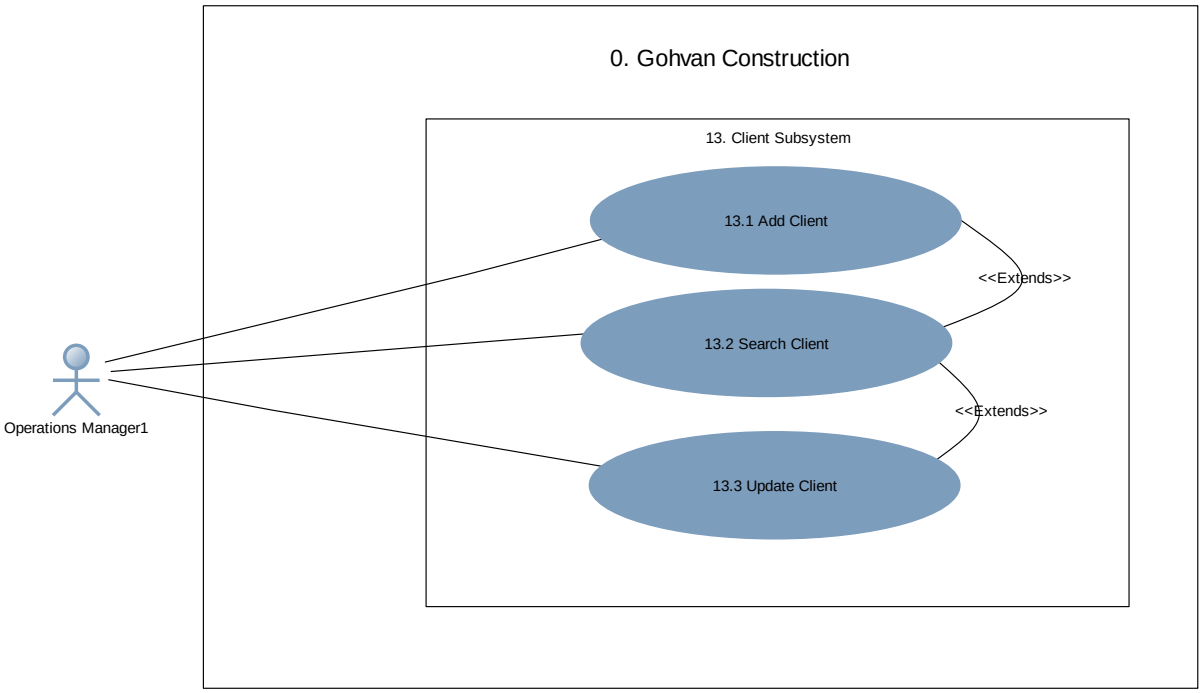


Figure 15: Use Case Diagram Client Subsystem

1.2 Use case Narratives

1.2.1. Establishment Subsystem

1.2.1.1 Create TSS

Gohvan Construction			
Author (s): **Amore Rossouw**	**Date: 28/06/2021**	**Version: 1.0**	
USE CASE NAME:	**Create TSS**	**USE CASE TYPE**	
USE CASE ID:	**1.1**	**Abstract:**	☐
PRIORITY:	High	**Extension:**	☑
SOURCE:	Gohvan Construction System		
PRIMARY BUSINESS ACTOR:	Operations Manager (PSA)		
PRIMARY THE SYSTEM ACTOR:	none		
OTHER PARTICIPATING ACTORS:	• None		
OTHER INTERESTED STAKEHOLDERS:	• None		
DESCRIPTION:	This use case narrative describes the process the operations manager will follow when they want to create a **new technical site survey (TSS)**. The client will send a request for construction of a new site which will require the operations manager to do a technical site survey. A new technical site survey is created on site. The operations manager will be prompted to fill in a form describing the site type, site information and the construction materials required for building the site. The system will capture and validate the site survey information and if the information was validated successfully the system will store it in the **TSS** table, **TSS PHOTO** table and **PROJECT LINE ITEMS** table.		
PRE-CONDITION:	•The operations manager needs to be logged into the system before creating a new TSS.		
TRIGGER:	The operations manager receives a new site request from the client and wants to create a new TSS.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
			Step 1:

The system uses a SQL read query to read the SiteType table, in order to get the following attributes for the “Select SiteType” dropdown:

- SiteTypeID
- Description

The system responds by loading the “Create TSS” Screen with TSS table information using the SQL Read query with the following controls:

Input Name	Control Type	Notes
Technical Site Survey	**Header**	**none**
Select Site Type	Label	none
Select Site Type	Select drop-down list	Populated with the Description attribute of each Site Type read. The value of each item in the drop down is set to the corresponding SiteTypeID.
Survey Date	label	none
Date	Date picker	This control is prepopulated with the current date.
Site Details	**Header**	**none**
Site Name	Label	none
Site Name	Input	none
Bs Number	Label	none
Bs Number	Input	This is the input for the base station number
Site Description	Label	none
Site Description	Text box	none
Project number	Label	none
Project Number	Input	none
GPS Coordinates	label	none

			GPS Map	Google maps	The system will use the geolocation function to get the current coordinates of the user
			Get Coordinates	button	none
			GPS Coordinates	Input	If the user does not want to user, the geolocation function they can manually type in the coordinates.
			Distance	Label	none
			Distance	Input	none
			Time Driving	Label	none
			Time Driving	Input	none
			Quotation Date	Label	none
			Date	Date picker	none
			Pre-Approval	**Header**	**none**
			Site drawings available	Label	none
			Site drawings available	Checkbox	none
			Landlord approval obtained	Label	none
			Landlord approval obtained	Checkbox	none
			CCA	Label	none
			CCA	Checkbox	none
			EIA	Label	none
			EIA	Checkbox	none
			BP approval obtained	Label	none
			BP approval obtained	Checkbox	none

			Landlord details	label	none
			Landlord details	Text box	none
			Special Conditions	label	none
			Special Conditions	Text box	none
			Accommodation	**Header**	**none**
			Container	Header	none
			Container Type	Label	none
			Container Type	Select drop-down list	Populated with the options “ARP 1 Door H&S Security Cab”, “ARP 2 Door H&S Security Cab”, “Mc Wade SV140” and “2x RED 1's”
			Container Other	Label	none
			Container Other	input	none
			Container Size	Label	none
			Container Size	Input	none
			Container size type	Label	none
			Container size type	Select drop-down list	Populated with the options “standard” and “other”.
			Container Paint	Label	none
			Container Paint	Checkbox	None
			Load Support	Label	none
			Load support	input	None
			Brick Cladding	Label	none
			Brick cladding	input	None
			Rock finish	Label	none
			Rock finish	Checkbox	None
			Door Direction	Label	none
			Door Direction	Select drop-down list	Populated with the options “North”, “East”,

					"South" and "West".
			Notes	label	none
			Notes	Textbox	none
			Plinth	Header	none
			Apron	label	none
			Apron	Checkbox	none
			Plinth Type	Label	none
			Plinth Type	Select drop-down list	Populated with the options "new", "flat on apron" and "blocks"
			Plinth Size	Label	none
			Plinth Size	Input	none
			Site Preparation details	**Header**	**none**
			Groundwork	Header	none
			Clear and clean site	Label	none
			Clear and clean site	Checkbox	None
			Other requirements	Label	none
			Site preparation requirements	Textbox	none
			Mast Details	**Header**	**none**
			Mast	Header	none
			Select Mast Type	Label	none
			Select Mast Type	Select drop-down list	Populated with the options: "Lattice", "Monopole" and "Monalatt".
			Select Mast height	Label	none
			Select Mast height	Select drop-down list	Populated with the options: "25", "35" and "other".
			Other	Label	none
			Mast height	Input	This control

					will only be enabled once the user selects 'other' from the "Select Mast height" control.
			Mast Accessories	Header	none
			Navigation light	Label	none
			Navigation light	Checkbox	None
			Fall arreast	Label	none
			Fall arreast	Checkbox	None
			Mast Paint	Label	none
			Mast Paint	Checkbox	None
			Environmental Adaptation	Label	
			Environmental Adaptation	Select drop-down list	Populated with the options: "Tree", "Palm tree" and 'other".
			Other	Label	none
			Environmental Adaptation other	Input	This control will only be enabled once the user selects 'other' from the "Environ mental Adaptation" control.
			Mast notes	Label	none
			Mast notes	Textbox	none
			RF Installation Details	**Header**	**none**
			Cable Management	Header	none
			RF on cable Tray	Header	none
			RF on Cable Tray	Label	none
			RF on Cable Tray	Checkbox	None
			Cable Tray Type	Label	none

			Cable Tray Type	Select drop-down list	Populated with the options: “114”, “228”, ‘304” and “other”.
			Other	Label	none
			Cable Tray Type other	Input	This control will only be enabled once the user selects ‘other’ from the “Cable Tray Type” control
			Length	label	none
			Length	Input	none
			Bends	Label	none
			Select Bends	Select drop-down list	Populated with the options: “90” and “other”.
			Other	Label	none
			Bends other	Input	This control will only be enabled once the user selects ‘other’ from the “select Bends” control
			Quantity	Label	none
			Bends Qty	Input	none
			Support Pole	Label	none
			Support Pole	Checkbox	None
			Quantity	Label	none
			Support Pole Qty	Input	This control will only be enabled once the user selects yes from the Support pole checkbox.
			Mounting Type	Label	none
			Select Mounting Type	Select drop-down list	Populated with the options: “Cantelvr Arms”, “Concrete

					Arms", "Concrete Blks" and "other".
			Other	Label	none
			Mounting Type other	Input	This control will only be enabled once the user selects 'other' from the "select Mounting Type" control
			Tray Covers	Label	none
			Tray Covers	Checkbox	None
			Quantity	Label	none
			Tray Covers Qty	Input	This control will only be enabled once the user selects yes from the Tray Covers checkbox.
			Cable Tray Painted	label	none
			Cable Tray Painted	Checkbox	None
			RF sleeves	Header	none
			RF in sleeves	label	none
			RF in sleeves	Checkbox	None
			Length	Label	none
			RF sleeve Length	Input	This control will only be enabled once the user selects yes from the RF in sleeves checkbox.
			Size	label	none
			RF sleeve Size	Input	This control will only be enabled once the user selects yes from the RF in sleeves checkbox.

Electrical Details	Header	none
AC Supply	Label	none
Select AC Supply	Select drop-down list	Populated with the options: “Site DB”, “Site Container” and “other”.
Other	Label	none
AC Supply other	Input	This control will only be enabled once the user selects ‘other’ from the “select AC Supply” control
Power Available	Label	none
Power Available	Select drop-down list	Populated with the options: “SP” and “3P”.
Power Required	Label	none
Power Required	Select drop-down list	Populated with the options: “SP” and “3P”.
Generator Required	label	none
Generator Required	Checkbox	None
Cable Size	Label	none
Select Cable Size	Select drop-down list	Populated with the options: “16mm³”, “25mm³”, “30mm³” and “other”.
Other	Label	none
Cable Size other	Input	This control will only be enabled once the user selects ‘other’ from the “select Cable Size” control
Cable Length	Label	none
Cable Length	Input	none

			In-Site Cable Route	Label	none
			Select In-Site Cable Route	Select drop-down list	Populated with the options: “Bosal” and “Sleeves”
			In-Site Cable Route Type	Label	none
			Select In-Site Cable Route Type	Select drop-down list	Populated with the options: “Conduit”, “Cable Tray” and “other”.
			Other	Label	none
			In-Site Cable Route Type other	Input	This control will only be enabled once the user selects ‘other’ from the “Select In-Site Cable Route Type” control
			In-Site Cable Route length	Label	none
			In-Site Cable Route length	Input	none
			Consumption Meter Required	Label	none
			Consumption Meter Required	Checkbox	None
			Site Light Position	Label	none
			Select Site Light Position	Select drop-down list	Populated with the options: “North”,” East”, “South “and “West”.
			Earthing details	Header	none
			Earthing Installation	Label	none
			Select Earthing Installation	Select drop-down list	Populated with the options: “New”, and” Existing”

Earthing Type	Label	none
Select Earthing Type	Select drop-down list	Populated with the options: “Green and yellow”, “Black” “Bare Copper” and” Other”
Other	Label	none
Earthing Type other	Input	This control will only be enabled once the user selects ‘other’ from the “Select Earthing Type” control
Construction Details	**Header**	**none**
Fencing	Header	none
Fence Details	Label	none
Select Fence Details	Select drop-down list	Populated with the options: “In-ground”, and “Above-ground”
Fence Type	Label	none
Select Fence Type	Select drop-down list	Populated with the options: “Palisade”, “Round table” and” Other”
Other	Label	none
Fence Type other	Input	This control will only be enabled once the user selects ‘other’ from the “Select Fence Type” control
Fence Dimensions	Label	none
Fence Dimensions	Input	none
Surface Finish	Label	none

			Select Surface Finish	Select drop-down list	Populated with the options: “Semi-Galv” and “Paint”
			Colour	label	none
			Fence Colour	Input	none
			Additional Extras	Label	none
			Select Fence Additional Extras	Select drop-down list	Populated with the options: “Razor Coil”, “Flat wrap” and “other”
			Other	Label	none
			Fence Additional Extras other	Input	This control will only be enabled once the user selects ‘other’ from the “Select Fence Additional Extras” control
			Gate Type	Label	none
			Select Gate Type	Select drop-down list	Populated with the options: “Swing” and “Slide”
			Gate opening direction	Label	none
			Select Gate opening direction	Select drop-down list	Populated with the options: “Inward” and “Outward”
			Gate Dimensions	Label	none
			Gate Dimensions	Input	none
			Site Security	Header	none
			Locks Required	Label	none
			Locks Required	Checkbox	None
			Lock Type	Label	none
			Select Lock Type	Select drop-down list	Populated with the options: “VCL”, “VM”, “NM” and “ARN”
			Shackle Length	Label	none

			Select Shackle Length	Select drop-down list	Populated with the options: “24mm” and “50mm”
			Lock Chain Required	Label	none
			Lock Chain Required	Checkbox	None
			Notes	Label	none
			Site Security Notes	Textbox	none
			Site Access	Header	none
			Clear Road	Label	none
			Clear Road	Checkbox	None
			Access Road	Label	none
			Select Access Road	Select drop-down list	Populated with the options: “Existing tar”, “Existing gravel” and “other”
			Other	Label	none
			Access Road other	Input	This control will only be enabled once the user selects ‘other’ from the “Select Access Road” control
			Road Suitability	Label	none
			Select Road Suitability	Select drop-down list	Populated with the options: “Car”, “4x4” and “other”
			Other	Label	none
			Road Suitability other	Input	This control will only be enabled once the user selects ‘other’ from the “Select Road Suitability” control
			Notes	Label	none
			Site Access Notes	Textbox	none

			Site Signage	Header	none
			HAZ 10 ID Sign	Label	none
			HAZ 10 ID Sign	Checkbox	None
			HAZ 29 Blue EMS	Label	none
			HAZ 29 Blue EMS	Checkbox	None
			HAZ Yellow EMS	Label	none
			HAZ Yellow EMS	Checkbox	None
			Other	label	none
			Site Signage other	Input	none
			Notes	Label	none
			Site Signage Notes	Textbox	none
			Risk Identified	Header	none
			Risk Identified	Textbox	none
			Site Photos	Header	none
			Site Photos upload	button	This control will allow the user to upload a new site photo
			Site Photos Table	Table	Populated with the "Site_Photo_ Name" attribute of each TSS photo from the photos that were added with the "Site Photos upload" Button. A "Remove" button will be next to each Site Photo
			Remove	button	This control will allow the user to remove a photo that they have uploaded.
			Site Drawing	Header	none

			<table><tr><td>Site Drawing upload</td><td>button</td><td>This control will allow the user to upload a photo.</td></tr><tr><td>Remove</td><td>button</td><td>This control will allow the user to remove the photo that they have uploaded.</td></tr><tr><td>Create TSS</td><td>button</td><td>none</td></tr><tr><td>Cancel</td><td>button</td><td>none</td></tr></table>
	Step 2: The operations manager fills in the necessary information into the appropriate input fields requested by the system. The operations manager selects the" Create TSS" button. **[ALT]**		**Step 3**: The system captures and validates the information supplied by the operations manager against the **TSSPHOTO** table using Angular Validation: <table><tr><td>Input Name</td><td>Attribute in table</td><td>Validation requirements</td></tr><tr><td>Site photo Name</td><td>TSS_Photo_Name</td><td>Required</td></tr><tr><td>Photo</td><td>TSS_Photo</td><td>required</td></tr></table> The system captures and validates the information supplied by the operations manager against the **DRAWING** table using Angular Validation: <table><tr><td>Input Name</td><td>Attribute in table</td><td>Validation requirements</td></tr><tr><td>TSS drawing</td><td>File</td><td>required</td></tr></table> The system captures and validates the information supplied by the operations manager against the **TSS** table using Angular Validation: <table><tr><td>Input Name</td><td>Attribute in table</td><td>Validation requirements</td></tr><tr><td>Select Site Type</td><td>Site_Type_ID</td><td>Required</td></tr><tr><td>Survey Date</td><td>Survey_Date</td><td>Required</td></tr><tr><td>Site Name</td><td>Site_Name</td><td>Required</td></tr><tr><td>Bs Number</td><td>Bs_Number</td><td>Required</td></tr><tr><td>Site Description</td><td>Site_Desc</td><td>Required</td></tr><tr><td>Project number</td><td>Project_No</td><td>Required</td></tr></table>

			GPS Coordinates	GPS_ Coordinates	Required
			Distance	Distance	Required
			Time Driving	Time_Driving	Required
			Quotation Date	Quotation_ Date	Required
			Site drawings available	Drawings_ available	Required
			Landlord approval obtained	LLA_obtained	Required
			CCA	CCA	Required
			EIA	EIA	Required
			BP approval obtained	BPA_obtained	Required
			Landlord details	Landlord_ Details	Required
			Special Conditions	Site_ Special_Cond	Required
			Container Type	Container_ Type	Required
			Container Size	Container_ Size	Required
			Container size type	Container_ Size_type	Required
			Special conditions	Container_ Special_Cond	Required
			Container Paint	Container_ Paint	Required
			Load Support	Load_support	Required
			Brick Cladding	Brick_ Cladding	Required
			Rock finish	Rock_finish	Required
			Door Direction	Door_Direction	Required
			Notes	Container_ notes	Required
			Apron	apron	Required
			Plinth Type	Plinth_Type	Required
			Plinth Size	Plinth_Size	Required

			Clear and clean site	CC_site	Required
			Site preparation requirements	Site_prep_ requirements	Required
			Select Mast Type	Mast_Type	Required
			Select Mast height	Mast_height	Required
			Navigation light	Navigation_ light	Required
			Fall arreast	Fall_arreast	Required
			Mast Paint	Mast_Paint	Required
			Environmental Adaptation	Environmental_ Adaptation	Required
			Mast notes	Mast_notes	Required
			RF on Cable Tray	RF_Cable_Tray	Required
			Cable Tray Type	Cable_Tray_ Type	Required
			Length	Cable_Tray_ Length	Required
			Select Bends	Bends	Required
			Bends Qty	Bends_Qty	Required
			Support Pole	Support_Pole	Required
			Support Pole Qty	Support_Pole_ Qty	Required
			Select Mounting Type	Mounting_Type	Required
			Tray Covers	Tray_Covers	Required
			Tray Covers Qty	Tray_Covers_ Qty	Required
			Cable Tray Painted	Cable_Tray_ Painted	Required
			RF in sleeves	RF_sleeves	Required
			RF sleeve Length	RF_sleeve_ Length	Required
			RF sleeve Size	RF_sleeve_ Size	Required

			Select AC Supply	AC_Supply	Required
			Power Available	Power_ Available	Required
			Power Required	Power_ Required	Required
			Generator Required	Generator_ Required	Required
			Select Cable Size	Cable_Size	Required
			Cable Length	Cable_Length	Required
			Select In-Site Cable Route	In-Site_Cable_ Route	Required
			Select In-Site Cable Route Type	In-Site_Cable_ Route_Type	Required
			In-Site Cable Route length	In-Site_Cable_ Route_length	Required
			Consumption Meter Required	Consumption_ Meter_Required	Required
			Site Light Required	Site_Light_ Required	Required
			Select Site Light Position	Site_Light _Position	Required
			Select Earthing Installation	Earthing_ Installation	Required
			Select Earthing Type	Earthing_Type	Required
			Earthing notes	Earthing_notes	Required
			Select Fence Details	Fence_Details	Required
			Select Fence Type	Fence_Type	Required
			Fence Dimensions	Fence_ Dimensions	Required
			Select Surface Finish	Fence_Surface_ Finish	Required
			Fence Colour	Fence_Colour	Required

			Select Fence Additional Extras	Fence_ Additional_ Extras	Required
			Select Gate Type	Gate_Type	Required
			Select Gate opening direction	Gate_opening_ direction	Required
			Gate Dimensions	Gate_ Dimensions	Required
			Locks Required	Locks_Required	Required
			Select Lock Type	Lock_Type	Required
			Select Shackle Length	Shackle_Length	Required
			Lock Chain Required	Lock_Chain_ Required	Required
			Site Security Notes	Site_Security_ Notes	Required
			Clear Road	Clear_Road	Required
			Select Access Road	Access_Road	Required
			Select Road Suitability	Road_Suitability	Required
			Site Access Notes	Site_Access_ Notes	Required
			HAZ 10 ID Sign	HAZ_10_ID_ Sign	Required
			HAZ 29 Blue EMS	HAZ_29_Blue_ EMS	Required
			HAZ Yellow EMS	HAZ_Yellow_ EMS	Required
			Site Signage other	Site_Signage_ other	Required
			Site Signage Notes	Site_Signage_ Notes	Required
			Risk Identified	Risk_Identified	Required
			[ALT]		
			Step 4:		

The system displays an Angular Material Confirmation Modal, with the following controls, to prompt the Operations manager for confirmation of the entered details:

Input Name	Control Type	Notes
Confirm information	Header	none
Please confirm that the details entered are correct?	Label	none
Yes	button	This button confirms that the entered details were correct
No	button	This button confirms that the entered details were not correct

Step 6:
The operations manager confirms the captured information by clicking the "Yes" button.
[ALT]

Step 7:
The system uses a SQL insert query to save the captured information in the **TSSPHOTO** table as follows:

Details	Attribute in table	Notes
TSS photo ID	TSS_photo_ID	ID is generated by the system by using the last ID in the table and incrementing it by one.
TSS ID	TSS_ID	none
Site photo Name	TSS_Photo_Name	none
Photo	TSS_Photo	none

The system uses a SQL insert query to save the captured information in the **DRAWING** table as follows:

Details	Attribute in table	Notes
Drawing ID	Drawing_ID	ID is generated by the system by using the last ID in the table and incrementing it by one.
Name	Name	The name is set to the name of the project
Description	Description	The description is set to the name of the file that was uploaded
Created Date	CreatedDate	The date is set to the date the TSS was created.
TSS drawing	File	none
Project ID	ProjectID	none
Drawing Type	DrawingTypeID	The drawing type is set to rough drawing

The system uses a SQL insert query to save the captured information in the **TSS** table as follows:

Details	Attribute in table	Notes
TSS_ID	TSS_ID	ID is generated by the system by using the last ID in the table and incrementing it by one.
Project ID	Project_ID	none

			Site Type	Site_Type_ID	none
			Survey Date	Survey_Date	none
			Site Name	Site_Name	none
			Bs Number	Bs_Number	none
			Site Description	Site_Desc	none
			Project number	Project_No	none
			GPS Coordinates	GPS_ Coordinates	none
			Distance	Distance	none
			Time Driving	Time_Driving	none
			Quotation Date	Quotation_ Date	none
			Site drawings available	Drawings_ available	none
			Landlord approval obtained	LLA_obtained	none
			CCA	CCA	none
			EIA	EIA	none
			BP approval obtained	BPA_obtained	none
			Landlord details	Landlord_ Details	none
			Special Conditions	Site_ Special_Cond	none
			Container Type	Container_ Type	none
			Container Size	Container_ Size	none
			Container size type	Container_ Size_type	none
			Special conditions	Container_ Special_Cond	none
			Container Paint	Container_ Paint	none
			Load Support	Load_support	none

			Brick Cladding	Brick_ Cladding	none
			Rock finish	Rock_finish	none
			Door Direction	Door_Direction	none
			Notes	Container_ notes	none
			Apron	apron	none
			Plinth Type	Plinth_Type	none
			Plinth Size	Plinth_Size	none
			Clear and clean site	CC_site	none
			Site preparation requirements	Site_prep_ requirements	none
			Select Mast Type	Mast_Type	none
			Select Mast height	Mast_height	none
			Navigation light	Navigation_ light	none
			Fall arreast	Fall_arreast	none
			Mast Paint	Mast_Paint	none
			Environmental Adaptation	Environmental_ Adaptation	none
			Mast notes	Mast_notes	none
			RF on Cable Tray	RF_Cable_Tray	none
			Cable Tray Type	Cable_Tray_ Type	none
			Length	Cable_Tray_ Length	none
			Bends	Bends	none
			Bends Qty	Bends_Qty	none
			Support Pole	Support_Pole	none
			Support Pole Qty	Support_Pole_ Qty	none
			Mounting Type	Mounting_Type	none
			Tray Covers	Tray_Covers	none

			Tray Covers Qty	Tray_Covers_ Qty	none
			Cable Tray Painted	Cable_Tray_ Painted	none
			RF in sleeves	RF_sleeves	none
			RF sleeve Length	RF_sleeve_ Length	none
			RF sleeve Size	RF_sleeve_ Size	none
			Select AC Supply	AC_Supply	none
			Power Available	Power_ Available	none
			Power Required	Power_ Required	none
			Generator Required	Generator_ Required	none
			Select Cable Size	Cable_Size	none
			Cable Length	Cable_Length	none
			Select In-Site Cable Route	In-Site_Cable_ Route	none
			Select In-Site Cable Route Type	In-Site_Cable_ Route_Type	none
			In-Site Cable Route length	In-Site_Cable_ Route_length	none
			Consumption Meter Required	Consumption_ Meter_Required	none
			Site Light Required	Site_Light_ Required	none
			Select Site Light Position	Site_Light _Position	none
			Select Earthing Installation	Earthing_ Installation	none
			Select Earthing Type	Earthing_Type	none
			Earthing notes	Earthing_notes	none
			Select Fence Details	Fence_Details	none

			Select Fence Type	Fence_Type	none
			Fence Dimensions	Fence_ Dimensions	none
			Surface Finish	Fence_Surface_ Finish	none
			Fence Colour	Fence_Colour	none
			Select Fence Additional Extras	Fence_ Additional_ Extras	none
			Gate Type	Gate_Type	none
			Select Gate opening direction	Gate_opening_ direction	none
			Gate Dimensions	Gate_ Dimensions	none
			Locks Required	Locks_Required	none
			Select Lock Type	Lock_Type	none
			Shackle Length	Shackle_Length	none
			Lock Chain Required	Lock_Chain_ Required	none
			Site Security Notes	Site_Security_ Notes	none
			Clear Road	Clear_Road	none
			Select Access Road	Access_Road	none
			Select Road Suitability	Road_Suitability	none
			Site Access Notes	Site_Access_ Notes	none
			HAZ 10 ID Sign	HAZ_10_ID_ Sign	none
			HAZ 29 Blue EMS	HAZ_29_Blue_ EMS	none
			HAZ Yellow EMS	HAZ_Yellow_ EMS	none
			Site Signage other	Site_Signage_ other	none

			<table><tr><td>Site Signage Notes</td><td>Site_Signage_ Notes</td><td>none</td></tr><tr><td>Risk Identified</td><td>Risk_Identified</td><td>none</td></tr></table> The system uses a SQL insert query to save the captured information in the **PROJECT LINE-ITEM** table. For each line Item in the TSS the line items will be stored as follows: <table><tr><th>Details</th><th>Attribute in table</th><th>Notes</th></tr><tr><td>Projects ID</td><td>TSS_photo_ID</td><td>none</td></tr><tr><td>Line-Item ID</td><td>Line-Item_ID</td><td>none</td></tr><tr><td>TSS ID</td><td>TSS_ID</td><td>none</td></tr></table>
			Step 8: The system notifies operations manager that the TSS was created successfully, using an Angular Material Success Notification Modal with the following controls: <table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Success</td><td>Header</td><td>none</td></tr><tr><td>The TSS was successfully created</td><td>Label</td><td>none</td></tr><tr><td>OK</td><td>button</td><td>None</td></tr></table>
	Step 9: The operations manager clicks the “Ok” button.		
ALTERNATE COURSES:	**ALT Step 3:** The Operations manager selects the “Cancel” button. The use case is terminated.		

ALT Step 4:

- The captured details are not valid. The system displays a notification indicating that the information received is incorrect or incomplete with an Angular Material Invalid Notification Modal as follows:

Input Name	Control Type	Notes
Information Invalid	Header	none
The entered details were incorrect. Please fill in the correct information.	Label	none
OK	button	None

The system returns to step 1.

ALT Step 6:

The Operations manager does not confirm that the information captured is correct by clicking the “No” button.

The system returns to step 1.

ALT Step 7:

The system does not successfully add the TSS information to the TSS, TSS PHOTO table, DRAWING table and PROJECT LINE-ITEM table. The system notifies the user that the TSS has not been added successfully with an Angular Material Error Modal as follows:

Input Name	Control Type	Notes
Error	Header	none
An error has occurred while saving the TSS details. Please try again.	Label	none
OK	button	None

The system terminates this use case.

CONCLUSION:

This use case concludes when the operations manager has completed the TSS form successfully.

POST-CONDITION:

- The technical site survey has been added to the TSS table, the required materials are added to the PROJECT LINE ITEMS table, the drawing was added to the DRAWINGS table and the site photos were added to the TSS PHOTO table.

BUSINESS RULES:	• Only the operations manager is allowed to create a new TSS.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	The operations manager will always have a smart device with him when visiting a site to make a new TSS.
OPEN ISSUES:	none

1.2.1.2 Edit TSS

Gohvan Construction			
Author (s): Amore Rossouw	**Date: 28/06/2021**		**Version: 1.0**
USE CASE NAME:	**Edit TSS**		**USE CASE TYPE**
USE CASE ID:	**1.2**		**Business Requirements:** ☐
PRIORITY:	High		**System Analysis:** ☐
SOURCE:	Gohvan Construction System		**System Design:** ☑
PRIMARY BUSINESS ACTOR:	Operations Manager (PSA)		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	• None		
OTHER INTERESTED STAKEHOLDERS:	• None		
DESCRIPTION:	This use case narrative describes the process the operations manager will follow when they want to edit an existing technical site survey.		
PRE-CONDITION:	• The user must be logged into the system. • The technical site survey that the user wishes to update must exist in the database.		
TRIGGER:	The operations manager wants to update an existing technical site survey.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The operations manager selects the project they want to update		**Step 2:** The system invokes use case “4.9 Search Project” and displays the “Project” page.
	Step 3:		**Step 4:**

	The operations manager clicks on the "Technical Site Survey" Card.		The system reads information about the selected TSS from the following Tables: • TSS table • TSS photo table • Drawing table • Project Table • Project Line-Item Table The system responds by loading the "Edit TSS" screen with the following controls:

Input Name	Control Type	Notes
Technical Site Survey	**Header**	**none**
Select Site Type	Label	none
Select Site Type	Select drop-down list	Populated with the Description attribute of each Site Type read. The value of each item in the drop down is set to the corresponding SiteTypeID.
Survey Date	label	none
Date	Date picker	This control is prepopulated with the current date.
Site Details	**Header**	**none**
Site Name	Label	none
Site Name	Input	none
Bs Number	Label	none
Bs Number	Input	This is the input for the base station number
Site Description	Label	none
Site Description	Text box	none
Project number	Label	none
Project Number	Input	none
GPS Coordinates	label	none

			GPS Map	Google maps	The system will use the geolocation function to get the current coordinates of the user
			Get Coordinates	button	none
			GPS Coordinates	Input	If the user does not want to user, the geolocation function they can manually type in the coordinates.
			Distance	Label	none
			Distance	Input	none
			Time Driving	Label	none
			Time Driving	Input	none
			Quotation Date	Label	none
			Date	Date picker	none
			Pre-Approval	**Header**	**none**
			Site drawings available	Label	none
			Site drawings available	Checkbox	none
			Landlord approval obtained	Label	none
			Landlord approval obtained	Checkbox	none
			CCA	Label	none
			CCA	Checkbox	none
			EIA	Label	none
			EIA	Checkbox	none
			BP approval obtained	Label	none
			BP approval obtained	Checkbox	none

			Landlord details	label	none
			Landlord details	Text box	none
			Special Conditions	label	none
			Special Conditions	Text box	none
			Accommodation	**Header**	**none**
			Container	Header	none
			Container Type	Label	none
			Container Type	Select drop-down list	Populated with the options “ARP 1 Door H&S Security Cab”, “ARP 2 Door H&S Security Cab”, “Mc Wade SV140” and “2x RED 1's”
			Container Other	Label	none
			Container Other	input	none
			Container Size	Label	none
			Container Size	Input	none
			Container size type	Label	none
			Container size type	Select drop-down list	Populated with the options “standard” and “other”.
			Special conditions	Label	none
			Container Paint	Label	none
			Container Paint	Checkbox	None
			Load Support	Label	none
			Load support	Checkbox	None
			Brick Cladding	Label	none
			Brick cladding	Checkbox	None
			Rock finish	Label	none
			Rock finish	Checkbox	None
			Door Direction	Label	none

			Door Direction	Select drop-down list	Populated with the options “North”, “East”, “South” and “West”.
			Notes	label	none
			Notes	Textbox	none
			Plinth	Header	none
			Apron	label	none
			Apron	Checkbox	none
			Plinth Type	Label	none
			Plinth Type	Select drop-down list	Populated with the options “new”, “flat on apron” and “blocks”
			Plinth Size	Label	none
			Plinth Size	Input	none
			Site Preparation details	**Header**	**none**
			Groundwork	Header	none
			Clear and clean site	Label	none
			Clear and clean site	Checkbox	None
			Other requirements	Label	none
			Site preparation requirements	Textbox	none
			Mast Details	**Header**	**none**
			Mast	Header	none
			Select Mast Type	Label	none
			Select Mast Type	Select drop-down list	Populated with the options: “Lattice”, “Monopole” and “Monalatt”.
			Select Mast height	Label	none
			Select Mast height	Select drop-down list	Populated with the options: “25”, “35” and “other”.

			Other	Label	none
			Mast height	Input	This control will only be enabled once the user selects 'other' from the "Select Mast height" control.
			Mast Accessories	Header	none
			Navigation light	Label	none
			Navigation light	Checkbox	None
			Fall arreast	Label	none
			Fall arreast	Checkbox	None
			Mast Paint	Label	none
			Mast Paint	Checkbox	None
			Environmental Adaptation	Label	
			Environmental Adaptation	Select drop-down list	Populated with the options: "Tree", "Palm tree" and 'other".
			Other	Label	none
			Environmental Adaptation other	Input	This control will only be enabled once the user selects 'other' from the "Environ mental Adaptation" control.
			Mast notes	Label	none
			Mast notes	Textbox	none
			RF Installation Details	**Header**	**none**
			Cable Management	Header	none
			RF on cable Tray	Header	none
			RF on Cable Tray	Label	none
			RF on Cable Tray	Checkbox	None
			Cable Tray Type	Label	none

			Cable Tray Type	Select drop-down list	Populated with the options: “114”, “228”, ‘304” and “other”.
			Other	Label	none
			Cable Tray Type other	Input	This control will only be enabled once the user selects ‘other’ from the “Cable Tray Type” control
			Length	label	none
			Length	Input	none
			Bends	Label	none
			Select Bends	Select drop-down list	Populated with the options: “90” and “other”.
			Other	Label	none
			Bends other	Input	This control will only be enabled once the user selects ‘other’ from the “select Bends” control
			Quantity	Label	none
			Bends Qty	Input	none
			Support Pole	Label	none
			Support Pole	Checkbox	None
			Quantity	Label	none
			Support Pole Qty	Input	This control will only be enabled once the user selects yes from the Support pole checkbox.
			Mounting Type	Label	none
			Select Mounting Type	Select drop-down list	Populated with the options: “Cantelvr Arms”, “Concrete

					Arms", "Concrete Blks" and "other".
			Other	Label	none
			Mounting Type other	Input	This control will only be enabled once the user selects 'other' from the "select Mounting Type" control
			Tray Covers	Label	none
			Tray Covers	Checkbox	None
			Quantity	Label	none
			Tray Covers Qty	Input	This control will only be enabled once the user selects yes from the Tray Covers checkbox.
			Cable Tray Painted	label	none
			Cable Tray Painted	Checkbox	None
			RF sleeves	Header	none
			RF in sleeves	label	none
			RF in sleeves	Checkbox	None
			Length	Label	none
			RF sleeve Length	Input	This control will only be enabled once the user selects yes from the RF in sleeves checkbox.
			Size	label	none
			RF sleeve Size	Input	This control will only be enabled once the user selects yes from the RF in sleeves checkbox.

			Electrical Details	**Header**	**none**
			AC Supply	Label	none
			Select AC Supply	Select drop-down list	Populated with the options: “Site DB”, “Site Container” and “other”.
			Other	Label	none
			AC Supply other	Input	This control will only be enabled once the user selects ‘other’ from the “select AC Supply” control
			Power Available	Label	none
			Power Available	Select drop-down list	Populated with the options: “SP” and “3P”.
			Power Required	Label	none
			Power Required	Select drop-down list	Populated with the options: “SP” and “3P”.
			Generator Required	label	none
			Generator Required	Checkbox	None
			Cable Size	Label	none
			Select Cable Size	Select drop-down list	Populated with the options: “16mm³”, “25mm³”, “30mm³” and “other”.
			Other	Label	none
			Cable Size other	Input	This control will only be enabled once the user selects ‘other’ from the “select Cable Size” control
			Cable Length	Label	none
			Cable Length	Input	none

			In-Site Cable Route	Label	none
			Select In-Site Cable Route	Select drop-down list	Populated with the options: “Bosal” and “Sleeves”
			In-Site Cable Route Type	Label	none
			Select In-Site Cable Route Type	Select drop-down list	Populated with the options: “Conduit”, “Cable Tray” and “other”.
			Other	Label	none
			In-Site Cable Route Type other	Input	This control will only be enabled once the user selects ‘other’ from the “Select In-Site Cable Route Type” control
			In-Site Cable Route length	Label	none
			In-Site Cable Route length	Input	none
			Consumption Meter Required	Label	none
			Consumption Meter Required	Checkbox	None
			Site Light Required	Label	none
			Site Light Required	Checkbox	None
			Site Light Position	Label	none
			Select Site Light Position	Select drop-down list	Populated with the options: “North”,” East”, “South “and “West”.
			Earthing details	Header	none

			Earthing Installation	Label	none
			Select Earthing Installation	Select drop-down list	Populated with the options: “New”, and” Existing”
			Earthing Type	Label	none
			Select Earthing Type	Select drop-down list	Populated with the options: “Green and yellow”, “Black” “Bare Copper” and” Other”
			Other	Label	none
			Earthing Type other	Input	This control will only be enabled once the user selects ‘other’ from the “Select Earthing Type” control
			Notes	Label	none
			Earthing notes	Textbox	none
			Construction Details	**Header**	**none**
			Fencing	Header	none
			Fence Details	Label	none
			Select Fence Details	Select drop-down list	Populated with the options: “In-ground”, and “Above-ground”
			Fence Type	Label	none
			Select Fence Type	Select drop-down list	Populated with the options: “Palisade”, “Round table” and” Other”
			Other	Label	none
			Fence Type other	Input	This control will only be enabled once the user selects

					'other' from the "Select Fence Type" control
			Fence Dimensions	Label	none
			Fence Dimensions	Input	none
			Surface Finish	Label	none
			Select Surface Finish	Select drop-down list	Populated with the options: "Semi-Galv" and "Paint"
			Colour	label	none
			Fence Colour	Input	none
			Additional Extras	Label	none
			Select Fence Additional Extras	Select drop-down list	Populated with the options: "Razor Coil", "Flat wrap" and "other"
			Other	Label	none
			Fence Additional Extras other	Input	This control will only be enabled once the user selects 'other' from the "Select Fence Additional Extras" control
			Gate Type	Label	none
			Select Gate Type	Select drop-down list	Populated with the options: "Swing" and "Slide"
			Gate opening direction	Label	none
			Select Gate opening direction	Select drop-down list	Populated with the options: "Inward" and "Outward"
			Gate Dimensions	Label	none
			Gate Dimensions	Input	none
			Site Security	Header	none
			Locks Required	Label	none
			Locks Required	Checkbox	None

			Lock Type	Label	none
			Select Lock Type	Select drop-down list	Populated with the options: “VCL”, “VM”, “NM” and “ARN”
			Shackle Length	Label	none
			Select Shackle Length	Select drop-down list	Populated with the options: “24mm” and “50mm”
			Lock Chain Required	Label	none
			Lock Chain Required	Checkbox	None
			Notes	Label	none
			Site Security Notes	Textbox	none
			Site Access	Header	none
			Clear Road	Label	none
			Clear Road	Checkbox	None
			Access Road	Label	none
			Select Access Road	Select drop-down list	Populated with the options: “Existing tar”, “Existing gravel” and “other”
			Other	Label	none
			Access Road other	Input	This control will only be enabled once the user selects ‘other’ from the “Select Access Road” control
			Road Suitability	Label	none
			Select Road Suitability	Select drop-down list	Populated with the options: “Car”, “4x4” and “other”
			Other	Label	none
			Road Suitability other	Input	This control will only be enabled once the user selects

					'other' from the "Select Road Suitability" control
			Notes	Label	none
			Site Access Notes	Textbox	none
			Site Signage	Header	none
			HAZ 10 ID Sign	Label	none
			HAZ 10 ID Sign	Checkbox	None
			HAZ 29 Blue EMS	Label	none
			HAZ 29 Blue EMS	Checkbox	None
			HAZ Yellow EMS	Label	none
			HAZ Yellow EMS	Checkbox	None
			Other	label	none
			Site Signage other	Input	none
			Notes	Label	none
			Site Signage Notes	Textbox	none
			Risk Identified	Header	none
			Risk Identified	Textbox	none
			Site Photos	Header	none
			Site Photos upload	button	This control will allow the user to upload a new site photo
			Site Photos Table	Table	Populated with the "Site_Photo_ Name" attribute of each TSS photo from the photos that were added with the "Site Photos upload" Button. A "Remove" button will be next to each Site Photo

<table>
<tr>
<td></td>
<td></td>
<td></td>
<td>
<table>
<tr><td>Remove</td><td>button</td><td>This control will allow the user to remove a photo that they have uploaded.</td></tr>
<tr><td>Site Drawing</td><td>Header</td><td>none</td></tr>
<tr><td>Site Drawing upload</td><td>button</td><td>This control will allow the user to upload a photo.</td></tr>
<tr><td>Remove</td><td>button</td><td>This control will allow the user to remove the photo that they have uploaded.</td></tr>
<tr><td>Save TSS</td><td>button</td><td>none</td></tr>
<tr><td>Submit TSS</td><td>button</td><td>none</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 5:
The Operations Manager fills in the necessary information into the appropriate input fields requested by the system.
The Operations Manager selects the “Save TSS” button.
[ALT]</td>
<td></td>
<td>
Step 6:
The system captures and validates the information supplied by the operations manager against the TSSPHOTO table using Angular Validation:
<table>
<tr><th>Input Name</th><th>Attribute in table</th><th>Validation requirements</th></tr>
<tr><td>Site photo Name</td><td>TSS_Photo_Name</td><td>Required</td></tr>
<tr><td>Photo</td><td>TSS_Photo</td><td>required</td></tr>
</table>
The system captures and validates the information supplied by the operations manager against the DRAWING table using Angular Validation:
<table>
<tr><th>Input Name</th><th>Attribute in table</th><th>Validation requirements</th></tr>
<tr><td>TSS drawing</td><td>File</td><td>required</td></tr>
</table>
The system captures and validates the information supplied by the operations manager against the TSS table using Angular Validation:
<table>
<tr><th>Input Name</th><th>Attribute in table</th><th>Validation requirements</th></tr>
<tr><td>Select Site Type</td><td>Site_Type_ID</td><td>Required</td></tr>
<tr><td>Survey Date</td><td>Survey_Date</td><td>Required</td></tr>
<tr><td>Site Name</td><td>Site_Name</td><td>Required</td></tr>
</table>
</td>
</tr>
</table>

			Bs Number	Bs_Number	Required
			Site Description	Site_Desc	Required
			Project number	Project_No	Required
			GPS Coordinates	GPS_ Coordinates	Required
			Distance	Distance	Required
			Time Driving	Time_Driving	Required
			Quotation Date	Quotation_ Date	Required
			Site drawings available	Drawings_ available	Required
			Landlord approval obtained	LLA_obtained	Required
			CCA	CCA	Required
			EIA	EIA	Required
			BP approval obtained	BPA_obtained	Required
			Landlord details	Landlord_ Details	Required
			Special Conditions	Site_ Special_Cond	Required
			Container Type	Container_ Type	Required
			Container Size	Container_ Size	Required
			Container size type	Container_ Size_type	Required
			Special conditions	Container_ Special_Cond	Required
			Container Paint	Container_ Paint	Required
			Load Support	Load_support	Required
			Brick Cladding	Brick_ Cladding	Required
			Rock finish	Rock_finish	Required
			Door Direction	Door_Direction	Required

			Notes	Container_ notes	Required
			Apron	apron	Required
			Plinth Type	Plinth_Type	Required
			Plinth Size	Plinth_Size	Required
			Clear and clean site	CC_site	Required
			Site preparation requirements	Site_prep_ requirements	Required
			Select Mast Type	Mast_Type	Required
			Select Mast height	Mast_height	Required
			Navigation light	Navigation_ light	Required
			Fall arreast	Fall_arreast	Required
			Mast Paint	Mast_Paint	Required
			Environmental Adaptation	Environmental_ Adaptation	Required
			Mast notes	Mast_notes	Required
			RF on Cable Tray	RF_Cable_Tray	Required
			Cable Tray Type	Cable_Tray_ Type	Required
			Length	Cable_Tray_ Length	Required
			Select Bends	Bends	Required
			Bends Qty	Bends_Qty	Required
			Support Pole	Support_Pole	Required
			Support Pole Qty	Support_Pole_ Qty	Required
			Select Mounting Type	Mounting_Type	Required
			Tray Covers	Tray_Covers	Required
			Tray Covers Qty	Tray_Covers_ Qty	Required
			Cable Tray Painted	Cable_Tray_ Painted	Required

			RF in sleeves	RF_sleeves	Required
			RF sleeve Length	RF_sleeve_ Length	Required
			RF sleeve Size	RF_sleeve_ Size	Required
			Select AC Supply	AC_Supply	Required
			Power Available	Power_ Available	Required
			Power Required	Power_ Required	Required
			Generator Required	Generator_ Required	Required
			Select Cable Size	Cable_Size	Required
			Cable Length	Cable_Length	Required
			Select In-Site Cable Route	In-Site_Cable_ Route	Required
			Select In-Site Cable Route Type	In-Site_Cable_ Route_Type	Required
			In-Site Cable Route length	In-Site_Cable_ Route_length	Required
			Consumption Meter Required	Consumption_ Meter_Required	Required
			Site Light Required	Site_Light_ Required	Required
			Select Site Light Position	Site_Light _Position	Required
			Select Earthing Installation	Earthing_ Installation	Required
			Select Earthing Type	Earthing_Type	Required
			Earthing notes	Earthing_notes	Required
			Select Fence Details	Fence_Details	Required
			Select Fence Type	Fence_Type	Required
			Fence Dimensions	Fence_ Dimensions	Required

			Select Surface Finish	Fence_Surface_ Finish	Required
			Fence Colour	Fence_Colour	Required
			Select Fence Additional Extras	Fence_ Additional_ Extras	Required
			Select Gate Type	Gate_Type	Required
			Select Gate opening direction	Gate_opening_ direction	Required
			Gate Dimensions	Gate_ Dimensions	Required
			Locks Required	Locks_Required	Required
			Select Lock Type	Lock_Type	Required
			Select Shackle Length	Shackle_Length	Required
			Lock Chain Required	Lock_Chain_ Required	Required
			Site Security Notes	Site_Security_ Notes	Required
			Clear Road	Clear_Road	Required
			Select Access Road	Access_Road	Required
			Select Road Suitability	Road_Suitability	Required
			Site Access Notes	Site_Access_ Notes	Required
			HAZ 10 ID Sign	HAZ_10_ID_ Sign	Required
			HAZ 29 Blue EMS	HAZ_29_Blue_ EMS	Required
			HAZ Yellow EMS	HAZ_Yellow_ EMS	Required
			Site Signage other	Site_Signage_ other	Required
			Site Signage Notes	Site_Signage_ Notes	Required
			Risk Identified	Risk_Identified	Required

<table>
<tr><td></td><td></td><td></td><td>[ALT]</td></tr>
<tr><td></td><td></td><td>.</td><td>Step 5:
The system displays an Angular Material Confirmation Modal, with the following controls, to prompt the Operations Manager for confirmation of the entered details:

<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Confirm information</td><td>Header</td><td>none</td></tr>
<tr><td>Please confirm that the details entered are correct?</td><td>Label</td><td>none</td></tr>
<tr><td>Yes</td><td>button</td><td>This button confirms that the entered details were correct</td></tr>
<tr><td>No</td><td>button</td><td>This button confirms that the entered details were not correct</td></tr>
</table></td></tr>
<tr><td></td><td>Step 7:
The Operations manager confirms the captured information by clicking the “Yes” button.
[ALT]</td><td></td><td>Step 8:
The system uses a SQL update query to save the captured information in the TSSPHOTO table as follows:

<table>
<tr><th>Details</th><th>Attribute in table</th><th>Notes</th></tr>
<tr><td>TSS photo ID</td><td>TSS_photo_ID</td><td>ID is generated by the system by using the last ID in the table and incrementing it by one.</td></tr>
</table></td></tr>
</table>

TSS ID	TSS_ID	none
Site photo Name	TSS_Photo_Name	none
Photo	TSS_Photo	none

The system uses a SQL update query to save the captured information in the **DRAWING** table as follows:

Details	Attribute in table	Notes
Drawing ID	Drawing_ID	ID is generated by the system by using the last ID in the table and incrementing it by one.
Name	Name	The name is set to the name of the project
Description	Description	The description is set to the name of the file that was uploaded
Created Date	CreatedDate	The date is set to the date the TSS was created.
TSS drawing	File	none
Project ID	ProjectID	none
Drawing Type	DrawingTypeID	The drawing type is set to rough drawing

The system uses a SQL update query to save the captured information in the **TSS** table as follows:

Details	Attribute in table	Notes
TSS_ID	TSS_ID	ID is generated

					by the system by using the last ID in the table and incrementing it by one.
			Project ID	Project_ID	none
			Site Type	Site_Type_ID	none
			Survey Date	Survey_Date	none
			Site Name	Site_Name	none
			Bs Number	Bs_Number	none
			Site Description	Site_Desc	none
			Project number	Project_No	none
			GPS Coordinates	GPS_ Coordinates	none
			Distance	Distance	none
			Time Driving	Time_Driving	none
			Quotation Date	Quotation_ Date	none
			Site drawings available	Drawings_ available	none
			Landlord approval obtained	LLA_obtained	none
			CCA	CCA	none
			EIA	EIA	none
			BP approval obtained	BPA_obtained	none
			Landlord details	Landlord_ Details	none
			Special Conditions	Site_ Special_Cond	none
			Container Type	Container_ Type	none
			Container Size	Container_ Size	none
			Container size type	Container_ Size_type	none

			Special conditions	Container_ Special_Cond	none
			Container Paint	Container_ Paint	none
			Load Support	Load_support	none
			Brick Cladding	Brick_ Cladding	none
			Rock finish	Rock_finish	none
			Door Direction	Door_Direction	none
			Notes	Container_ notes	none
			Apron	apron	none
			Plinth Type	Plinth_Type	none
			Plinth Size	Plinth_Size	none
			Clear and clean site	CC_site	none
			Site preparation requirements	Site_prep_ requirements	none
			Select Mast Type	Mast_Type	none
			Select Mast height	Mast_height	none
			Navigation light	Navigation_ light	none
			Fall arreast	Fall_arreast	none
			Mast Paint	Mast_Paint	none
			Environmental Adaptation	Environmental_ Adaptation	none
			Mast notes	Mast_notes	none
			RF on Cable Tray	RF_Cable_Tray	none
			Cable Tray Type	Cable_Tray_ Type	none
			Length	Cable_Tray_ Length	none
			Bends	Bends	none
			Bends Qty	Bends_Qty	none
			Support Pole	Support_Pole	none

			Support Pole Qty	Support_Pole_ Qty	none
			Mounting Type	Mounting_Type	none
			Tray Covers	Tray_Covers	none
			Tray Covers Qty	Tray_Covers_ Qty	none
			Cable Tray Painted	Cable_Tray_ Painted	none
			RF in sleeves	RF_sleeves	none
			RF sleeve Length	RF_sleeve_ Length	none
			RF sleeve Size	RF_sleeve_ Size	none
			Select AC Supply	AC_Supply	none
			Power Available	Power_ Available	none
			Power Required	Power_ Required	none
			Generator Required	Generator_ Required	none
			Select Cable Size	Cable_Size	none
			Cable Length	Cable_Length	none
			Select In-Site Cable Route	In-Site_Cable_ Route	none
			Select In-Site Cable Route Type	In-Site_Cable_ Route_Type	none
			In-Site Cable Route length	In-Site_Cable_ Route_length	none
			Consumption Meter Required	Consumption_ Meter_Required	none
			Site Light Required	Site_Light_ Required	none
			Select Site Light Position	Site_Light _Position	none
			Select Earthing Installation	Earthing_ Installation	none

			Select Earthing Type	Earthing_Type	none
			Earthing notes	Earthing_notes	none
			Select Fence Details	Fence_Details	none
			Select Fence Type	Fence_Type	none
			Fence Dimensions	Fence_ Dimensions	none
			Surface Finish	Fence_Surface_ Finish	none
			Fence Colour	Fence_Colour	none
			Select Fence Additional Extras	Fence_ Additional_ Extras	none
			Gate Type	Gate_Type	none
			Select Gate opening direction	Gate_opening_ direction	none
			Gate Dimensions	Gate_ Dimensions	none
			Locks Required	Locks_Required	none
			Select Lock Type	Lock_Type	none
			Shackle Length	Shackle_Length	none
			Lock Chain Required	Lock_Chain_ Required	none
			Site Security Notes	Site_Security_ Notes	none
			Clear Road	Clear_Road	none
			Select Access Road	Access_Road	none
			Select Road Suitability	Road_Suitability	none
			Site Access Notes	Site_Access_ Notes	none
			HAZ 10 ID Sign	HAZ_10_ID_ Sign	none

			HAZ 29 Blue EMS | HAZ_29_Blue_EMS | none HAZ Yellow EMS | HAZ_Yellow_EMS | none Site Signage other | Site_Signage_other | none Site Signage Notes | Site_Signage_Notes | none Risk Identified | Risk_Identified | none
			Step 9: The system notifies Operations Manager that the TSS was updated successfully, using an Angular Material Success Notification Modal with the following controls: Input Name | Control Type | Notes Success | Header | none The TSS was successfully updated | Label | none OK | button | None
	Step 10: The Operations manager clicks the "Ok" button.		
ALTERNATE COURSES:	**ALT Step 5A:** The Operations Manager selects the "Cancel" button. The use case is terminated.		
	ALT Step 5B: The Operations Manager clicks on the "Submit TSS" button. The System invokes use case "1.3 Submit TSS".		
	ALT Step 6: • The captured details are not valid. The system displays a notification indicating that the information received is incorrect or incomplete with an Angular Material Invalid Notification Modal as follows: Input Name | Control Type | Notes Information Invalid | Header | none		

<table>
<tr><td></td><td>
<table>
<tr><td>The entered details were incorrect. Please fill in the correct information.</td><td>Label</td><td>none</td></tr>
<tr><td>OK</td><td>button</td><td>None</td></tr>
</table>

The system returns to step 1.</td></tr>
<tr><td></td><td>ALT Step 7:
The Operations Manager does not confirm that the information captured is correct by clicking the “No” button.
The system returns to step 1.</td></tr>
<tr><td></td><td>ALT Step 8:
The system does not successfully update the TSS information to the related tables. The system notifies the user that the TSS has not been updated successfully with an Angular Material Error Modal as follows:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Error</td><td>Header</td><td>none</td></tr>
<tr><td>An error has occurred while saving the TSS details. Please try again.</td><td>Label</td><td>none</td></tr>
<tr><td>OK</td><td>button</td><td>None</td></tr>
</table>
The system terminates this use case</td></tr>
<tr><td>CONCLUSION:</td><td>This use concludes when the operations manager has successfully updated the technical site survey’s details.</td></tr>
<tr><td>POST-CONDITION:</td><td>• The technical site survey has been updated to the TSS table, the required materials are updated to the PROJECT LINE ITEMS table, the drawing was updated to the DRAWINGS table and the site photos were updated to the TSS PHOTO table.</td></tr>
<tr><td>BUSINESS RULES:</td><td>• Only the operations manager is allowed to edit the technical site survey.</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td>none</td></tr>
<tr><td>ASSUMPTIONS:</td><td>none</td></tr>
<tr><td>OPEN ISSUES:</td><td>none</td></tr>
</table>

1.2.1.3 Submit TSS Report

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Submit TSS</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">1.3</td><td>Abstract : ☐</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td><td>Extension : ☑</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td><td></td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">The Operations Manager (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case narrative describes the process the operations manager will follow when they want to submit the finished technical site survey (TSS) report. Once the technical site survey is complete the operations manager will request the system to create a TSS report which will be sent to the operations manager via email.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The operations manager must be logged into the system.
• The technical site survey must be complete.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The operations manager wants to submit the technical site survey report to the client.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td></td><td>Step 1:
The system responds by loading an Angular Material Confirmation Modal, with the following controls, to prompt the Operations Manager to confirm the TSS details:

<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
</table>
</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>Confirm information</td><td>Header</td><td>none</td></tr>
<tr><td>Please confirm that the details entered are correct?</td><td>Label</td><td>none</td></tr>
<tr><td>Yes</td><td>button</td><td>This button confirms that the entered details were correct</td></tr>
<tr><td>No</td><td>button</td><td>This button confirms that the entered details were not correct</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 2:
The Operations manager confirms the TSS information by clicking the “Yes” button. [ALT]</td><td></td><td>Step 4:
The system will change the status in the PROJECT STATUS table to “TSS submitted”.</td></tr>
<tr><td></td><td></td><td></td><td>Step 5:
The system notifies Operations Manager that the TSS was Submitted successfully, using an Angular Material Success Notification Modal with the following controls:
[ALT]
<table>
<tr><td>Input Name</td><td>Control Type</td><td>Notes</td></tr>
<tr><td>Success</td><td>Header</td><td>none</td></tr>
<tr><td>The TSS was successfully updated</td><td>Label</td><td>none</td></tr>
<tr><td>OK</td><td>button</td><td>None</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 6:</td><td></td><td></td></tr>
</table>

<table>
<tr><td></td><td>The Operations manager clicks the “Ok” button.</td><td></td><td></td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 2:
The Operations Manager does not confirm that the information captured is correct by clicking the “No” button.
The system terminates this use case.</td></tr>
<tr><td></td><td colspan="3">ALT Step 5:
The system does not successfully submit the TSS information. The system notifies the user that the TSS has not been submitted successfully with an Angular Material Error Modal as follows:

<table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Error</td><td>Header</td><td>none</td></tr><tr><td>An error has occurred while submitting the TSS details. Please try again.</td><td>Label</td><td>none</td></tr><tr><td>OK</td><td>button</td><td>None</td></tr></table>The system terminates this use case</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">This use concludes when the operations manager has successfully submitted the TSS report.</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">The technical site survey report has been compiled and submitted.
The operations manager has received the technical site survey report.</td></tr>
</table>

1.2.2. Quotation Subsystem

1.2.2.1 Create Quote

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Tshepang Mashao</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Create Quote</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">2.1</td><td rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Client</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">Operations Manager</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case starts once the client requests a quote. The system will then invoke Use Case 8.2 Search Project Line-Item List to retrieve the line-item list and 4.9 Search Project to retrieve the project details. The quote details are then stored in the database before the system makes the required Excel sheets that will be sent to the client by the Operations Manager.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">A TSS Booking has been made.
Materials and items have been confirmed</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The client requests a quote</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td>ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td colspan="2">Automated Action</td></tr>
<tr><td>Step 1: The client requests a quote</td><td>Step 2: The Operations Managers clicks the “Quote” button in the Home Screen.</td><td colspan="2">Step 3: The system responds by loading the “Quote” screen with the following controls:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Quote</td><td>Heading</td><td>None</td></tr>
</table></td></tr>
</table>

<table>
<tr><td rowspan="4"></td><td></td><td></td><td>
<table>
<tr><td>Insert keyword:</td><td>Label</td><td>This label is for the search parameter</td></tr>
<tr><td>Search Param-eter</td><td>Textbox</td><td>None</td></tr>
<tr><td>Search</td><td>Button</td><td>None</td></tr>
<tr><td>Search Result Table</td><td>Table</td><td>This table is not visible until the system has searched for a Quote and a match has been found</td></tr>
<tr><td>View</td><td>Button</td><td>This button is found in each row of the table to view the result.</td></tr>
<tr><td>+</td><td>Button</td><td>This control is used to add a quote</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 4: The Operations Manager clicks on the “+” button.</td><td>Step 5: Invoke Use Case 8.2 “Search Project Line-Item List”. The system will search for and retrieve the following information:
Line-Item List Item
Line-Item List Price
Line-Item List Unit</td></tr>
<tr><td></td><td></td><td>Step 6: Invoke Use Case 4.9 “Search Project”. The system will search for and retrieve the following information:
Site Name
Site Size
Site Type
Region
BS Number
Site Location
Budget Code
Mast Type
Design Load
Equipment Housing
Site Size</td></tr>
<tr><td></td><td></td><td>Step 7: The system uses a SQL read query to read the CompanyInformation table, to get the following attributes:</td></tr>
</table>

- Company Name
- Company Logo
- Signature

And to read the **VAT** table, to get the following attribute:

- VAT Amount

And to read the **Client** table, to get the following attribute:

- Client Logo

Step 8: The system responds by loading the "Create Quote" **Screen** with the following controls:

Input Name	Control Name	Notes
Create Quote	Heading	None
Title	Label	This is the label for the title of the quote
Title	Textbox	None
Date of Quotation	Label	This is the label for the date of the quotation
Date of Quotation	Textbox	None
Quotation Number	Label	This is the label for the number of the quotation
Quotation Number	Textbox	None
Quote Revision	Label	This is the label for the quote revision number
Quote Revision	Textbox	None
Packages	Heading	None
Package Pricing Table	Table	This is the table for package

					pricing, pre-populated with materials required for this quote
			Package Pricing	Table Heading	This is the label for the
			Program Management	Table Sub-heading	This includes the program management packages
			Design Packages	Table Sub-heading	This includes the design packages
			SAP Material ID	Column Heading	This includes the identification of the material
			Material Number Used By Vendor	Column Heading	This includes the number that the vendor uses to identify the material
			Packages	Column Heading	This includes the name of the package
			Unit	Column Heading	This includes the unit used by the vendor
			Unit Price	Column Heading	This includes the price of the particular unit
			Quantity	Column Heading	This includes the quantity desired by the Operations Manager
			Quantity	Textbox	Used by the Operations Manager to fill in his desired quantity

			Total	Column Heading	This includes the totals of the price of the price multiplied by the quantity
			Totals	Column Content	Automatically updated based on the quantity chosen to show the total of each row.
			Material Table	Table	This table is pre-populated with material
			RF / CAP/ General	Column Heading	This separates the item of the table by use
			SAP ID	Column Heading	This includes the identification of the item
			Description	Column Heading	This includes the description of the item
			Size	Column Heading	This includes the size of the item
			Unit	Column Heading	This includes the unit of the item
			Quantity	Textbox	Used by the Operations Manager to fill in his desired quantity
			Unit Price	Column Heading	This includes the price of the particular unit
			Total Cost	Column Content	Automatically updated

					based on the quantity chosen
			Material Supplied by Vodacom Table	Table	This table is pre-populated with material supplied by Vodacom
			Part Number		
			Material Number	Column Heading	This includes the identification of the item
			Description	Column Heading	This includes the description of the item
			Unit	Column Heading	This includes the unit of the item
			Quantity	Textbox	Used by the Operations Manager to fill in his desired quantity
			Civil Requirement	Table	This is for civil requirement related items
			Item	Column Heading	This includes the identification of the item
			Description	Column Heading	This includes the description of the item
			Unit	Column Heading	This includes the unit of the item
			Quantity	Textbox	Used by the Operations Manager to fill in his desired quantity

			Unit Price	Column Heading	This includes the price of the particular unit
			Total Cost	Column Content	Automatically updated based on the quantity chosen
			Transportation / Warehousing	Table	This table is for transportation and warehousing related item
			Item	Column Heading	This includes the identification of the item
			Item Number	Column Heading	This includes the identification of the item
			Description	Column Heading	This includes the description of the item
			Unit	Column Heading	This includes the unit of the item
			Quantity	Textbox	Used by the Operations Manager to fill in his desired quantity
			Unit Price	Column Heading	This includes the price of the particular unit
			Total Cost	Column Content	Automatically updated based on the quantity chosen
			Download	Button	This button is used when the om wants

<table>
<tr>
<td></td>
<td></td>
<td></td>
<td>
<table>
<tr><td></td><td></td><td>to download the Quote Excel sheets.</td></tr>
<tr><td>Create</td><td>Button</td><td>None</td></tr>
<tr><td>Cancel</td><td>Button</td><td>This button is used when the officer wishes to terminate the process</td></tr>
</table>
</td>
</tr>
<tr>
<td rowspan="3"></td>
<td></td>
<td>Step 9: The Operations Manager provides the appropriate details.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Step 10: The Operations Manager clicks the “Create” button in order to generate the quote.</td>
<td>Step 11: The system verifies all the required field have been filled in, and that all the fields have the correct format and contains valid details. [ALT]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Step 12: The system will prompt the officer to confirm the details entered by displaying a “Confirmation” Modal
with the following controls:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Confirmation Required</td><td>Label</td><td>This is the header for the modal</td></tr>
<tr><td>Are you sure the entered details are correct?</td><td>Label</td><td>This is the information used to prompt confirmation from the officer</td></tr>
<tr><td>Yes</td><td>Button</td><td>Selected when the officer</td></tr>
</table>
</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td></td>
<td></td>
<td>
<table>
<tr><td></td><td></td><td>accepts confirmation</td></tr>
<tr><td>No</td><td>Button</td><td>Selected when the officer denies confirmation</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Step 13: The officer selects the “Yes” button to confirm all details. [ALT]</td>
<td>Step 14: The system uses a SQL insert query to save the captured information in the Quote table as follows:
• Quotation Title
• Quotation Number
• Date of Quotation
• Quote Revision
• Quotation Amount
To Client Quote table</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Step 15: The quote will be generated in Excel and will include several sheets, namely:
The Cover Sheet
<table>
<tr><th>Input Name</th><th>Control Name</th><th>Notes</th></tr>
<tr><td>Quotation</td><td>Heading</td><td>None</td></tr>
<tr><td>Company Logo</td><td>image</td><td>None</td></tr>
<tr><td>Client Logo</td><td>Image</td><td>None</td></tr>
<tr><td>Region</td><td>Label</td><td>This is the label for the region</td></tr>
<tr><td>Region</td><td>Textbox</td><td>None</td></tr>
<tr><td>BS Number</td><td>Label</td><td>This is the label for the BS number of the project</td></tr>
<tr><td>BS Number</td><td>Textbox</td><td>None</td></tr>
<tr><td>Site Name</td><td>Label</td><td>This is the label for the name of the particular site.</td></tr>
<tr><td>Site Name</td><td>Textbox</td><td>None</td></tr>
<tr><td>Budget Code</td><td>Label</td><td>This is the label for the budget code of the site</td></tr>
</table>
</td>
</tr>
</table>

			Budget Code	Textbox	None
			Mast Type	Label	This is the label for the mast type
			Mast Type	Textbox	None
			Design Load	Label	This is the label for the design load
			Design Load	Textbox	None
			Equipment Housing	Label	This is the label for housing of the equipment name
			Equipment Housing	Textbox	None
			Site Size	Label	This label is for the size of the particular site
			Site Size	Textbox	None
			Quotation Title	Heading	This heading indicates the title of the quotation
			Contractor Details	Sub-heading	None
			Name	Label	None
			Company Name	Label	This label is for the name of the company
			Quotation Number	Label	None
			Quotation Number	Label	This label indicates the quotation number
			Date of Quotation	Label	None
			Date of Quotation	Label	This label indicates the date of the quotation

Quote Revision	Label	None	
Quote Revision	Label	This is the revision number of the quote	

The Civils Summary Sheet:

Input Name	Control Name	Notes
Name:	Label	None
Company Name	Label	This label is for the name of the company
BS Number:	Label	None
BS Number	Label	This label is for the BS Number
Site Name:	Label	None
Site Name	Label	This label is for the name of the site
Region:	Label	None
Region	Label	This label is for the region
Quotation Number:	Label	None
Date of Quotation	Label	This label is for the date of the quotation
Quotation Date	Label	None
Program Manage-ment	Label	None
Total	Label	This is the total amount of the Program Management cost
Design Packages	Label	None

Total	Label	This is the total amount of the Design Packages cost
Build Packages	Label	None
Total	Label	This is the total amount of the Build Packages cost
Non Standard Bill Items	Label	None
Total	Label	This is the total amount of the non-standard bill items cost
Sub Total	Label	None
Sub Total Amount	Label	This is the total of all the activity costs minus VAT
Add : Value Added Tax	Label	None
VAT Amount	Label	This is the subtotal including VAT
Total	Label	None
Grand Total	Label	This is the subtotal and the VAT amount
Company signature	Label	This is the signature of the Director
Quotation Date	Label	This is the date of the quotation
Approved by Vodacom	Label	This is left empty and will contain a signature by the client
Date	Label	This is left empty and will be completed by the client

The Package Pricing Sheet:

Input Name	Control Name	Notes
Package table	Table	None
SAP Material ID	Table Column	Includes the identification numbers of the materials
Material Number Used by Vendor	Table Column	Includes the identification of the materials used by the vendor
Packages	Table Column	Includes the name of the packages
Program Manage-ment Packages	Label	Sections the table by Program Management Packages
Design Packages	Label	Sections the table by Design Packages
Build Packages	Label	Sections the table by Build Packages
Unit	Table Column	Includes the unit measurement
Unit Price	Table Column	Includes the unit price
Quantity	Table Column	Includes the quantity chosen by the Operation Manager
Total	Table Column	Includes the total

The Calculation Sheet:

Input Name	Control Name	Notes
Calculation Sheet	Heading	None
Cellroy RF Quote	Table	This table is for the quote of Cellroy

			Total	Label	None
			Total Number	Label	Includes the total cost
			RFQ Allowable SH RF Supply and Installation	Label	This label describes the work that Cellroy intends to complete
			Total Amount	Label	This label indicates the total amount
			Quantity Factor	Label	This label indicates the quantity factor
			Site Name:	Label	None
			Site Name and BS Number	Label	This is the label for the site name and BS number of the project
			Material Table	Table	None
			RF/Cap/ General	Column Heading	This separates the item of the table by use
			SAP ID	Column Heading	This includes the identification of the item
			Description	Column Heading	This includes the description of the item
			Size	Column Heading	This includes the size of the item
			Unit	Column Heading	This includes the unit of the item
			Quantity	Column Heading	This includes the quantity for the item
			Unit Price	Column Heading	This includes the price of

					the particular unit
			Total Cost	Column Content	Automatically updated based on the quantity chosen
			General Total	Label	None
			General Total Amount	Label	This includes the general total of all the item costs
			Material Total	Label	None
			Material Total Amount	Label	This includes the total amounts from the whole Material table
			Material to be supplied by Vodacom:	Table	None
			Part Number	Label	This includes the number of each part
			Material Number	Label	This includes the number of the material
			Description	Column Heading	This includes the description of the item
			Unit	Column Heading	This includes the size of the item
			Quantity	Column Heading	This includes the unit of the item
			Mode	Label	None
			Civil Requirement: Vodacom	Table	None
			Item	Column Heading	This includes the name of the title

			Description	Column Heading	This includes the description of the item
			Unit	Column Heading	This includes the unit of the item
			Quantity	Column Heading	This includes the quantity for the item
			Unit Price	Column Heading	This includes the price of the particular unit
			Total Cost	Column Content	Automatically updated based on the quantity chosen
			Total Labour & Material	Label	None
			Total Labour & Material Amount	Label	This includes the total amounts from the whole Material and Labour table
			Transport-ion Warehousing	Table	None
			Item	Column Heading	This includes the name of the title
			Description	Column Heading	This includes the description of the item
			Unit	Column Heading	This includes the size of the item
			Quantity	Column Heading	This includes the unit of the item
			Unit Price	Column Heading	This includes the quantity for the item

			Total Cost	Column Heading	This includes the price of the particular unit
			Total Transport	Label	None
			Total Transport Amount	Label	None
			Labour	Table Label	None
			Item	Column Heading	This includes the name of the title
			Description	Column Heading	This includes the description of the item
			Unit	Column Heading	This includes the size of the item
			Quantity	Column Heading	This includes the unit of the item
			Unit Price	Column Heading	This includes the quantity for the item
			Total Cost	Column Heading	This includes the price of the particular unit
			Total Labour	Label	None
			Total Labour Amount	Label	This includes the total amounts from the whole Material table
			SOW Description	Table	This includes SOW details
			Summary of cost table	Table	None
			Summary of cost	Heading	None

1 Material	Label	This label refers to the Material table
2 Civil Requirement	Label	This label refers to the Civil Requirement table
3 Transport-ation/ Warehousing	Label	This label refers to the appropriate table
4 Labour	Label	This label refers to the Labour table
Amount	Column Heading	None
Sub Total	Label	This is the total of all the activity costs minus VAT
VAT @15%	Label	This is the subtotal including VAT
Total	Label	This is the subtotal and the VAT amount

Step 15: The system notifies the officer that the quote was created successfully on the system, by displaying a "Successful Creation" Modal with the following controls: **[ALT]**

Input Name	Control Type	Notes
Result	Label	This is the header for the modal
The quote has been created successfully	Label	This is the information that indicates that the quote was created successfully.
OK	Button	Selected when the officer

<table>
<tr><td rowspan="4">ALTERNATE COURSES:</td><td></td><td></td><td colspan="3">accepts confirmation</td></tr>
<tr><td></td><td>Step 16: The Operation Manager clicks the “Download” button.</td><td colspan="3">Step 17: An Excel file is sent to the Downloads folder of the Operation Manager’s computer.</td></tr>
<tr><td colspan="5">Step 11: The details provided by the Operations Manager is not valid.

<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Validation Error</td><td>Label</td><td>This is the header for the modal</td></tr>
<tr><td>The entered details are not in the correct format. Please try again.</td><td>Label</td><td>This is the information used to indicate that the entered details are not valid.</td></tr>
<tr><td>OK</td><td>Button</td><td>None</td></tr>
</table>
The Operations Manager clicks the “OK” button.
Return to step 9.</td></tr>
<tr><td colspan="5">Step 12: The quote was not created successfully
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Result</td><td>Label</td><td>This is the header for the modal</td></tr>
<tr><td>An unexpected error has been encountered. Please try again later.</td><td>Label</td><td>This is the information that indicates that the quote was not created successfully.</td></tr>
<tr><td>OK</td><td>Button</td><td>Selected when the Operations Manager accepts confirmation</td></tr>
</table>
The Operations Manager clicks the “OK” button
Terminate use case.</td></tr>
<tr><td>CONCLUSION:</td><td colspan="5">This use case concludes when a quote has been created successfully</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="5">• A quote is created and added to the Client_Quote table.
• The status is updated to “Awaiting Purchase Order” in the Project table</td></tr>
<tr><td>BUSINESS RULES:</td><td colspan="5">• Only the Operations Manager can compile the quote.</td></tr>
</table>

IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	The Operations Manager sends the quote to the client for confirmation
OPEN ISSUES:	none

1.2.2.2 Search Quote

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Tshepang Mashao</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Search Quote</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">2.2</td><td rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">User (PBA) [Director, Operations Manager, Procurements and Logistics Officer and Financial Officer]</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">The user provides search criteria to search for a quote. The system will capture the search criteria and display the results. The Operations Manager will then select the quote he desired to search for.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The user must be logged into the system.
• The quote must exist on the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">A user desires to search for a quote</td></tr>
<tr><td>TYPICAL COURSE OF EVENTS:</td><td>ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td></td><td>Manual Action</td><td colspan="2">Automated Action</td></tr>
<tr><td></td><td>Step 1: The user clicks the “Quote” button in the Home Screen.</td><td></td><td>Step 2: The system responds by loading the “Quote” screen with the following controls:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Quote</td><td>Heading</td><td>None</td></tr>
<tr><td>Insert keyword:</td><td>Label</td><td>This label is for the search parameter</td></tr>
<tr><td>Search Param-eter</td><td>Textbox</td><td>None</td></tr>
</table></td></tr>
</table>

			Search | Button | None **Search Result Table** | Table | This table is not visible until the system has searched for a Quote and a match has been found **View** | Button | This button is found in each row of the table to view the result. **Cancel** | Button | The button is used when the user wishes to terminate the use case **+** | Button | This control is used to add a quote
	Step 3: The user provides the appropriate search parameter for the quote and continues by clicking the "Search" button. **[ALT]**		**Step 4:** The system captures the search criteria provided and validates the information, ensures that it is filled in and in the correct format against the Quote table **[ALT]:** **Input Name** | **Attribute in Table** | **Validation Requirements** **Title** | | Required **Quotation Number** | | Required **Date of Quotation** | | Required
			Step 5: An SQL read query is used to read the following attributes from the following tables based on the parameter provided: From the **Quote** table: • QuoteTitle • QuoteNumber • QuoteDate
			Step 6: The system populates the Search Result Table with the data received in the previous step. **[ALT]**

Step 7: The user selects the quote he requested to search for from the table by clicking the "View" button

Step 8: The system displays the selected Quote information by loading the View Quote Screen: **[ALT]**

Control Name	Control Type	Notes
View Quote	Heading	This serves as a header for the modal
Title	Label	This label shows the title of the quote
Document	Viewer	This viewer displays the quote
X	Button	Selected when the user wants to close the modal
Update	Button	A floating action button selected when the user wants to update the quote

ALTERNATE COURSES:

Step 4: The system failed to validate the search parameters.

Input Name	Control Type	Notes
Validation Error	Label	This is the header for the modal
The entered details are not in the correct format. Please try again.	Label	This is the information used to indicate that the entered details are not valid.
OK	Button	None

The user clicks the "OK" button.

Return to step 3.

Step 5: The entered keyword results in no matches found. A **modal** will display to indicate the error, with the following controls:

Input Name	Control Type	Notes
Result	Label	This is the header for the modal.
There is no quote that matches with the entered details.	Label	This is the information used to indicate that the entered details have no matches.

<table>
<tr><td></td><td>
<table>
<tr><td>OK</td><td>Button</td><td>None</td></tr>
</table>
The user clicks the “OK” button
Terminate use case.
</td></tr>
<tr><td></td><td>
Step 8: The system failed to display the details of the selected Client Quote. A “Encountered error” modal will display, with the following controls:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Result</td><td>Label</td><td>This is the header for the modal</td></tr>
<tr><td>An unexpected error has been encountered. Please try again later.</td><td>Label</td><td>This is the information that indicates that the employee type was not created successfully.</td></tr>
<tr><td>OK</td><td>Button</td><td>Selected when the officer accepts confirmation</td></tr>
</table>
The officer clicks the “OK” button
Terminate use case.
</td></tr>
<tr><td>CONCLUSION:</td><td>The system retrieves and displays the quote from the Client Quote table.</td></tr>
<tr><td>POST-CONDITION:</td><td>• The user can view information of the quote.
• The searched quote has been found.</td></tr>
<tr><td>BUSINESS RULES:</td><td>none</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td>none</td></tr>
<tr><td>ASSUMPTIONS:</td><td>none</td></tr>
<tr><td>OPEN ISSUES:</td><td>none</td></tr>
</table>

1.2.2.3 Capture Purchase Order

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s): Tshepang Mashao</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Capture Purchase Order</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">2.3</td><td rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Client</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">Financial Officer</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">The system must allow the financial officer to upload the purchase document to the system. The user can attach the employee’s certifications and health and safety certificates to the employee’s file.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The officer must be logged onto the system</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The officer requests to upload an employee document.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td>ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td colspan="2">Automated Action</td></tr>
<tr><td>Step 1: The client sends a purchase order</td><td>Step 2: The Financial Officer navigates to the Quote screen and clicks on the “+” button, then selects “Add</td><td colspan="2">Step 3: The system responds by displaying the Upload Purchase Order Modal, with the following controls:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Upload Purchase Order</td><td>Heading</td><td>None</td></tr>
</table></td></tr>
</table>

<table>
<tr>
<td></td>
<td></td>
<td>Purchase Order”</td>
<td>
<table>
<tr><td>Cancel</td><td>Button</td><td>This button is used when the Financial Officer wishes to terminate the process</td></tr>
<tr><td>Upload</td><td>Button</td><td>This button is used when the Financial Officer wishes to upload the document</td></tr>
<tr><td>Save</td><td>Button</td><td>This button is used when the officer wants the file to be stored on the system. Until all the details are valid, it is disabled.</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Step 4: The Financial Officer enters the date received and total amount.</td>
<td>Step 5: The system ensures that the expiry date is valid. [ALT]
<table>
<tr><th>Input Name</th><th>Attribute in Table</th><th>Validation Requirements</th></tr>
<tr><td>Date Received</td><td>ClientPO_ DateReceived</td><td>Date</td></tr>
<tr><td>Amount</td><td>ClientPO_ Amount</td><td>Integer(10)</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Step 6: The Financial Officer clicks the “Upload” button to upload the document.</td>
<td>Step 7: The system displays an upload dialogue for the Financial Officer to choose the file to upload.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Step 8: The Financial Officer uploads the Purchase Order by choosing</td>
<td>Step 9: The system uploads the selected file. [ALT]</td>
</tr>
</table>

<table>
<tr><td></td><td></td><td>the appropriate file.</td><td></td></tr>
<tr><td></td><td></td><td>Step 10: The Financial Officer clicks the “Save” button. [ALT]</td><td>Step 11: The system prompts the officer to confirm the saving of the employee document by displaying a modal with the following controls:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Confirmation Required</td><td>Label</td><td>This is the header for the modal</td></tr>
<tr><td>Are you sure the entered details and document are correct?</td><td>Label</td><td>This is the information used to prompt confirmation from the Financial Officer</td></tr>
<tr><td>Yes</td><td>Button</td><td>Selected when the Financial Officer accepts confirmation</td></tr>
<tr><td>No</td><td>Button</td><td>Selected when the Financial Officer denies confirmation</td></tr>
</table></td></tr>
<tr><td></td><td></td><td>Step 12: The Financial Officer confirms the captured information by clicking the “Yes” button. [ALT]</td><td>Step 13: The system uses an SQL insert query to save the uploaded employee document form in the Client Purchase Order table and saves the Expiry Date and Employee information.[ALT]
<table>
<tr><th>Details to be saved</th><th>Attribute in Table</th><th>Details</th></tr>
<tr><td>Client Purchase Order</td><td>ClientPO_ID</td><td>Generated by the system by reading the last ClientPurchase</td></tr>
</table></td></tr>
</table>

					Order_ID from the Client Purchase Order table and incrementing it by one.
			Document	ClientPO_Doc	None
			Date Received	ClientPO_DateReceived	None
			Amount	ClientPO_Amount	None
			Step 14: The system notifies the officer that the Purchase Order document was uploaded successfully by using a Modal with the following controls: **[ALT]**		
			Input Name	**Control Type**	**Notes**
			Result	Label	This is the header for the modal
			The document has been uploaded successfully	Label	This is the information that indicates that the document was uploaded successfully.
			OK	Button	Selected when the Financial Officer accepts confirmation
		Step 15: The Financial Officer acknowledges the notification by clicking			

		the “OK” button.	

ALTERNATE COURSES:

Step 5: The details provided by the Financial Officer is not valid.

Input Name	Control Type	Notes
Validation Error	Label	This is the header for the modal
The entered details are not in the correct format. Please try again.	Label	This is the information used to indicate that the entered details are not valid.
OK	Button	None

The Financial Officer clicks the “OK” button.

Return to step 4.

Step 9: The Financial Officer uploads a file that is not in the correct format. The system will show an error message below the text: “This file is not in the correct format”. The “Save” button will remain disabled.

Return to Step 8.

Step 10: The Financial Officer wishes to terminate this use case by clicking on the “Cancel” button.

The system then displays a **Modal** with the following controls, to request confirmation from the Financial Officer:

Input Name	Control Type	Notes
Confirmation Required	Label	This is the header for the modal
Are you sure you want to cancel this process?	Label	This is the information used to prompt confirmation from the Financial Officer
Yes	Button	Selected when the Financial Officer accepts confirmation
No	Button	Selected when the Financial Officer denies confirmation

The Financial Officer confirms the termination of this use case by clicking on the “Yes” button. The system will respond by terminating the use case.

ALT: The Financial Officer denies confirmation for the termination of this use case by clicking on the “No” button. Return to Step 5.

Step 12: The Financial Officer does not confirm the upload of the Employee Compliance Document by clicking “No”. The system displays a confirmation Modal:

Input Name	Control Type	Notes
Confirmation Required	Label	This is the header for the modal
Are you sure you want to cancel this process?	Label	This is the information used to prompt confirmation from the Financial Officer
Yes	Button	Selected when the Financial Officer accepts confirmation
No	Button	Selected when the Financial Officer denies confirmation

The Financial Officer confirms the termination of this use case by clicking on the “Yes” button. The system will respond by terminating the use case.

ALT: The Financial Officer denies confirmation for the termination of this use case by clicking on the “No” button. Return to Step 12.

Step 13: A database error occurs in the upload process. Display error message, by means of a Modal, with the following controls:

Input Name	Control Type	Notes
Result	Label	This is the header for the modal
An unexpected error has been encountered. Please try again later.	Label	This is the information that indicates that the document was not uploaded successfully.
OK	Button	Selected when the Financial Officer accepts confirmation

The Financial Officer clicks the “OK” button

Terminate use case.

Step 14: The document was not uploaded successfully

Input Name	Control Type	Notes
Result	Label	This is the header for the modal

	An unexpected error has been encountered. Please try again later.	Label	This is the information that indicates that the document was not uploaded successfully.
	OK	Button	Selected when the Financial Officer accepts confirmation
	The Financial Officer clicks the “OK” button Terminate use case.		
CONCLUSION:	The system displays a message that the document was uploaded successfully.		
POST-CONDITION:	• An employee document is uploaded and stored in the Employee Compliance Document table. • The database must maintain data integrity after the employee document is uploaded.		
BUSINESS RULES:	• The file must be in PDF format.		
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none		
ASSUMPTIONS:	none		
OPEN ISSUES:	none		

1.2.3. Administration Subsystem

1.2.3.1 Create Health and Safety file

Gohvan Construction			
Author(s): Matthew Veltman	**Date: 28/06/2021**	**Version: 1.0**	
USE CASE NAME:	**Create Health and Safety file**	**USE CASE TYPE**	
USE CASE ID:	**3.1**	**Business Requirements:** ☐	
PRIORITY:	High	**System Analysis:** ☐	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	Health and Safety Officer		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	Client		
OTHER INTERESTED STAKEHOLDERS:	None		
DESCRIPTION:	This use case describes the process the Health and Safety Officer will go through to create a Health and Safety file containing Health and Safety documents. The use case begins when the Health and Safety officer needs to create a Health and Safety file. The use case concludes when the Health and Safety officer has created and sent the file to the client.		
PRE-CONDITION:	Health and Safety officer needs to be logged into the system.		
TRIGGER:	A health and safety file is required.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action

<table>
<tr>
<td rowspan="5"></td>
<td>Step 1: The Health and Safety officer clicks on the Health and Safety button</td>
<td></td>
<td>Step 2: The system displays the “Upload Health and Safety File” screen with the following elements:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Health and Safety</td><td>Heading</td><td>None</td></tr>
<tr><td>Name</td><td>Label</td><td>None</td></tr>
<tr><td>Name</td><td>Text Box</td><td>None</td></tr>
<tr><td>Type</td><td>Drop-down</td><td>None</td></tr>
<tr><td>Description</td><td>Label</td><td>None</td></tr>
<tr><td>Description</td><td>Long textbox</td><td>None</td></tr>
<tr><td>Upload File</td><td>Label</td><td>None</td></tr>
<tr><td>Upload File</td><td>Button</td><td>None</td></tr>
<tr><td>Confirm</td><td>Button</td><td>None</td></tr>
<tr><td>Cancel</td><td>Button</td><td>None</td></tr>
</table></td>
</tr>
<tr>
<td>Step 3: The Health and Safety officer inputs the details of the file into the required inputs and attaches a file. (ALT)</td>
<td></td>
<td>Step 4: The system captures and validates the inputted information. (ALT)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Step 5: The system displays a modal with the following attributes:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Health and Safety upload message</td><td>Heading</td><td>None</td></tr>
<tr><td>Confirm</td><td>Button</td><td>None</td></tr>
<tr><td>Cancel</td><td>Button</td><td>None</td></tr>
</table></td>
</tr>
<tr>
<td>Step 6: The Health and Safety officer clicks on the confirm button. (ALT)</td>
<td></td>
<td>Step 7: The system uses a SQL query to upload the Name, Type, Description and file to the Health and Safety table.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Step 8: The system displays a modal with the following attributes:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Another Health and Safety upload message</td><td>Heading</td><td>None</td></tr>
</table></td>
</tr>
</table>

<table>
<tr><td></td><td></td><td></td><td><table><tr><td>Confirm</td><td>Button</td><td>None</td></tr><tr><td>Cancel</td><td>Button</td><td>None</td></tr></table></td></tr>
<tr><td></td><td>Step 9: The Health and Safety officer clicks on the confirm button. (ALT)</td><td></td><td>Step 10: The system will return the user to step 3.</td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 3: The Health and Safety officer cancels the uploading of a file. Terminate the use case.</td></tr>
<tr><td></td><td colspan="3">ALT Step 4: The Health and Safety officer’s inputted information has a Validation error. Return to step 3</td></tr>
<tr><td></td><td colspan="3">ALT Step 6: The Health and Safety officer cancels the uploading of a file. Terminate the use case.</td></tr>
<tr><td></td><td colspan="3">ALT Step 9: The Health and Safety officer does not want to upload another file. Terminate the use case</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">The Health and Safety file has been uploaded to the system.</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">The Health and Safety file has been uploaded to the system.</td></tr>
<tr><td>BUSINESS RULES:</td><td colspan="3">Only the Health and Safety officer may upload the Health and Safety file.
The client needs to receive a copy of the Health and Safety file
User needs to be logged into the system</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td colspan="3">none</td></tr>
<tr><td>ASSUMPTIONS:</td><td colspan="3">none</td></tr>
<tr><td>OPEN ISSUES:</td><td colspan="3">none</td></tr>
</table>

1.2.3.2 Search Health and Safety file

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author(s):
Matthew Veltman</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Search Health and Safety file</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">3.2</td><td rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Health and Safety Officer</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the process a Health and Safety officer will go through to find a specific Health and Safety file. The use case begins when the Health and Safety officer wants to search for a specific Health and Safety file. The use case concludes when the Health and Safety officer downloads and views the Health and Safety file</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">Needs to be a health and safety file with the correct criteria on the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">A health and safety file is needed by the Health and Safety officer.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The Health and safety officer clicks on Search Health and Safety File.</td><td></td><td colspan="2">Step 2: The system displays the “Search Health and Safety File” screen with the following elements:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Health and Safety</td><td>Heading</td><td>None</td></tr>
<tr><td>Name</td><td>Label</td><td>None</td></tr>
<tr><td>Name</td><td>Text Box</td><td>None</td></tr>
</table></td></tr>
</table>

			<table><tr><td>Type</td><td>Label</td><td>None</td></tr><tr><td>Type</td><td>Drop-down</td><td>None</td></tr><tr><td>Begin Date</td><td>Label</td><td>None</td></tr><tr><td>Begin Date</td><td>Date selector</td><td>None</td></tr><tr><td>End Date</td><td>Label</td><td>None</td></tr><tr><td>End Date</td><td>Date selector</td><td>None</td></tr><tr><td>Confirm</td><td>Button</td><td>None</td></tr><tr><td>Cancel</td><td>Button</td><td>None</td></tr></table>
	Step 3: The Health and safety officer inputs the Health and Safety file information. **(ALT)**		**Step 4:** The system captures the user's inputted information. **(ALT)**
			Step 5: A SQL query is used to retrieve the Health and safety file from the **Health and Safety** table that matches the user's inputted information (**Name, Type, Date, Description, File). (ALT)**
			Step 6: The system displays the "Searched Health and Safety File" screen with the following elements:<table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Health and Safety</td><td>Heading</td><td>None</td></tr><tr><td>Name</td><td>Label</td><td>None</td></tr><tr><td>Name</td><td>Label</td><td>None</td></tr><tr><td>Type</td><td>Label</td><td>None</td></tr><tr><td>Type</td><td>Label</td><td>None</td></tr><tr><td>Date</td><td>Label</td><td>None</td></tr><tr><td>Date</td><td>Label</td><td>None</td></tr><tr><td>Download</td><td>Button</td><td>None</td></tr><tr><td>Close</td><td>Button</td><td>None</td></tr></table>
	Step 7: The Health and Safety officer clicks the download button. **(ALT)**		**Step 8:** The system allows the user to download the file.

ALTERNATE COURSES:	**ALT Step 3:** The Health and Safety officer cancels the searching of a file. Terminate the use case.
	ALT Step 4: The Health and Safety officer's inputted information has a Validation error. Return to step 3
	ALT Step 5: The Health and safety officer's inputted data does not match any records in the health and safety table. Return to use case 4
	ALT Step 7: The Health and Safety officer cancels the searching of a file. Terminate the use case.
CONCLUSION:	The user receives the desired Health and Safety file.
POST-CONDITION:	The desired Health and Safety file has been received.
BUSINESS RULES:	User needs to be logged into the system
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	There is a Health and Safety file meeting the criteria uploaded on the system
OPEN ISSUES:	none

1.2.3.3 Create Health and Safety Type

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author(s):
Matthew
Veltman</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Create Health and Safety Type</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">3.3</td><td rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Moderate</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Health and Safety Officer</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes how the health and safety officer creates a new type of Health and safety document to add to the system. The use case begins with the Health and Safety wanting to create a new Health and Safety type. This use case concludes when the Health and Safety officer has created a new Health and Safety type.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">Health and Safety officer needs to be logged into the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">A new health and safety file type is required.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The health and safety officer clicks Create new Health and Safety type.</td><td></td><td colspan="2">Step 2: The system displays a modal with the following attributes:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Health and Safety file type</td><td>Heading</td><td>None</td></tr>
<tr><td>Type</td><td>Label</td><td>None</td></tr>
<tr><td>Type</td><td>Textbox</td><td>None</td></tr>
<tr><td>Confirm</td><td>Button</td><td>None</td></tr>
<tr><td>Cancel</td><td>Button</td><td>None</td></tr>
</table></td></tr>
</table>

<table>
<tr><td rowspan="3"></td><td>Step 3: The Health and Safety officer enters the new Health and Safety file type in the textbox. (ALT)</td><td></td><td>Step 4: The system captures and validates the user's inputted data. (ALT)</td></tr>
<tr><td></td><td></td><td>Step 5: The system displays a modal with the following attributes:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Confirm Health and Safety file type</td><td>Heading</td><td>None</td></tr>
<tr><td>Type</td><td>Label</td><td>None</td></tr>
<tr><td>Type</td><td>label</td><td>None</td></tr>
<tr><td>Confirm</td><td>Button</td><td>None</td></tr>
<tr><td>Cancel</td><td>Button</td><td>None</td></tr>
</table></td></tr>
<tr><td>Step 6: The Health and Safety officer clicks the confirm button. (ALT)</td><td></td><td>Step 7: The system uses a SQL query to add the new Health and Safety file type to the Health and Safety table's type attribute.</td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 3: The Health and Safety officer cancels the creation of a file type. Terminate the use case.</td></tr>
<tr><td></td><td colspan="3">ALT Step 4: The Health and Safety officer's inputted information has a Validation error. Return to step 3</td></tr>
<tr><td></td><td colspan="3">ALT Step 6: The Health and Safety officer cancels the creation of a file type. Terminate the use case.</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">The Health and Safety officer has created a new File type</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">New file type has been created</td></tr>
<tr><td>BUSINESS RULES:</td><td colspan="3">Only the Health and Safety officer may create a new file type.</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td colspan="3">none</td></tr>
<tr><td>ASSUMPTIONS:</td><td colspan="3">none</td></tr>
<tr><td>OPEN ISSUES:</td><td colspan="3">none</td></tr>
</table>

1.2.3.4 Search Health and Safety Type

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author(s):
Matthew
Veltman</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Search HS Type</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">3.4</td><td rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Moderate</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Health and Safety Officer</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes how the health and safety officer searches for type of Health and safety document to add to the system. The use case begins with the Health and Safety wanting to search for a Health and Safety type. This use case concludes when the Health and Safety officer has viewed the Health and Safety type.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">Health and Safety officer needs to be logged into the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">Health and Safety officer wants to view a Health and safety file.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td colspan="2">Step 1: Health and Safety officer clicks Search Health and Safety file type button.</td><td></td><td>Step 2: The system displays a modal with the following attributes:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Search Health and Safety file type</td><td>Heading</td><td>None</td></tr>
<tr><td>Type</td><td>Label</td><td>None</td></tr>
<tr><td>Type</td><td>Textbox</td><td>None</td></tr>
<tr><td>Confirm</td><td>Button</td><td>None</td></tr>
<tr><td>Cancel</td><td>Button</td><td>None</td></tr>
</table></td></tr>
</table>

<table>
<tr><td rowspan="3"></td><td>Step 3: The Health and Safety officer enters the Health and Safety file type in the textbox. (ALT)</td><td></td><td>Step 4: The system captures and validates the user's inputted data (ALT)</td></tr>
<tr><td></td><td></td><td>Step 5: A SQL query is used to retrieve the Health and Safety file type given by the user to the type attribute in the Health and Safety Type table. (ALT)</td></tr>
<tr><td></td><td></td><td>Step 6: The system displays a modal with the following attributes:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Searched Health and Safety file type</td><td>Heading</td><td>None</td></tr>
<tr><td>Type</td><td>Label</td><td>None</td></tr>
<tr><td>Type</td><td>label</td><td>None</td></tr>
<tr><td>Close</td><td>Button</td><td>None</td></tr>
</table></td></tr>
<tr><td rowspan="3">ALTERNATE COURSES:</td><td colspan="3">ALT Step 3: The Health and Safety officer cancels the search of a file type. Terminate the use case.</td></tr>
<tr><td colspan="3">ALT Step 4: The Health and Safety officer's inputted information has a Validation error. Return to step 3</td></tr>
<tr><td colspan="3">ALT Step 5: The Health and safety officer's inputted data does not match any records in the health and safety type table. Return to use case 4</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">The health and safety officer has viewed the desired HS file type.</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">The health and safety officer has viewed the desired HS file type.</td></tr>
<tr><td>BUSINESS RULES:</td><td colspan="3">The user needs to be logged into the system.</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td colspan="3">none</td></tr>
<tr><td>ASSUMPTIONS:</td><td colspan="3">none</td></tr>
<tr><td>OPEN ISSUES:</td><td colspan="3">none</td></tr>
</table>

1.2.3.5 Create Drawing Type

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author(s):
Matthew
Veltman</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Create Drawing Type</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">3.5</td><td rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Moderate</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Draftsperson</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes how the Draftsperson creates a new type of drawing to add to the system. The use case begins with the Draftsperson wanting to create a new Drawing type. This use case concludes when the Draftsperson has created a new Drawing type.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">Draftsperson needs to be logged into the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">A new drawing type is required.</td></tr>
<tr><td rowspan="3">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The Draftsperson clicks Create new drawing type.</td><td></td><td>Step 2: The system displays a modal with the following attributes:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Drawing type</td><td>Heading</td><td>None</td></tr>
<tr><td>Type</td><td>Label</td><td>None</td></tr>
<tr><td>Type</td><td>Textbox</td><td>None</td></tr>
<tr><td>Confirm</td><td>Button</td><td>None</td></tr>
<tr><td>Cancel</td><td>Button</td><td>None</td></tr>
</table></td></tr>
<tr><td>Step 3: The Draftsperson enters the</td><td></td><td>Step 4: The system captures and validates the user's inputted data. (ALT)</td></tr>
</table>

<table>
<tr><td rowspan="3"></td><td>new Drawing type in the textbox. (ALT)</td><td></td><td></td></tr>
<tr><td></td><td></td><td>Step 5: The system displays a modal with the following attributes:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Confirm drawing type</td><td>Heading</td><td>None</td></tr>
<tr><td>Type</td><td>Label</td><td>None</td></tr>
<tr><td>Type</td><td>label</td><td>None</td></tr>
<tr><td>Confirm</td><td>Button</td><td>None</td></tr>
<tr><td>Cancel</td><td>Button</td><td>None</td></tr>
</table></td></tr>
<tr><td>Step 6: The Draftsperson clicks the confirm button. (ALT)</td><td></td><td>Step 7: The system uses a SQL query to add the new drawing type to the Drawing table’s type attribute.</td></tr>
<tr><td rowspan="3">ALTERNATE COURSES:</td><td colspan="3">ALT Step 3: The draftsperson cancels the creation of a drawing type. Terminate the use case.</td></tr>
<tr><td colspan="3">ALT Step 4: The draftsperson’s inputted information has a Validation error. Return to step 3</td></tr>
<tr><td colspan="3">ALT Step 6: The draftsperson cancels the creation of a drawing type. Terminate the use case.</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">The Draftsperson has created a new drawing type</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">New drawing type has been created</td></tr>
<tr><td>BUSINESS RULES:</td><td colspan="3">Only the draftsperson may create a new drawing type.</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td colspan="3">none</td></tr>
<tr><td>ASSUMPTIONS:</td><td colspan="3">none</td></tr>
<tr><td>OPEN ISSUES:</td><td colspan="3">none</td></tr>
</table>

1.2.3.6 Search Drawing Type

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author(s):
Matthew
Veltman</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Search Drawing Type</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">3.6</td><td rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Moderate</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Draftsperson</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes how the Draftsperson searches for type of drawing to add to the system. The use case begins with the Draftsperson wanting to search for a drawing type. This use case concludes when the Draftsperson has viewed the drawing type.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">Draftsperson needs to be logged into the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">Draftsperson wants to view a drawing file.</td></tr>
<tr><td rowspan="3">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: Draftsperson clicks Search drawing type button.</td><td></td><td>Step 2: The system displays a modal with the following attributes:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Search drawing type</td><td>Heading</td><td>None</td></tr>
<tr><td>Type</td><td>Label</td><td>None</td></tr>
<tr><td>Type</td><td>Textbox</td><td>None</td></tr>
<tr><td>Confirm</td><td>Button</td><td>None</td></tr>
<tr><td>Cancel</td><td>Button</td><td>None</td></tr>
</table></td></tr>
<tr><td>Step 3: The Draftsperson</td><td></td><td>Step 4: The system captures and validates the user's inputted data (ALT)</td></tr>
</table>

<table>
<tr><td rowspan="3"></td><td>enters the drawing type in the textbox. (ALT)</td><td></td><td></td></tr>
<tr><td></td><td></td><td>Step 5: A SQL query is used to retrieve the drawing type given by the user to the type attribute in the Drawing Type table. (ALT)</td></tr>
<tr><td></td><td></td><td>Step 6: The system displays a modal with the following attributes:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Searched drawing type</td><td>Heading</td><td>None</td></tr>
<tr><td>Type</td><td>Label</td><td>None</td></tr>
<tr><td>Type</td><td>label</td><td>None</td></tr>
<tr><td>Close</td><td>Button</td><td>None</td></tr>
</table></td></tr>
<tr><td rowspan="2">ALTERNATE COURSES:</td><td colspan="3">ALT Step 3: The draftsperson cancels the search of a drawing type. Terminate the use case.</td></tr>
<tr><td colspan="3">ALT Step 4: The draftsperson’s inputted information has a Validation error. Return to step 3</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">ALT Step 5: The draftsperson’s inputted data does not match any records in the drawing type table. Return to use case 4</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">The draftsperson has viewed the desired drawing type.</td></tr>
<tr><td>BUSINESS RULES:</td><td colspan="3">The user needs to be logged into the system.</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td colspan="3">None</td></tr>
<tr><td>ASSUMPTIONS:</td><td colspan="3">none</td></tr>
<tr><td>OPEN ISSUES:</td><td colspan="3">none</td></tr>
</table>

1.2.3.7 Upload Formal Drawing

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author(s):
Matthew Veltman</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Upload Formal Drawing</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">3.7</td><td rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Draftsperson</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">Client</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the process where the drafts person uploads the formal drawing they have created for a specific site. The use case begins when the drafts person wants to upload the formal drawing. The use case concludes when the drafts person has uploaded the formal drawing.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">The site go-ahead needs to be confirmed before a formal drawing can be made.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The drafts person wants to upload the formal drawing.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The drafts person clicks on the upload formal drawing button</td><td></td><td colspan="2">Step 2: The system displays the “Upload formal drawing” screen with the following elements:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>formal drawing</td><td>Heading</td><td>None</td></tr>
<tr><td>Name</td><td>Label</td><td>None</td></tr>
<tr><td>Name</td><td>Text Box</td><td>None</td></tr>
<tr><td>Type</td><td>Drop-down</td><td>None</td></tr>
<tr><td>Description</td><td>Label</td><td>None</td></tr>
</table></td></tr>
</table>

<table>
<tr><td rowspan="4"></td><td></td><td></td><td><table><tr><td>Description</td><td>Long textbox</td><td>None</td></tr><tr><td>Upload File</td><td>Label</td><td>None</td></tr><tr><td>Upload File</td><td>Button</td><td>None</td></tr><tr><td>Confirm</td><td>Button</td><td>None</td></tr><tr><td>Cancel</td><td>Button</td><td>None</td></tr></table></td></tr>
<tr><td>Step 3: The drafts person inputs the details of the file into the required inputs and attaches a file. (ALT)</td><td></td><td>Step 4: The system captures and validates the inputted information. (ALT)</td></tr>
<tr><td></td><td></td><td>Step 5: The system displays a modal with the following attributes:<table><tr><td>Input Name</td><td>Control Type</td><td>Notes</td></tr><tr><td>Drawing upload confirmation</td><td>Heading</td><td>None</td></tr><tr><td>Confirm</td><td>Button</td><td>None</td></tr><tr><td>Cancel</td><td>Button</td><td>None</td></tr></table></td></tr>
<tr><td>Step 6: The drafts person clicks on the confirm button. (ALT)</td><td></td><td>Step 7: The system uses a SQL query to upload the Name, Type, Description and file to the Drawing table.</td></tr>
<tr><td rowspan="3">ALTERNATE COURSES:</td><td colspan="3">ALT Step 3: The drafts person cancels the uploading of a formal drawing. Terminate the use case.</td></tr>
<tr><td colspan="3">ALT Step 4: The drafts person’s inputted information has a Validation error. Return to step 3</td></tr>
<tr><td colspan="3">ALT Step 6: The drafts person cancels the uploading of a formal drawing. Terminate the use case.</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">The formal drawing has been uploaded.</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">The client needs to review the drawing and present feedback.</td></tr>
<tr><td>BUSINESS RULES:</td><td colspan="3">The site go-ahead needs to be confirmed before a formal drawing can be made.
Only the drafts person can upload a formal drawing.
User needs to be logged into the system</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td colspan="3">none</td></tr>
<tr><td>ASSUMPTIONS:</td><td colspan="3">none</td></tr>
</table>

OPEN ISSUES:	none

1.2.3.8 Upload Second cut Drawing

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author(s):
Matthew
Veltman</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Upload Second cut Drawing</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">3.8</td><td rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Draftsperson</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">Client</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the process of receiving feedback from the client about the formal drawing and the drafts person making a second cut drawing based on the feedback in which the drafts person will upload the second cut drawing to the system. The use case begins when the client gives feedback on the formal drawing. The use case concludes when the drafts person uploads the second cut drawing to the system.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">The formal drawing needs to have been uploaded to the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The client provides feedback on the formal drawing.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The drafts person clicks on the upload second cut drawing button</td><td></td><td colspan="2">Step 2: The system displays the “Upload second cut drawing” screen with the following elements:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>second cut drawing</td><td>Heading</td><td>None</td></tr>
<tr><td>Name</td><td>Label</td><td>None</td></tr>
<tr><td>Name</td><td>Text Box</td><td>None</td></tr>
</table></td></tr>
</table>

<table>
<tr><td rowspan="5"></td><td></td><td></td><td><table><tr><td>Type</td><td>Drop-down</td><td>None</td></tr><tr><td>Description</td><td>Label</td><td>None</td></tr><tr><td>Description</td><td>Long textbox</td><td>None</td></tr><tr><td>Upload File</td><td>Label</td><td>None</td></tr><tr><td>Upload File</td><td>Button</td><td>None</td></tr><tr><td>Confirm</td><td>Button</td><td>None</td></tr><tr><td>Cancel</td><td>Button</td><td>None</td></tr></table></td></tr>
<tr><td>Step 3: The drafts person inputs the details of the file into the required inputs and attaches a file. (ALT)</td><td></td><td>Step 4: The system captures and validates the inputted information. (ALT)</td></tr>
<tr><td></td><td></td><td>Step 5: The system displays a modal with the following attributes:<table><tr><td>Input Name</td><td>Control Type</td><td>Notes</td></tr><tr><td>Drawing upload confirmation</td><td>Heading</td><td>None</td></tr><tr><td>Confirm</td><td>Button</td><td>None</td></tr><tr><td>Cancel</td><td>Button</td><td>None</td></tr></table></td></tr>
<tr><td>Step 6: The drafts person clicks on the confirm button. (ALT)</td><td></td><td>Step 7: The system uses a SQL query to upload the Name, Type, Description and file to the Drawing table.</td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 3: The drafts person cancels the uploading of a second cut drawing. Terminate the use case.</td></tr>
<tr><td></td><td colspan="3">ALT Step 4: The drafts person’s inputted information has a Validation error. Return to step 3</td></tr>
<tr><td></td><td colspan="3">ALT Step 6: The drafts person cancels the uploading of a second cut drawing. Terminate the use case.</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">The second cut drawing has been uploaded.</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">Uploaded second cut drawing.</td></tr>
<tr><td>BUSINESS RULES:</td><td colspan="3">• The site go-ahead needs to be confirmed before a formal drawing can be made.
• Only the drafts person can upload a formal drawing.
• User needs to be logged into the system</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td colspan="3">none</td></tr>
</table>

ASSUMPTIONS:	The client gives feedback on the formal drawing.
OPEN ISSUES:	none

1.2.3.9 Search As-built document

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td colspan="2">Author(s):
Matthew
Veltman</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Search As-built document</td><td colspan="2">USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">3.9</td><td colspan="2" rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="4">Health and Safety Officer</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="4">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="4">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="4">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="4">This use case describes the process in which the Health and Safety Officer searches for a specific As-built document on the system. The use case begins when the Health and Safety Officer wants to find a specific As-built document. The use case concludes when the Health and Safety Officer downloads and views the specific As-built document.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="4">There needs to be an As-built with the certain criteria already on the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="4">The Health and Safety Officer is looking for a specific As-built document.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td>Step 1: The Health and safety officer clicks on Search As-built File.</td><td></td><td>Step 2: The system displays the “Search As-built File” screen with the following elements:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>As-built</td><td>Heading</td><td>None</td></tr>
<tr><td>Name</td><td>Label</td><td>None</td></tr>
<tr><td>Name</td><td>Text Box</td><td>None</td></tr>
<tr><td>Project Name</td><td>Label</td><td>None</td></tr>
</table></td></tr>
</table>

<table>
<tr><td rowspan="5"></td><td></td><td></td><td><table><tr><td>Project Name</td><td>Drop-down</td><td>None</td></tr><tr><td>Begin Date</td><td>Label</td><td>None</td></tr><tr><td>Begin Date</td><td>Date selector</td><td>None</td></tr><tr><td>End Date</td><td>Label</td><td>None</td></tr><tr><td>End Date</td><td>Date selector</td><td>None</td></tr><tr><td>Confirm</td><td>Button</td><td>None</td></tr><tr><td>Cancel</td><td>Button</td><td>None</td></tr></table></td></tr>
<tr><td>Step 3: The Health and safety officer inputs the As-built file information. (ALT)</td><td></td><td>Step 4: The system captures the user’s inputted information. (ALT)</td></tr>
<tr><td></td><td></td><td>Step 5: A SQL query is used to retrieve the As-built file from the As-built table that matches the user’s inputted information (Name, project name, Date, Description, File). (ALT)</td></tr>
<tr><td></td><td></td><td>Step 6: The system displays the “Searched Health and Safety File” screen with the following elements:<table><tr><td>Input Name</td><td>Control Type</td><td>Notes</td></tr><tr><td>As-built</td><td>Heading</td><td>None</td></tr><tr><td>Name</td><td>Label</td><td>None</td></tr><tr><td>Name</td><td>Label</td><td>None</td></tr><tr><td>Project Name</td><td>Label</td><td>None</td></tr><tr><td>Project Name</td><td>Label</td><td>None</td></tr><tr><td>Date</td><td>Label</td><td>None</td></tr><tr><td>Date</td><td>Label</td><td>None</td></tr><tr><td>Download</td><td>Button</td><td>None</td></tr><tr><td>Close</td><td>Button</td><td>None</td></tr></table></td></tr>
<tr><td>Step 7: The Health and Safety officer clicks the download button. (ALT)</td><td></td><td>Step 8: The system allows the user to download the file.</td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 3: The Health and Safety officer cancels the searching of the file. Terminate the use case.</td></tr>
<tr><td></td><td colspan="3">ALT Step 4: The Health and Safety officer’s inputted information has a Validation error. Return to step 3</td></tr>
</table>

	ALT Step 5: The Health and safety officer's inputted data does not match any records in the As-built table. Return to use case 4
	ALT Step 7: The Health and Safety officer cancels the searching of a file. Terminate the use case.
CONCLUSION:	The system displays the desired As-built document.
POST-CONDITION:	The displays to the Health and Safety Officer the desired As-built document.
BUSINESS RULES:	User needs to be logged into the system
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	There is an As-built document with the specific criteria on the system.
OPEN ISSUES:	none

1.2.3.10 Generate As-Built Document

1.2.4. Construction Subsystem

1.2.4.1 Capture Site-Go-Ahead

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Michael Vosloo</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Capture Site-Go-Ahead</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">4.1</td><td rowspan="3">Business Requirements: ☐
System Analysis: o
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Client (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">Operations Manager (PSA)</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">• Procurement and logistic Officer (ERA)
• Health and Safety Officer (ERA)
• Draftsperson (ERA)</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">The client will email the operations manager that construction on the site may commence, the operations manager will capture the Email in a screen shot and upload the screenshot to the system to activate the project status and indicate that the construction site is under progress on the system. This update will indicate to the Procurement and Logistics Officer, Health and Safety Officer and Draftsperson that the site construction may commence.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• TSS report must be created.
• Client must have sent the Purchase order indicating that quotations created were correct, thus budgets are set out for the construction project.
• User must be logged onto the system</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">. Operations manager receives an email allowing site go-ahead.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The Client sends the Operations</td><td>Step 2: The Operations Manager wants to</td><td></td></tr>
</table>

	Manager the Site-Go-Ahead for the construction site.	upload and capture the Site-Go-Ahead and start the Construction site progress by uploading the corresponding details needed for the construction site.	
		Step 3: The Operations Manager Hovers over the Construction menu strip.	**Step 4:** The system will display the dropdown with the following items: • Projects • Construction Sites • Allocated teams • Compliance Documents • Radio Frequency reports • Snags • Acceptance Documents • Site sign-offs • Commission reports • Site types
		Step 5: The Operations Manager clicks on the “Projects” dropdown menu item.	**Step 6:** The system invokes use case 4.9 “Search Project”
		Step 7: The user clicks on the Site-Go-Ahead card displayed on the Project page.	**Step 8:** The system reads from the **Construction site** table to determine if there is no data present within the table that relates to the **Project** table via a foreign key attribute of the project ID within the **Construction site** table. The system performs a basic select statement performed within the API via an HttpGet request, the request is called by the Typescript component. **[ALT]**
			Step 9: The system displays a modal to the user prompting the user to capture the site-go-ahead

<table>
<tr>
<td></td>
<td></td>
<td></td>
<td>details and Construction site details relating to the project. The following elements are present in the modal:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Capture Site-Go-Ahead</td><td>Heading</td><td>Modal heading</td></tr>
<tr><td>Do you want to capture the Site-Go-Ahead details</td><td>Paragraph</td><td>This paragraph is displayed to the user prompting the user for a response to capture the Site-Go-Ahead</td></tr>
<tr><td>Yes</td><td>Button</td><td>The user clicks the button to confirm to capture the Site-Go-Ahead details</td></tr>
<tr><td>No</td><td>Button</td><td>The user clicks the button to cancel the request to capture the Site-Go-Ahead.</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Step 10: The user clicks on the “Yes” button. [ALT]</td>
<td>Step 11: The system displays the capture Site-Go-Ahead page to the user with the following elements displayed on the page:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Capture Site-Go-Ahead</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Site Name</td><td>Label</td><td>This label indicates to the user that the following input must be the site name.</td></tr>
<tr><td></td><td>Input</td><td>User inputs the site name</td></tr>
<tr><td>Site Address</td><td>Label</td><td>None</td></tr>
</table>
</td>
</tr>
</table>

				Input	User inputs the site Address
			Type of installation	Label	None
				Input	User inputs the sites type of installation
			Area	Label	None
				Input	User inputs the sites area size
			Date started	Label	Prepopulated by the current date the page is opened.
			Date last updated	Label	Prepopulated by the current date the page is opened as it is the first entry for the update.
			Site Contact person Name	Label	None
				Input	User inputs the site contact person number
			Site contact person number	Label	None
				Input	User inputs the site contact person number.
			Physical details	Heading	This is the label to indicate to the user the details of the Physical details need to be entered
			Site Size	Label	None
				Input	User inputs the site size
			Size of equipment housing	Label	None

				Input	User inputs the sites size of equipment housing
			Fencing	Label	None
				Input	User inputs the sites fencing details
			Covering	Label	None
				Input	User inputs the sites covering details.
			Site Details	Heading	This is the Heading to indicate to the user the following inputs are about the Site details.
			Breaker size	Label	None
				Input	User inputs the sites breaker size.
			Signs installed	Label	None
				Input	User inputs the site signs installed.
			Manhole installed	Label	None
				Input	User inputs whether there is a manhole installed on the construction site.
			Equipment housing	Label	None
			Upload	Button	The user is prompted to select the sites Site-Go-Ahead document.
			Design Load	Label	None
				Input	User inputs the sites design load details

			Cooling system details	Heading	This heading indicates that the following input needs to capture the sites cooling system details.
			Item Code	Label	None
				Input (optional)	User inputs the sites cooling system item code.
			Serial Number	Label	None
				Input (optional)	User inputs the sites cooling systems serial number.
			Air conditioner Description	Label	None
				Input (optional)	User inputs the sites Air conditioner description
			Air conditioner size	Label	None
				Input (optional)	User inputs the sites air conditioner size.
			Site Acceptance Document	Heading	This heading indicates that the following input needs to capture the Site Acceptance Document.
			Upload	button	The user is prompted to select the sites Site-Go-Ahead document.
			Capture details	Button	This button is clicked to capture all the entered information in the input fields.

<table>
<tr><td rowspan="4"></td><td></td><td>Step 12: The Operations Manager enters the necessary information into the appropriate input fields requested by the system.</td><td></td></tr>
<tr><td></td><td>Step 13: The Operations Manager Selects the “capture details” button.</td><td>Step 14: The system captures and validates the information supplied by the Operations Manager using angular validation. All input fields that are correctly validated displays a green border around the input fields. [ALT]</td></tr>
<tr><td></td><td></td><td>Step 15: The system displays a confirmation modal to the Operations Manager to confirm the uploading of the entered input fields.
The confirmation modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Confirmation required</td><td>heading</td><td>Modal heading</td></tr>
<tr><td>Are you sure you want to capture the entered construction site details?</td><td>Paragraph</td><td>Queries the user to confirm to upload the entered data in the input fields.</td></tr>
<tr><td>Yes</td><td>Button</td><td>Clicked to confirm to upload the captured data.</td></tr>
<tr><td>No</td><td>Button</td><td>Clicked to reject the captured data.</td></tr>
</table></td></tr>
<tr><td></td><td>Step 16: The Operations Manager clicks the “yes” button to confirm to upload the</td><td>Step 17: The system uses an SQL insert query performed within the API via a HttpPost request made within the Typescript component. The following attributes are inserted and a new entry is</td></tr>
</table>

		captured data. **[ALT]**	created in the **Construction Site** table: • Construction Site ID • Project ID • Site Name • Construction Start date • Date Updated • Site Address • Area • Type of installation • Site Contact person Name • Site contact person number • Site Size • Size of equipment housing • Fencing • Covering • Breaker size • Signs installed • Manhole installed • Design load • Equipment housing • Cooling system document • Item code • Serial Number • Air conditioner • Air conditioner size • Site Acceptance document.
			Step 18: The system updates the Project status in the **Project** table. The following attributes in the Project table are affected with the update performed in the API via an HttpPut request made in the Typescript component: **Project** table: • Project status_ID (read from the **Project Status** table) • Date updated (updates to the current date the use case was performed on)
			Step 19: The system notifies the Operations Manager via a modal that the Site-Go-Ahead and Construction site details have

<table>
<tr><td></td><td></td><td></td><td>been captured and stored in the system. The modal contains the following elements:<table><tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr><tr><td>Site-Go-Ahead and construction site details have been successfully stored in the system.</td><td>label</td><td>Indicates the success of the captured construction site details.</td></tr><tr><td>Ok</td><td>Button</td><td>None</td></tr></table></td></tr>
<tr><td></td><td></td><td>Step 20: Operations Manager clicks the “Ok” button displayed by the modal.</td><td>Step 21: The System generates an email that displays the Site-Go-Ahead has been captured to the system.</td></tr>
<tr><td></td><td></td><td></td><td>Step 22: The system reads the employee email address attribute stored in the Employee table to be able to send the email to the appropriate employees. The following employees are emailed the generated email:
Procurement and logistic Officer
Health and Safety Officer
Draftsperson
Operations Manager</td></tr>
<tr><td></td><td></td><td></td><td>Step 23: The system displays a modal to the Operations manager indicating that the email has been sent to the appropriate employees. The modal contains the following elements:<table><tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr><tr><td>The Site-Go-Ahead email has been sent to the following employees:</td><td>Modal Heading</td><td>Heading of the modal indicating the modal purpose.</td></tr><tr><td>Employee Name</td><td>Label</td><td>Label indicating the Procurement</td></tr></table></td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td></td><td></td><td>and logistic Officers name</td></tr>
<tr><td>Employee name</td><td>Label</td><td>Label indicating the Health and Safety Officer name</td></tr>
<tr><td>Employee name</td><td>Label</td><td>Label indicating the Draftsperson name</td></tr>
<tr><td>Employee Name</td><td>Label</td><td>Label indicating the Operations Manager name</td></tr>
<tr><td>Ok</td><td>Button</td><td>None</td></tr>
</table>
</td></tr>
<tr><td></td><td></td><td>Step 24: The Operations manager clicks the “Ok” button.</td><td></td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 8: The API HttpGet request returns a value as there is data present for the Construction Site details and Site-Go-Ahead. The System redirects the Operations Manager to the view Construction Site page.
The use case is terminated.</td></tr>
<tr><td></td><td colspan="3">ALT Step 10: The Operations Manager clicks on the “No” button displayed on the modal. Indicating the user does not want to capture the Site-Go-Ahead details.
The use case is terminated.</td></tr>
<tr><td></td><td colspan="3">ALT Step 14: The entered information is invalid; the system displays a red border around the incorrect input fields indicating that the entered data is invalid.
Return to step 13</td></tr>
<tr><td></td><td colspan="3">ALT Step 16: The Operations Manager clicks on the “No” button displayed on the modal. Indicating the Operations manager does not want to upload the Construction site details and Site-Go-Ahead details.
Return to step 13</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">The use case will conclude once the Procurement and logistic Officer, Health and Safety Officer and Draftsperson are notified on the Site-Go-Ahead for the construction site.</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">The email on the captured Site-Go-Ahead and construction site details has been sent.</td></tr>
<tr><td>BUSINESS RULES:</td><td colspan="3">• Only the Operations manager can capture the Site-Go-Ahead email onto the system.</td></tr>
</table>

IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.4.2 Search construction

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Michael Vosloo</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Search construction</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">4.2</td><td rowspan="3">Business Requirements: ☐
System Analysis: o
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Medium</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">User (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• Operations Manager
• Quality Control officer
• Director</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case entails the process the system will go through when searching for a specific Construction site details that has been captured and stored on the system relating to the project associated to the Construction site details. The User will enter the desired search criteria into the system and the corresponding Construction site details matching the search criteria will be displayed to the user.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">The Project page is being displayed to the user.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The user wants to search for a Construction site details.</td></tr>
<tr><td rowspan="3">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The user wants to search for a Construction.</td><td></td><td></td></tr>
<tr><td>Step 2: The User hovers over the Construction menu strip.</td><td></td><td>Step 3: The system will display the dropdown with the following items:
• Projects
• Construction Sites</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>

Allocated teams
Compliance Documents
Radio Frequency reports
Snags
Acceptance Documents
Site sign-offs
Commission reports
Site types

</td></tr>
<tr><td></td><td>Step 4: The user clicks on the “Construction Sites” dropdown menu item.</td><td></td><td>Step 5: The system displays the Search Construction page with the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Construction Site</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Search Label</td><td>Label</td><td>This is the label for the search filter function.</td></tr>
<tr><td>Keyword</td><td>Textbox</td><td>To enter a Construction Site name to search for.</td></tr>
<tr><td>Search</td><td>Button</td><td>This button will filter the table for the Searched for construction site details in the table.</td></tr>
<tr><td>Table</td><td>Table</td><td>This table will hold the information about the Construction Site details.</td></tr>
<tr><td>Construction Site ID</td><td>Label</td><td>Part of the table:
Displays the Construction site ID</td></tr>
<tr><td>Construction Site Name</td><td>Label</td><td>Part of the table:
Displays the Construction Site Name.</td></tr>
<tr><td>Site Address</td><td>Label</td><td>Part of the table:
Displays the Construction Site Address.</td></tr>
</table>
</td></tr>
</table>

<table>
<tr>
<td></td>
<td></td>
<td></td>
<td>
<table>
<tr><td>Construction site start date</td><td>Label</td><td>Part of the table:
Displays the Construction start date.</td></tr>
<tr><td>Site progress description title</td><td>Label</td><td>Part of the table:
Displays the Construction progress descriptions title.</td></tr>
</table>
This page will display a table of all the currently active construction sites stored in the system. The table data is read from the Construction Site and the Site Progress description tables in the database and the relevant attributes that are prevalent in the table being displayed are listed accordingly:

Construction_Site_ID
Site Name
Site Address
Construction start date
Site Progress Title (Foreign attribute obtained from the Site Progress description table via an inner join)

The table attributes are read via an HttpGet method in the Typescript component and an API request.
</td>
</tr>
<tr>
<td></td>
<td>Step 6: The user enters the Construction Site search criteria in the search textbox. [ALT]</td>
<td></td>
<td>Step 7: The system will capture and validate the entered information that was typed into the search textbox. The field is validated and displays a green border around the textbox. [ALT]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Step 8: The system will submit the captured search criteria and search for the Construction Site details in the Construction Site table using a parametrised HttpGet request in the typescript component and API request. The system captures and saves the searched result to a list object and</td>
</tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>displays the search result in the list components, replacing the previous Construction site table items as they do not follow the searched criteria. [ALT]</td></tr>
<tr><td></td><td>Step 9: The user clicks on the desired construction site details displayed on the table.</td><td></td><td>Step 10: The system displays Construction Site details page that is attributed by the Construction Site table according to the searched for Construction Site details, the Construction site details are read from the Construction Site table and passed via the API control to the type script component to then be displayed on the Construction site page elements. The Construction site page has the following elements displayed:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Site Name</td><td>Heading</td><td></td></tr>
<tr><td>Site Address</td><td>Label</td><td>This is the label to indicate to the user the details of the Site Address being displayed.</td></tr>
<tr><td>Type of installation</td><td>Label</td><td>This is the label to indicate to the user the details of the Type of installation being displayed.</td></tr>
<tr><td>Area</td><td>Label</td><td>This is the label to indicate to the user the details of the Area being displayed.</td></tr>
<tr><td>Date started</td><td>Label</td><td>This is the label to indicate to the user the details of the Date started being displayed.</td></tr>
<tr><td>Date last updated</td><td>Label</td><td>This is the label to indicate to</td></tr>
</table></td></tr>
</table>

					the user the details of the Date last updated being displayed.
			Site Contact person Name	Label	This is the label to indicate to the user the details of the Site Contact person Name being displayed.
			Site contact person number	Label	This is the label to indicate to the user the details of the Site contact person number being displayed.
			Physical details	Heading	This is the label to indicate to the user the details of the Physical details being displayed.
			Site Size	Label	This is the label to indicate to the user the details of the Site Size being displayed.
			Size of equipment housing	Label	This is the label to indicate to the user the details of the Size of equipment housing being displayed.
			Fencing	Label	This is the label to indicate to the user the details of the Fencing being displayed.
			Covering	Label	This is the label to

					indicate to the user the details of the Covering being displayed.
			Site Details	Heading	This is the Heading to indicate to the user the following details of the Construction site.
			Breaker size	Label	This is the label to indicate to the user the details of the Breaker size being displayed.
			Signs installed	Label	This is the label to indicate to the user the details of the Signs installed being displayed.
			Manhole installed	Label	This is the label to indicate to the user the details of the Manhole installed being displayed.
			Equipment housing	Label	This is the label to indicate to the user the details of the Equipment housing being displayed.
			View	Button	Button is clicked to view the equipment housing document.
			Design Load	Label	This is the label to indicate to the user the details of the Design Load

					being displayed.
			Cooling system details	Label	This is the label to indicate to the user the details of the Cooling system details being displayed.
			Item Code	Label	This is the label to indicate to the user the details of the Item Code being displayed.
			Serial Number	Label	This is the label to indicate to the user the details of the Serial Number being displayed.
			Air conditioner Description	Label	This is the label to indicate to the user the details of the Air conditioner Description being displayed.
			Air conditioner size	Label	This is the label to indicate to the user the details of the Air conditioner size being displayed.
			Site Acceptance Document	Heading	This heading indicates that the following items displayed are relating to the Site Acceptance document.
			View	Button	This button is clicked to view the Site acceptance

					document for the construction site. **Disabled** if there is no acceptance document for the construction site.)
			Edit	Button	This button is clicked to edit the construction site details. Will direct the user to the Edit Construction use case.
ALTERNATE COURSES:	**ALT Step 6:** The user selects a Construction site from the displayed list present on screen. **Skip to step 9**				
	ALT step 7: The entered search criteria is invalid; the system displays a red border around the search textbox indicating that the entered search criteria is invalid. **Return to step 6**				
	ALT Step 8: The search criteria entered into the search textbox does not match any existing records in the database. The system notifies and displays a modal to the user to indicate that the searched for project details do not exist within the database as a null value was obtained from the HttpGet request. **Return to step 6**				
CONCLUSION:	This use case concludes when the Construction Site page is displayed to the user.				
POST-CONDITION:	The Construction site page is displayed to the user with the correct details				
BUSINESS RULES:	All users can search for construction site details.				
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none				
ASSUMPTIONS:	none				
OPEN ISSUES:	none				

1.2.4.3 Edit construction

Gohvan Construction		
Author (s): **Michael Vosloo**	**Date: 28/06/2021**	**Version: 1.0**
USE CASE NAME:	**Edit construction**	**USE CASE TYPE**
USE CASE ID:	**4.3**	**Business Requirements:** ☐
PRIORITY:	Medium	**System Analysis:** o
SOURCE:	Gohvan Construction System	**System Design:** ☑
PRIMARY BUSINESS ACTOR:	Operations Manager (PBA)	
PRIMARY THE SYSTEM ACTOR:	None	
OTHER PARTICIPATING ACTORS:	None	
OTHER INTERESTED STAKEHOLDERS:	• Client • Director • Quality Control Officer • Health and Safety Officer • Financial Officer	
DESCRIPTION:	This use case describes the process the Operations Manager will go through on the system when editing the construction site details. The Operations manager will search for the desired Construction site details that needs to be changed. The System will then present the Construction site details to the Operations manager and prompt the Operations manager to edit the construction site details. The Operations Manager will edit the details on a form presented by the system for the construction site details. The system will then capture the details given, validate that the given data types in the fields are correct and valid. The system will then save the newly entered site details to the system and notify the Operations manager via a pop up that the construction site details are correct.	
PRE-CONDITION:	The Construction site must exist on the system.	
TRIGGER:	The Operations Manger wants to edit the details of the Construction site.	
TYPICAL COURSE		**SYSTEM RESPONSE:**

<table>
<tr><td rowspan="5">OF EVENTS:</td><td>ACTOR ACTION:</td><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: Operations Manger wants to edit the Construction Site details.</td><td></td><td></td></tr>
<tr><td>Step 2: The Operations Manager Hovers over the Construction menu strip.</td><td></td><td>Step 3: The system will display the dropdown with the following items:
Projects
Construction Sites
Allocated teams
Compliance Documents
Radio Frequency reports
Snags
Acceptance Documents
Site sign-offs
Commission reports
Site types</td></tr>
<tr><td>Step 4: The Operations Manager clicks on the “Construction Sites” dropdown menu item.</td><td></td><td>Step 5: The system invokes use case 4.2 “Search Construction”.</td></tr>
<tr><td>Step 6: The Operations manager clicks on the “Update” button displayed on the Construction site details page.</td><td></td><td>Step 7: The system displays the “Update Construction Site details” page with the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Capture Site-Go-Ahead</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Site Name</td><td>Label</td><td>This label indicates to the user that the following input must be the site name.</td></tr>
<tr><td></td><td>Input (Disabled)</td><td>User inputs the site name</td></tr>
</table></td></tr>
</table>

			Site Address	Label	The corresponding input is populated with the current value obtained in the search construction site.
				Input (optional)	User inputs the site Address
			Type of installation	Label	The corresponding input is populated with the current value obtained in the search construction site.
				Input	User inputs the sites type of installation
			Area	Label	The corresponding input is populated with the current value obtained in the search construction site.
				Input (optional)	User inputs the sites area size
			Date started	Label	Populated by the current attribute obtained in the search construction site.
			Date last updated	Label	Prepopulated by the current date the page is opened as it is the first entry for the update.
			Site Contact person Name	Label	The corresponding input is populated with the current value obtained in

					the search construction site.
				Input (optional)	User inputs the site contact person number
			Site contact person number	Label	The corresponding input is populated with the current value obtained in the search construction site.
				Input (optional)	User inputs the site contact person number.
			Physical details	Heading	This is the label to indicate to the user the details of the Physical details need to be entered
			Site Size	Label	The corresponding input is populated with the current value obtained in the search construction site.
				Input (optional)	User inputs the site size
			Size of equipment housing	Label	The corresponding input is populated with the current value obtained in the search construction site.
				Input (optional)	User inputs the sites size of equipment housing
			Fencing	Label	The corresponding input is

					populated with the current value obtained in the search construction site.
				Input (optional)	User inputs the sites fencing details
			Covering	Label	The corresponding input is populated with the current value obtained in the search construction site.
				Input (optional)	User inputs the sites covering details.
			Site Details	Heading	This is the Heading to indicate to the user the following inputs are about the Site details.
			Breaker size	Label	The corresponding input is populated with the current value obtained in the search construction site.
				Input (optional)	User inputs the sites breaker size.
			Signs installed	Label	The corresponding input is populated with the current value obtained in the search construction site.
				Input (optional)	User inputs the site signs installed.

			Manhole installed	Label	The corresponding input is populated with the current value obtained in the search construction site.
				Input (optional)	User inputs whether there is a manhole installed on the construction site.
			Equipment housing	Label	The corresponding input is populated with the current value obtained in the search construction site.
			Upload	Button (optional)	The user is prompted to select the sites Site-Go-Ahead document.
			Design Load	Label	The corresponding input is populated with the current value obtained in the search construction site.
				Input (optional)	User inputs the sites design load details
			Cooling system details	Heading	This heading indicates that the following input needs to capture the sites cooling system details.
			Item Code	Label	The corresponding input is populated with the current value

					obtained in the search construction site.
				Input (optional)	User inputs the sites cooling system item code.
			Serial Number	Label	The corresponding input is populated with the current value obtained in the search construction site.
				Input (optional)	User inputs the sites cooling systems serial number.
			Air conditioner Description	Label	The corresponding input is populated with the current value obtained in the search construction site.
				Input (optional)	User inputs the sites Air conditioner description
			Air conditioner size	Label	The corresponding input is populated with the current value obtained in the search construction site.
				Input (optional)	User inputs the sites air conditioner size.
			Site Acceptance Document	Heading	This heading indicates that the following input needs to capture the Site Acceptance Document.

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>Upload</td><td>button</td><td>The user is prompted to select the sites Site-Go-Ahead document.</td></tr>
<tr><td>Capture details</td><td>Button</td><td>This button is clicked to capture all the entered information in the input fields.</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 8: The Operations Manager enters the necessary information into the appropriate input fields wished to be updated.</td><td></td><td>Step 9: The system captures and validates the information supplied by the Operations Manager using angular validation. All input fields that are correctly validated displays a green border around the input fields. [ALT]</td></tr>
<tr><td></td><td>Step 10: The Operations Manager Selects the “capture details” button.</td><td></td><td>Step 11: The system displays a confirmation modal to the Operations Manager to confirm the updating of the entered input fields.
The confirmation modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Confirmation required</td><td>heading</td><td>Modal heading</td></tr>
<tr><td>Are you sure you want to update the entered construction site details?</td><td>Paragraph</td><td>Queries the user to confirm to update the entered data in the input fields.</td></tr>
<tr><td>Yes</td><td>Button</td><td>Clicked to confirm to update the captured data.</td></tr>
<tr><td>No</td><td>Button</td><td>Clicked to reject the captured updated data.</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 12: The Operations Manager clicks the</td><td></td><td>Step 13: The system uses an SQL update query performed within the API via a HttpPut request made within the Typescript component.</td></tr>
</table>

<table>
<tr><td rowspan="2"></td><td>“yes” button to confirm to update the captured data. [ALT]</td><td></td><td>The following attributes are inserted and update the existing entry in the Construction Site table:
Site Name
Date Updated
Area
Type of installation
Site Contact person Name
Site contact person number
Site Size
Size of equipment housing
Fencing
Covering
Breaker size
Signs installed
Manhole installed
Design load
Equipment housing
Cooling system document
Item code
Serial Number
Air conditioner
Air conditioner size
Site Acceptance document.Air conditioner size
Site Acceptance document.</td></tr>
<tr><td></td><td></td><td>Step 14: The system notifies the Operations Manager via a modal that the Site-Go-Ahead and Construction site details have been updated and stored in the system. The modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Construction Site details have been successfully updated.</td><td>heading</td><td>Modal heading</td></tr>
<tr><td>Ok</td><td>Button</td><td>None</td></tr>
</table></td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 9: The entered information is invalid; the system displays a red border around the incorrect input fields indicating that the entered data is invalid.</td></tr>
</table>

	Return to step 8
	ALT Step 12: The Operations Manager clicks on the “No” button displayed on the modal. Indicating the user does not want to update the Site-Go-Ahead details. **Return to step 8**
CONCLUSION:	This use case concludes when the Operations Manager has been successfully notified on the updated the Construction site details notification.
POST-CONDITION:	The updated construction site details are saved to the **Construction Site** table
BUSINESS RULES:	• The associated project cannot be changed within the construction site. • Only the Operations manager can update the Construction site details.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.4.4 Allocate Team Site

Gohvan Construction			
Author (s): Michael Vosloo	**Date: 28/06/2021**	**Version: 1.0**	
USE CASE NAME:	**Allocate Team Site**	**USE CASE TYPE**	
USE CASE ID:	**4.4**	**Business Requirements:** ☐	
PRIORITY:	High	**System Analysis:** o	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	Operations Manager (PBA)		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	• none		
OTHER INTERESTED STAKEHOLDERS:	• General employees • Site Managers • Client • Human Resource Management • Health and Safety Officer		
DESCRIPTION:	This use describes the process the Operations manager and system will go through when an existing team is allocated to a project to then begin construction. The Operations Manager will select the desired team for the specified project. Once the Teams are allocated to the project, Construction may begin on the construction site.		
PRE-CONDITION:	• Project status is updated once Site-Go-Ahead is received, indicating that teams need to be allocated to the specific project.		
TRIGGER:	Operations Manager requests to assign a team to the project after receiving Site-Go-Ahead		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The Operations manager wants to allocate a		

<table>
<tr><td></td><td>team to a Construction site.</td><td></td><td></td></tr>
<tr><td></td><td>Step 2: The Operations Manager hovers over the Construction menu strip.</td><td></td><td>Step 3: The system invokes use case 4.9 “Search Project”</td></tr>
<tr><td></td><td>Step 4: The Operations Manager clicks on the “Allocated Teams” card on the project page.</td><td></td><td>Step 5: The system displays the Search Allocated teams page populated with the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Allocated Teams</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Search label</td><td>Label</td><td>This is the label for the search filter function.</td></tr>
<tr><td>Keyword</td><td>Textbox</td><td>To enter an allocated teams’ search criteria to search for.</td></tr>
<tr><td>Search</td><td>Button</td><td>This button will filter the table for the Searched for allocated teams’ details in the table.</td></tr>
<tr><td>Table</td><td>Table</td><td>This table will hold the information about the Construction Site details.</td></tr>
<tr><td>Team ID</td><td>Label</td><td>Part of the table: Displays the Team ID.</td></tr>
<tr><td>Team name</td><td>Label</td><td>Part of the table: Displays the Team Name.</td></tr>
<tr><td>Date assigned</td><td>Label</td><td>Part of the table: Displays the date the team was assigned to the Construction site.</td></tr>
</table></td></tr>
</table>

			<table><tr><td>Construction Site Name</td><td>Label</td><td>Part of the table: Displays the Construction Site the team was assigned to.</td></tr><tr><td>Allocate Team</td><td>Button</td><td>This button is clicked to allocate a team to a construction site.</td></tr></table> This page will display a table of all the assigned teams to the construction site specific to the data that was obtained in the Search project use invoke. The table data is read from the **Team Construction Site, Construction Site** and **Team** table. The data obtained and displayed attributes from the tables that are prevalent to the table listed above are: • Team ID • Team Name • Date Assigned • Construction Site Name (inner joined via a foreign key) The table attributes are read via an HttpGet method in the Typescript component and an API request. **[ALT]**
	Step 6: The Operations Manager clicks on the "Allocate Team" button displayed		**Step 7:** The system displays the Allocate Team page. The system **invokes use case 7.2 "Search Team"** to obtain all the teams created within the system. The system captures all the team's details via an HttpGet API request made in the Type script component. The team details are obtained and saved as an array of object. The displayed page contains the following elements: <table><tr><td>Input name:</td><td>Control Type:</td><td>Notes:</td></tr><tr><td>Allocate Team</td><td>Heading</td><td>Page heading</td></tr></table>

			Team Name	Label	None
				Dropdown input	Dropdown displaying all the teams available for the Operations Manager to select. Populated from the team object returned.
			Team ID	Label	None
			Team ID	Input (disabled)	Input displaying the Team ID correlating to the team selected in the previous item selected in the dropdown menu.
			Project Name	Label	None
				Dropdown (disabled)	Dropdown input displaying the list of projects available to allocate the team to.
			Table:	table	Table displaying all the employees in the selected team.

			Employee ID	label	Table Element: displaying the Employees ID
			Employee Name	Label	Table Element: Displaying the Employees Name
			Employee Surname	Label	Table Element: Displaying the Employees Surname
			Employee Role	Label	Table Element: Displaying the Employees Role within the team.
			Allocate Team	Button (disabled)	The user clicks this button to allocate the team to the project.
			Cancel	Button	This button is clicked to cancel the process of allocating a team.
	Step 8: The Operations Manager enters the necessary information into the appropriate input fields for the desired team.		**Step 9:** The system captures and validates the provided information entered into the fields. Allocate Team Button is enabled. **[ALT]**		

<table>
<tr>
<td></td>
<td>Step 10: The Operations Manager selects the “Allocate Team” button. [ALT]</td>
<td></td>
<td>Step 11: The system displays a confirmation modal to the Operations Manager to confirm the allocating of the team to the project.
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Confirmation required</td><td>heading</td><td>Modal heading</td></tr>
<tr><td>Are you sure you want to allocate the team?</td><td>Paragraph</td><td>Queries the user to confirm to allocate the team.</td></tr>
<tr><td>Yes</td><td>Button</td><td>Clicked to confirm to upload the captured data.</td></tr>
<tr><td>No</td><td>Button</td><td>Clicked to reject the captured data.</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 12: The Operations manager clicks the “Yes” button displayed on the modal. [ALT]</td>
<td></td>
<td>Step 13: The system uses an SQL insert query performed via an HttpPost request made within the Typescript component to create a new entry within the Team Construction Site table. The system will enter the following attributes obtained from the entered inputs to create the allocated team.
Team Construction Site table:
• Team ID (foreign key, Primary key) (read from the corresponding team selected)
• Construction Site ID (foreign key, Primary key) (read from the searched for project)
• Date assigned (created on current date the team is allocated)</td>
</tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>Step 14: The system notifies the Operations Manager via a modal that the Team was successfully allocated to the construction site. The modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Team has been successfully allocated to the construction site.</td><td>heading</td><td>Modal heading</td></tr>
<tr><td>Ok</td><td>Button</td><td>None</td></tr>
</table></td></tr>
<tr><td></td><td>Step 15: The Operations manager clicks the “Ok” button.</td><td></td><td>Step 16: The System generates an email and SMS that indicates the team has been assigned to the construction site.</td></tr>
<tr><td></td><td></td><td></td><td>Step 17: The system reads the employee email address and number attribute stored in the Employee table to be able to send the email and SMS to the appropriate employees. The following employees are emailed the generated email:
Health and Safety Officer
Site Manager (obtained from the newly assigned team to the construction site)</td></tr>
<tr><td></td><td></td><td></td><td>Step 18: The system displays a modal to the operations manager indicating that the email has been sent to the appropriate employees. The modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>The health and safety officer and Site manager have been notified on the newly assigned team to the</td><td>Modal Heading</td><td>Heading of the modal indicating the modal purpose.</td></tr>
</table></td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>construction site.</td><td></td><td></td></tr>
<tr><td>Employee Name</td><td>Label</td><td>Label indicating the Health and Safety Officer name.</td></tr>
<tr><td>Employee name</td><td>Label</td><td>Label indicating the Site Managers name.</td></tr>
<tr><td>Ok</td><td>Button</td><td>None</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 19: The Operations manager clicks the “ok” button displayed.</td><td></td><td></td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 9: The entered information is invalid; the system displays a red border around the incorrect input fields indicating that the entered data is invalid. Return to step 10</td></tr>
<tr><td></td><td colspan="3">ALT Step 10: Operations Manager clicks the “Cancel” button.
The Use case is terminated</td></tr>
<tr><td></td><td colspan="3">ALT Step 12: The Operations Manager clicks the “No” button displayed.
Return to step 9</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">This use case concludes once the Operations Manager, Site Manager and Health and safety officer have been notified that the Team has been successfully allocated to the construction project.</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">• At least one team is assigned to a construction project.</td></tr>
<tr><td>BUSINESS RULES:</td><td colspan="3">• More than one team can be assigned to a construction project.
• Only Operations Manager can assign a team to a construction project.</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td colspan="3">none</td></tr>
<tr><td>ASSUMPTIONS:</td><td colspan="3">none</td></tr>
<tr><td>OPEN ISSUES:</td><td colspan="3">none</td></tr>
</table>

1.2.4.5 Search Allocated Team

Gohvan Construction			
Author (s): Michael Vosloo	**Date: 28/06/2021**	**Version: 1.0**	
USE CASE NAME:	**Search Allocated Team**	**USE CASE TYPE**	
USE CASE ID:	**4.5**	**Business Requirements:** ☐	
PRIORITY:	Medium	**System Analysis:** o	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	User (PBA)		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	• Operations Manager • Health and Safety Officer • Director		
DESCRIPTION:	This use case entails the process the system will go through when searching for a specific allocated team that has been captured and stored on the system relating to the specific project. The User will enter the desired search criteria into the system and the corresponding allocated team matching the search criteria will be displayed to the user.		
PRE-CONDITION:	• The user is on the project page. • There are teams allocated to the project.		
TRIGGER:	The User wants to search for allocated team.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The user wants to search for allocated teams to the Construction site.		

<table>
<tr>
<td></td>
<td>Step 2: The user hovers over the Construction menu strip.</td>
<td></td>
<td>Step 3: The system displays the dropdown with the following items:
Projects
Construction Sites
Allocated teams
Compliance Documents
Radio Frequency reports
Snags
Acceptance Documents
Site sign-offs
Commission reports
Site types</td>
</tr>
<tr>
<td></td>
<td>Step 4: The user clicks on the “Allocated Teams” dropdown menu item. [ALT]</td>
<td></td>
<td>Step 5: The system displays the search Allocated Teams page with the following details:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Allocated Teams</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Search label</td><td>Label</td><td>This is the label for the search filter function.</td></tr>
<tr><td>Keyword</td><td>Textbox</td><td>To enter an allocated teams’ search criteria to search for.</td></tr>
<tr><td>Search</td><td>Button</td><td>This button will filter the table for the Searched for allocated teams’ details in the table.</td></tr>
<tr><td>Table</td><td>Table</td><td>This table will hold the information about the Construction Site details.</td></tr>
<tr><td>Team ID</td><td>Label</td><td>Part of the table: Displays the Team ID.</td></tr>
<tr><td>Team name</td><td>Label</td><td>Part of the table: Displays the</td></tr>
</table></td>
</tr>
</table>

			<table><tr><td></td><td></td><td>Team Name.</td></tr><tr><td>Date assigned</td><td>Label</td><td>Part of the table: Displays the date the team was assigned to the Construction site.</td></tr><tr><td>Construction Site Name</td><td>Label</td><td>Part of the table: Displays the Construction Site the team was assigned to.</td></tr></table> This page will display a table of all the assigned teams to the construction site. The table data is read from the **Team Construction Site, Construction Site** and **Team** table. The data obtained and displayed attributes from the tables that are prevalent to the table listed above are: • Team ID • Team Name • Date Assigned • Construction Site Name The table attributes are read via an HttpGet method in the Typescript component and an API request.
	Step 6: The user enters the Allocated teams search criteria in the search text box. **[ALT]**		**Step 7:** The system will capture and validate the entered information that was typed into the search textbox. The field is validated and displays a green border around the textbox. **[ALT]**
			Step 8: The system will submit the captured search criteria and search for the Allocated team details in the **Team Construction Site, Construction Site** and **Team** table using a parametrised HttpGet request in the typescript component and API request. The system captures and saves the

<table>
<tr><td></td><td></td><td></td><td>searched result to a list object and displays the search result in the table components, replacing the previous Team Construction site table items as they do not follow the searched criteria. [ALT]</td></tr>
<tr><td></td><td>Step 9: The user clicks on the desired Team Construction site details displayed on the table.</td><td></td><td rowspan="2">Step 10: The system displays the Allocated Team page that is attributed by the Team Construction Site, Team, Employee Team, Employee and Construction tables according to the searched for Allocated Team details. The Allocated Team details are read from the Team Construction Site, Team and Construction Site tables and passed via the API control to the type script component to then be displayed on the Allocated Team page elements. The Allocated Team page has the following elements displayed:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Allocated Team</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Team Name</td><td>Label</td><td>Displays the Name of the team being displayed.</td></tr>
<tr><td>Construction Site Name</td><td>Label</td><td>Displays the Name of the Construction site the team is assigned to.</td></tr>
<tr><td>Site Manager</td><td>Label</td><td>Displays the Site Manager name of the assigned team.</td></tr>
<tr><td>Date Assigned</td><td>Label</td><td>Displays the date the Team was assigned to the Construction Site.</td></tr>
<tr><td>Total Team Members</td><td>Label</td><td>Displays the number of members within the team.</td></tr>
</table></td></tr>
<tr><td></td><td></td><td></td></tr>
</table>

			Table	table	Table displaying the details of the team members within the team assigned to the Construction site.
			Employee ID	label	Table Element: displaying the Employees ID
			Employee Name	Label	Table Element: Displaying the Employees Name
			Employee Surname	Label	Table Element: Displaying the Employees Surname
			Employee Role	label	Table Element: Displaying the Employees Role within the team.
			Edit	Button	When this button is clicked the user can edit the assigned team details.
ALTERNATE COURSES:	**Alt Step 4:** The user selects the "Allocated Teams" elements on the "Project" page, the system will **skip to step 5** but only display the teams associated to that project. The project is linked to the construction site which entails the **Construction site team** as the **Project** table has a foreign key within the **Construction site** table to then identify the **Construction site** allocated with the **Construction site team**. The Project and Construction Site team will be joined for this step via an inner join function performed in SQL and the API. **Skip to step 5**				
	ALT Step 6a: The user selects a "Allocated Team" from the displayed list present on screen. **Skip to step 9**				

	ALT step 7: The entered search criteria is invalid; the system displays a red border around the search textbox indicating that the entered search criteria is invalid. **Return to step 6**
	ALT Step 8: The search criteria entered into the search textbox does not match any existing records in the database. The system notifies and displays a modal to the user to indicate that the searched for allocated team details do not exist within the database as a null value was obtained from the HttpGet request. **Return to step 6**
CONCLUSION:	The allocated team is displayed to the user
POST-CONDITION:	The Allocated Team page is displayed to the user with the correct details.
BUSINESS RULES:	All users can search for allocated teams for a construction site.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.4.6 Update Allocated Team

Gohvan Construction			
Author (s): **Michael Vosloo**	**Date: 28/06/2021**	**Version: 1.0**	
USE CASE NAME:	**Update Allocated Team**	**USE CASE TYPE**	
USE CASE ID:	**4.6**	**Business Requirements:** ☐	
PRIORITY:	High	**System Analysis:** o	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	Operations Manager (PBA)		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	• none		
OTHER INTERESTED STAKEHOLDERS:	• Director • General employees • Site Managers • Human resource Management • Health and Safety Officer • Client		
DESCRIPTION:	This use case describes the process the Operations manager and system will go through when the Operations Manager wishes to removes a team from the Project due to unforeseen circumstances.		
PRE-CONDITION:	• Team must be allocated to an existing and active project. • Operations Manager must be logged into the system.		
TRIGGER:	The Operations Manager wants to remove a team from construction project.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The Operations Manager wants to edit the allocated team to the		

<table>
<tr><td></td><td>Construction Site.</td><td></td><td></td></tr>
<tr><td></td><td>Step 2: The Operations Manager Hovers over the Construction Menu Strip.</td><td></td><td>Step 3: The System displays the dropdown with the following items:
Projects
Construction Sites
Allocated teams
Compliance Documents
Radio Frequency reports
Snags
Acceptance Documents
Site Sign-offs
Commission reports
Site types</td></tr>
<tr><td></td><td>Step 4: The Operations Manager clicks on the “Allocated Teams” dropdown menu item.</td><td></td><td>Step 5: The system invokes use case 4.5 “Search Allocated Team”</td></tr>
<tr><td></td><td>Step 6: The Operations Manager clicks the “Edit” Button displayed on the searched for team.</td><td></td><td>Step 7: The elements displayed on the Searched for team are displayed and all inputs where applicable are enabled for editing. The following elements are displayed:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Edit Team</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Team Name</td><td>Label</td><td>None</td></tr>
<tr><td></td><td>Dropdown input (enabled)</td><td>Dropdown displaying all the teams available for the Operations Manager to select. Populated from the team object returned.</td></tr>
<tr><td>Team ID</td><td>Label</td><td>None</td></tr>
</table></td></tr>
</table>

			Team ID	Input (disabled)	Input displaying the Team ID correlating to the team selected in the previous item selected in the dropdown menu.
			Project Name	Label	None
				Dropdown (enabled)	Dropdown input displaying the list of projects available to allocate the team to.
			Table:	table	Table displaying all the employees in the selected team.
			Employee ID	label	Table Element: displaying the Employees ID
			Employee Name	Label	Table Element: Displaying the Employees Name
			Employee Surname	Label	Table Element: Displaying the Employees Surname
			Employee Role	Label	Table Element: Displaying the Employees Role within the team.
			Confirm Edit	Button	The user clicks this button begin the

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td></td><td></td><td>process to finalize the new details entered on the allocated team.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>This button is clicked to cancel the process of editing the allocated team.</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 8: The Operations Manager updates the allocated team details.</td><td></td><td></td></tr>
<tr><td></td><td>Step 9: The Operations Manager clicks the “Confirm Edit” button displayed. [ALT]</td><td></td><td>Step 10: The system will capture and validate the entered search criteria that was typed into the search textbox. The field is validated and displays a green border around the textbox. [ALT]</td></tr>
<tr><td></td><td></td><td></td><td>Step 11: The system displays a modal to the Operations Manager to confirm the editing of the allocated team details.
The modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Confirmation required</td><td>heading</td><td>Modal heading</td></tr>
<tr><td>Edit the Allocated team details?</td><td>Paragraph</td><td>Queries the user to confirm to edit the allocated team.</td></tr>
<tr><td>Yes</td><td>Button</td><td>Clicked to confirm the edit of the Allocated team.</td></tr>
<tr><td>No</td><td>Button</td><td>Clicked to reject the edit of the</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td><table><tr><td></td><td></td><td>Allocated team.</td></tr></table></td></tr>
<tr><td></td><td>Step 12: The Operations Manager clicks the “yes Button. [ALT]</td><td></td><td>Step 13: The system updates the allocated team details in the Team Construction Site table. The Details are updated in the API via an HttpPut request made in the TypeScript Component. The following details are captured in the Team Construction Site table:
Team ID
Construction Site ID
Date Assigned</td></tr>
<tr><td></td><td></td><td></td><td>Step 14: System displays a modal to the Operations manager indicating the allocated team details have been edited. The modal contains the following elements:<table><tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr><tr><td>Allocated team details have been successfully updated.</td><td>heading</td><td>Modal heading</td></tr><tr><td>Ok</td><td>Button</td><td>None</td></tr></table></td></tr>
<tr><td></td><td>Step 15: The Operations manager clicks the “Ok” button.</td><td></td><td>Step 16: The System generates an email and SMS that indicates the team has been assigned to the construction site.</td></tr>
<tr><td></td><td></td><td></td><td>Step 17: The system reads the employee email address and number attribute stored in the Employee table to be able to send the email and SMS to the appropriate employees. The following employees are emailed the generated email:
Health and Safety Officer
Site Manager (obtained from the newly assigned team to the construction site)</td></tr>
<tr><td></td><td></td><td></td><td>Step 18: The system displays a modal to the operations manager</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>indicating that the email has been sent to the appropriate employees. The modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>The health and safety officer and Site manager have been notified on the newly edited team details to the construction site.</td><td>Modal Heading</td><td>Heading of the modal indicating the modal purpose.</td></tr>
<tr><td>Employee Name</td><td>Label</td><td>Label indicating the Health and Safety Officer name.</td></tr>
<tr><td>Employee name</td><td>Label</td><td>Label indicating the Site Managers name.</td></tr>
<tr><td>Ok</td><td>Button</td><td>None</td></tr>
</table></td></tr>
<tr><td></td><td>Step 19: The Operations Manager clicks the “ok” button displayed.</td><td></td><td></td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 9: The Operations manager clicks the “Cancel” Button. System displays the Construction Sites page.
The Use case is terminated</td></tr>
<tr><td></td><td colspan="3">ALT Step 10: The entered information is invalid; the system displays a red border around the incorrect input fields indicating that the entered data is invalid.
Return to step 8</td></tr>
<tr><td></td><td colspan="3">ALT Step 12: The Operations Manage clicks the “No” button displayed on the modal.
Return to step 7</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">This sue case is concluded when the Health and Safety Officer and Site Manager have been notified on the changes in allocated teams.</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">The Allocated team details have been updated.</td></tr>
</table>

BUSINESS RULES:	• Only the Operations Manager can update allocated teams to the construction site.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.4.7 Upload Compliance Document

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Michael Vosloo</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Upload Compliance Document</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">4.7</td><td rowspan="3">Business Requirements: ☐
System Analysis: o
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Client (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">Health and Safety Officer (PSA)</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">• None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• Operations Manager
• Director
• Procurement and logistics officer
• Human resource management</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case narrative describes the process the system will go through when the Health and Safety Officer wishes to upload the various Compliance documents unique to the specific construction project.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• Client needs to send Health and Safety Officer compliance documents needed for the construction site, throughout the life cycle of the construction process.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">Client sends Health and Safety Officer the compliance documents needed for the active construction site.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td>Step 1: The client sends the Health and Safety Officer the compliance document via</td><td>Step 2: The Health and Safety Officer receives the compliance document</td><td></td></tr>
</table>

<table>
<tr><td></td><td>email for the project.</td><td>via email from the client.</td><td></td></tr>
<tr><td></td><td></td><td>Step 3: The Health and Safety Officer wants to upload the Compliance document to the system for the project.</td><td></td></tr>
<tr><td></td><td></td><td>Step 4: The Health and Safety Officer hovers over Construction menu strip.</td><td>Step 5: The system invokes use case 4.9 “Search Project”</td></tr>
<tr><td></td><td></td><td>Step 6: The Health and Safety Officer clicks on the “Compliance Document” card displayed on the project page.</td><td>Step 7: The System displays the Compliance Documents page with a list of compliance documents specific to the searched for project. The page contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Compliance Documents</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Search</td><td>Label</td><td>This is the label for the search filter function.</td></tr>
<tr><td>keyword</td><td>Textbox</td><td>To enter a Compliance Documents name to search for.</td></tr>
<tr><td>Search</td><td>Button</td><td>This button will filter the table for the Searched for Compliance Document details in the table.</td></tr>
<tr><td>Table</td><td>Table</td><td>This table will hold the information about the Compliance</td></tr>
</table></td></tr>
</table>

		Document Details.
Compliance Document ID	label	Part of the table: Displays the Compliance documents ID
Compliance Document Name	label	Part of the table: Displays the Compliance documents name
Date Uploaded	label	Part of the table: Displays the date the compliance document was uploaded.
Project Name	label	Part of the table: Displays the project name that the compliance document is associated to.
Upload	button	This button is clicked when the user wants to upload a new compliance document.

This page will display a table of all the Compliance documents stored on the system specific to the data that was obtained in the Search project use case invoke. The table data is read from the **Compliance Document** table. The data obtained and displayed attributes from the tables that are prevalent to the table listed above are:

- Compliance Document ID
- Compliance Document Name
- Date uploaded
- Project Name (inner joined via a foreign key)

<table>
<tr><td></td><td></td><td></td><td>The table attributes are read via an HttpGet method in the Typescript component and an API request.
[ALT]</td></tr>
<tr><td></td><td></td><td>Step 8: The Health and Safety Officer clicks the “Upload” button to upload a new compliance Document.</td><td>Step 9: The system displays the “Upload Compliance Document” page with the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Upload Compliance Document</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Name</td><td>Label</td><td>None</td></tr>
<tr><td></td><td>input</td><td>This input is used to capture the Compliance document name.</td></tr>
<tr><td>Description</td><td>Label</td><td>None</td></tr>
<tr><td></td><td>Input</td><td>This input is used to capture the Compliance document description.</td></tr>
<tr><td>Compliance Document</td><td>Label</td><td>Displays to the user that the following button after this label will be clicked to capture the compliance document</td></tr>
<tr><td>Upload</td><td>button</td><td>None</td></tr>
<tr><td>Project Name</td><td>Label</td><td>None</td></tr>
<tr><td></td><td>Input (disabled)</td><td>Auto selects the project Name as the search project use case was invoked.</td></tr>
<tr><td>Create</td><td>Button (disabled)</td><td>This button is clicked to create the new compliance Document.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>This button is clicked to</td></tr>
</table></td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td></td><td></td><td>cancel the creation of the new compliance document</td></tr>
</table>
</td></tr>
<tr><td></td><td></td><td>Step 10: The Health and Safety Officer enters the compliance document information in the correct inputs</td><td>Step 11: The system will capture and validate the entered information. The field is validated and displays a green border around the textbox.
Create Button is enabled.
[ALT]</td></tr>
<tr><td></td><td></td><td>Step 12: The health and safety Officer clicks the “Create” button displayed. [ALT]</td><td>Step 13: The system displays a confirmation modal to the Health and Safety Officer to confirm to upload the compliance document. The modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Confirmation required</td><td>heading</td><td>Modal heading</td></tr>
<tr><td>Are you sure you want to upload the Compliance Document?</td><td>Paragraph</td><td>Queries the user to confirm to edit the allocated team.</td></tr>
<tr><td>Yes</td><td>Button</td><td>Clicked to confirm to upload the compliance documents</td></tr>
<tr><td>No</td><td>Button</td><td>Clicked to reject the upload of the compliance document.</td></tr>
</table>
</td></tr>
<tr><td></td><td></td><td>Step 14: The Health and Safety Officer clicks the “yes” button displayed. [ALT]</td><td>Step 15: The system uses an SQL insert query performed via an HttpPost request made within the Typescript component to create a new entry within the Compliance Document table. The system will enter the following attributes obtained from the entered inputs to create the Compliance document.
Compliance Document table:</td></tr>
</table>

<table>
<tr><td rowspan="3"></td><td></td><td></td><td>

Compliance Document ID
Project ID
Name
Description
Date uploaded (takes the current date the use case takes place on)

Compliance Document
</td></tr>
<tr><td></td><td></td><td>
Step 16: The system displays a modal to the Health and Safety Officer indicating the Compliance document has been saved to the system. The modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Compliance Document has been successfully uploaded to the system</td><td>heading</td><td>Modal heading</td></tr>
<tr><td>ok</td><td>Button</td><td>None</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 17: The Health and Safety Officer clicks the “Ok” button.</td><td></td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 7: The system reads that no Compliance documents have been associated with the Project.
Skip to step 9</td></tr>
<tr><td></td><td colspan="3">ALT Step 11: The entered information is invalid; the system displays a red border around the incorrect input fields indicating that the entered data is invalid.
Return to step 10</td></tr>
<tr><td></td><td colspan="3">ALT Step 12: The Health and Safety Officer clicks the “No” button.
The Use case is terminated</td></tr>
<tr><td></td><td colspan="3">ALT Step 14: The Health and Safety Officer clicks the “No” button displayed.
Return to step 9</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">This use case is concluded when the Health and Safety Officer is notified on the success of the compliance document being uploaded.</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">The construction site compliance document has been uploaded to the Compliance document table.</td></tr>
</table>

BUSINESS RULES:	• Only Health and safety Officer may upload compliance documents for the construction site. • Each construction site will have unique compliance documents relating specifically to that construction site. • The Client (Vodacom) is responsible for creating the compliance documents and submitting them to the Health and Safety Officer. • Compliance documents are not always needed for construction projects.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.4.8 Search Compliance Document

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Michael Vosloo</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Search Compliance Document</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">4.8</td><td rowspan="3">Business Requirements: ☐
System Analysis: o
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Medium</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">User (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• Operations Manager
• Director
• Client
• Quality Control Officer
• Health and Safety Officer</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case entails the process the system will go through when searching for a specific Compliance Document that has been captured and stored on the system relating to the project associated to the Compliance Document. The User will enter the desired search criteria into the system and the corresponding Compliance Document matching the search criteria will be displayed to the user.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">The User is on the specific project page.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The user wants to search for a Compliance Document.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The user wants to search for a compliance document.</td><td>Step 2: The user hovers over the Construction menu strip.</td><td>Step 3: The system will display the dropdown with the following items:
• Projects
• Construction Sites
• Allocated teams</td></tr>
</table>

<table>
<tr><td rowspan="2"></td><td></td><td></td><td>

- Compliance Documents
- Radio Frequency reports
- Snags
- Acceptance Documents
- Site sign-offs
- Commission reports
- Site types

</td></tr>
<tr><td></td><td>Step 4: The user clicks on the “Compliance Documents” dropdown menu strip. [ALT]</td><td>Step 5: The system displays the Search Compliance Documents page with the following items:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Compliance Documents</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Search</td><td>Label</td><td>This is the label for the search filter function.</td></tr>
<tr><td>keyword</td><td>Textbox</td><td>To enter a Compliance Documents name to search for.</td></tr>
<tr><td>Search</td><td>Button</td><td>This button will filter the table for the Searched for Compliance Document details in the table.</td></tr>
<tr><td>Table</td><td>Table</td><td>This table will hold the information about the Project Details.</td></tr>
<tr><td>Compliance Document ID</td><td>label</td><td>Part of the table: Displays the Compliance documents ID</td></tr>
<tr><td>Compliance Document Name</td><td>label</td><td>Part of the table: Displays the Compliance documents name</td></tr>
<tr><td>Date Uploaded</td><td>label</td><td>Part of the table: Displays</td></tr>
</table>
</td></tr>
</table>

<table>
<tr>
<td></td>
<td></td>
<td></td>
<td>
<table>
<tr><td></td><td></td><td>the date the compliance document was uploaded.</td></tr>
<tr><td>Project Name</td><td>label</td><td>Part of the table: Displays the project name that the compliance document is associated to.</td></tr>
<tr><td>Upload</td><td>button</td><td>This button is clicked when the user wants to upload a new compliance document.</td></tr>
</table>

This page will display a table of all the compliance documents that are stored within the system. The table data is read from the Compliance document and project tables. The data obtained and displayed attributes that are prevalent to the table listed above are:
Compliance Document Table:

Compliance Document ID
Name
Date Uploaded

Project Table:

Project Name

The table attributes are read Via an HttpGet method in the Typescript component and an API request.
</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Step 6: The user enters the Compliance Documents search criteria in the</td>
<td>Step 7: The system will capture and validate the entered information that was typed into the search textbox. The field is validated and displays a green border around the textbox. [ALT]</td>
</tr>
</table>

<table>
<tr><td></td><td></td><td>search text box. [ALT]</td><td></td></tr>
<tr><td></td><td></td><td></td><td>Step 8: The system will submit the captured search criteria and search for the Compliance document details in the Compliance Document table using a parametrised HttpGet request in the typescript component and API request. The system captures and saves the searched result to a list object and displays the search result in the table components, replacing the previous Compliance document table items as they do not follow the searched criteria. [ALT]</td></tr>
<tr><td></td><td></td><td>Step 9: The user clicks on the desired Compliance document details displayed on the table.</td><td>Step 10: The system displays the Compliance Document page that is attributed by the Compliance Document and Project tables according to the searched for Compliance document details. The Compliance Document details are read from the Compliance Document and Project tables and passed via the API control to the type script component to then be displayed on the Compliance document page elements. The Compliance document page has the following elements displayed:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Compliance Document</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Name</td><td>Label</td><td>Displays the compliance documents name.</td></tr>
<tr><td>Description</td><td>Label</td><td>Displays the description of the compliance document.</td></tr>
<tr><td>Compliance Document</td><td>Label</td><td>Displays to the user</td></tr>
</table>
</td></tr>
</table>

					that the following button after this label will display the compliance document.
			View	button	When this button is clicked the system will display the compliance document to the user.
			Project Name	Label	Displays the project name that the compliance document is linked to.
			Date Uploaded	label	Displays the date the compliance document was uploaded.
ALTERNATE COURSES:	**ALT Step 4**: The user selects the "Compliance Document" element on the "Project" page, the system will **skip to step 5** but only display the compliance documents associated to that project that was being displayed. The compliance document is linked to the project which entails that the **Compliance document** and **Project** tables have respective foreign keys within the **Compliance documents** table. The compliance documents are retrieved via an inner join function in SQL for matching primary keys and the data is retrieved via an API and typescript component. **Skip to step 5**				
	ALT Step 6: The user selects a Compliance Document from the displayed list present on the screen. Skip to step 9:				
	ALT Step 7: The entered search criteria is invalid; the system displays a red border around the search textbox indicating that the entered search criteria is invalid. **Return to step 6**				
	ALT Step 8: The search criteria entered into the search textbox does not match any existing records in the database. The system notifies and displays a modal to the user to indicate that the searched for compliance document details do not exist within the database as a null value was obtained from the HttpGet request.				

	Return to step 6
CONCLUSION:	This use called is concluded when the compliance document page is displayed to the user.
POST-CONDITION:	The correct compliance document is displayed to the user.
BUSINESS RULES:	All users can search for a compliance document linked to a project.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.4.9 Search Project

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s): Michael Vosloo</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Search Project</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">4.9</td><td rowspan="3">Business Requirements:☐
System Analysis: o
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">User (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case entails the process the system will go through when the user requires to search for a specific project that is currently active on the system. The user will enter the details of the project for the system to then read the details of the project from the associated tables.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• User must be logged onto the system
• The project must exist on the system to view.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The user wants to search for a project.</td></tr>
<tr><td rowspan="3">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The user wants to search for a project.</td><td></td><td></td></tr>
<tr><td>Step 2: The user hovers over the Construction menu strip.</td><td></td><td>Step 3: The system will display the dropdown with the following items:
• Projects
• Construction Sites
• Allocated teams
• Compliance Documents</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>

- Radio Frequency reports
- Snags
- Acceptance Documents
- Site sign-offs
- Commission reports
- Site types

</td></tr>
<tr><td></td><td>Step 4: The user clicks on the “Projects” dropdown menu item.</td><td></td><td>Step 5: The system displays the Search Project page with the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Projects</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Search</td><td>Label</td><td>None</td></tr>
<tr><td>keyword</td><td>Textbox</td><td>To enter a projects name to search for.</td></tr>
<tr><td>Search</td><td>Button</td><td>This button will filter the table for the Searched for the Project details in the table.</td></tr>
<tr><td>Table</td><td>Table</td><td>This table will hold the information about the Project Details.</td></tr>
<tr><td>Site number</td><td>Label</td><td>Part of the table: displays the Projects site number.</td></tr>
<tr><td>Project name</td><td>Label</td><td>Part of the table: Displays the Projects name.</td></tr>
<tr><td>Site location</td><td>Label</td><td>Part of the table: Displays the Projects location</td></tr>
<tr><td>Site type</td><td>Label</td><td>Part of the table: Displays the Projects site type</td></tr>
<tr><td>Date started</td><td>Label</td><td>Part of the table: Displays the date the project started.</td></tr>
<tr><td>Status</td><td>Label</td><td>Part of the table: Displays the current status of the project.</td></tr>
<tr><td>Status progress</td><td>Progress bar</td><td>Part of the table: Displays the progress the project has made in a progress bar view for a visual representation.</td></tr>
<tr><td>New Project</td><td>Button</td><td>When this button is clicked a new</td></tr>
</table>
</td></tr>
</table>

			(+) | | Technical Site survey process will begin and the user will be redirected to the Technical Site survey subsystem. This page will display a list of all the currently active and existing projects stored on the system. The list is read from the stored project entries on the **Project** table in the database and the relevant attributes previously mentioned are populated with the following table attributes: • Project Name • Date started • Site Number • Site location • Region • Project Status ID (inner joined with the **Project_Status** table) The table attributes are done via an HttpGet method in the Typescript component and an API request.
	Step 6: The user enters the project name as a search criteria in the search text box. **[ALT]**		**Step 7:** The system will capture and validate the entered search criteria that was typed into the search textbox. The field is validated and displays a green border around the textbox. **[ALT]**
			Step 8: The system will submit the captured search criteria and search for the project in the **Project** table using a parametrised HttpGet request in the typescript component and API request. The system captures and saves the searched result to a list object and displays the search result in the list components, replacing the previous project list items as they do not follow the searched criteria. **[ALT]**
	Step 9: The user clicks on the list object displayed via		**Step 10:** The system displays Project page that is attributed by the Project table according to the searched for project details, the Project information

<table>
<tr>
<td></td>
<td>the list component.
[ALT]</td>
<td></td>
<td>is read from the Project table and passed via the API controller to the type script component to be displayed on the project page elements. The Project page has the following elements displayed:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Project Name</td><td>label</td><td>Displays the project name</td></tr>
<tr><td>Site Number</td><td>label</td><td>Displays the Site number</td></tr>
<tr><td>Location</td><td>label</td><td>Displays the location of the project</td></tr>
<tr><td>Region</td><td>Label</td><td>Displays the Region the project is within.</td></tr>
<tr><td>Date started</td><td>label</td><td>To indicate when the project was started.</td></tr>
<tr><td>Actual date</td><td>Date</td><td>The date the project was started.</td></tr>
<tr><td>Last updated</td><td>Label</td><td>To differentiate the date of when the project status was last changed.</td></tr>
<tr><td>Actual date</td><td>Date</td><td>Displays the actual date the project was last updated.</td></tr>
<tr><td>Status</td><td>Label</td><td>Indicating the current status of the project</td></tr>
<tr><td>Technical Site Survey</td><td>Card</td><td>When clicked the user will be directed to the Technical Site Survey page.</td></tr>
<tr><td>Formal Drawing</td><td>Card</td><td>When clicked the user will be directed to the Formal Drawing page.</td></tr>
<tr><td>Second Cut Drawing</td><td>Card</td><td>When clicked the user will be directed to the Second Cut Drawing page.</td></tr>
<tr><td>Allocated Teams</td><td>Card</td><td>When click the user will be directed to the</td></tr>
</table>
</td>
</tr>
</table>

					Allocated teams page
			Quotations	Card	When clicked the user will be directed to the Quotations page.
			Purchase Orders	Card	When clicked the user will be directed to the Purchase Orders page.
			Site-Go-Ahead	Card	When clicked the user will be directed to the Site-Go-Ahead page.
			Health and Safety File	Card	When clicked the user will be directed to the Health and Safety File page.
			Compliance Documents	Card	When clicked the user will be directed to the Compliance Documents page.
			Construction Progress description	Card	When clicked the user will be directed to the Construction Progress description page.
			Complete Site Photos	Card	When clicked the user will be directed to the Complete Site Photos page.
			Radio Frequency report	Card	When clicked the user will be directed to the Radio Frequency report page.
			Request Site-Visit	Card	When clicked the user will be directed to the Request Site-Visit page.
			Site Acceptance Document	Card	When clicked the user will be directed to the Acceptance Document page.
			Invoices	Card	When clicked the user will be directed to the Invoices page.

			Commission report	Card	When clicked the user will be directed to the Commission report page.
			As-built	Card	When clicked the user will be directed to the As-Built page.
			Site Sign-off	Card	When clicked the user will be directed to the Site Sign-off page.

ALTERNATE COURSES:	ALT Step 6: The user selects a project from the displayed list present on screen. **Skip to step 9**
	ALT step 6: The entered information is invalid; the system displays a red border around the search textbox indicating that the entered search criteria is invalid. **Return to step 6**
	ALT Step 8: The search criteria entered into the search textbox does not match any existing records in the database. The system notifies and displays a modal to the user to indicate that the searched for project details do not exist within the database as a null value was obtained from the HttpGet request. **Return to step 6**
	ALT Step 9: The Operations manager clicks on the “Add (+)” button. The system invokes use case “1.1 Create TSS”
CONCLUSION:	This use case concludes when the project page is displayed to the user.
POST-CONDITION:	The Project page is displayed to the user with the correct project details
BUSINESS RULES:	All users can search for project details.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.4.10 Create Site Progress description

Gohvan Construction			
Author (s): **Michael Vosloo**	**Date: 28/06/2021**	**Version: 1.0**	
USE CASE NAME:	**Create Site Progress description**	**USE CASE TYPE**	
USE CASE ID:	**4.10**	**Business Requirements:** ☐	
PRIORITY:	High	**System Analysis:** o	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	Site Manager (PBA)		
PRIMARY THE SYSTEM ACTOR:	Operations Manager (PSA)		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	• Client • Director • Construction Team • Health and Safety Officer		
DESCRIPTION:	This use case describes the process that the system and operations manager will go through when the construction site progress is to be updated. The site Manager will update the operations manager on the progress of the site via a call and photos. The Operations Manager will then create a site progress description entry in the system to keep track of the progress made on the construction site.		
PRE-CONDITION:	• The Site Manager updates the Operations Manager on progress made to the construction site. • Photos on the construction site are sent to the manager		
TRIGGER:	The Operations Manager requests to update and create a new entry on the construction site progress.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The Site Manager updates the	**Step 2:** The Operations Manager receives the	

<table>
<tr><td></td><td>Operations Manager on the progress made on the constructio n site.</td><td>information from the Site Manager.</td><td></td></tr>
<tr><td></td><td></td><td>Step 3: The Operations Manager wants to create a new entry for the site progress description.</td><td></td></tr>
<tr><td></td><td></td><td>Step 4: The Operations Manager hovers over the Construction menu strip.</td><td>Step 5: The system invokes use case 4.11 “View Site Progress description”</td></tr>
<tr><td></td><td></td><td>Step 6: The Operations Manager clicks on the “Create new Site Progress description” button displayed.</td><td>Step 7: The system displays the Create Site Progress description page with the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Create Site Progress Description</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Site Name</td><td>Label</td><td>Displays the Project the Site progress description will be associated to.</td></tr>
<tr><td>Start Date</td><td>Label</td><td>Contains the date the Construction site was started.</td></tr>
<tr><td>Last updated</td><td>Label</td><td>Contains the previous date the construction site was updated.</td></tr>
<tr><td>Title</td><td>Label</td><td>None</td></tr>
<tr><td></td><td>Input</td><td>User inputs the Title of</td></tr>
</table></td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td></td><td></td><td>the site progress.</td></tr>
<tr><td>Description</td><td>Label</td><td>None</td></tr>
<tr><td></td><td>Input</td><td>User inputs the description of the site progress.</td></tr>
<tr><td>Attach Progress Photos</td><td>Button</td><td>User clicks this button to upload the Site progress description photos.</td></tr>
<tr><td>Submit</td><td>Button (Disabled)</td><td>User clicks this button to create the site progress description.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>User clicks this button to end the process of creating a site progress description.</td></tr>
</table>
</td></tr>
<tr><td></td><td></td><td>Step 8: The Operations Manager enters the received information into the corresponding inputs.</td><td>Step 9: The system captures and validates the provided information entered into the fields.
Submit Button is enabled. [ALT]</td></tr>
<tr><td></td><td></td><td>Step 10: The Operations Manager clicks on the “Submit” button. [ALT]</td><td>Step 11: The system loads the confirmation modal and displays it to the Operations Manager to prompt the confirmation of creating the new site progress description. The Modal contains the following elements:
<table>
<tr><td>Input name:</td><td>Control Type:</td><td>Notes:</td></tr>
<tr><td>Confirmation required</td><td>heading</td><td>Modal heading</td></tr>
<tr><td>Are you sure you want to create the site progress description?</td><td>Paragraph</td><td>Queries the user to confirm to create the site progress description.</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><td rowspan="4"></td><td></td><td></td><td><table><tr><td>Yes</td><td>Button</td><td>Clicked to confirm to create the site progress description .</td></tr><tr><td>No</td><td>Button</td><td>Clicked to reject the creation of the site progress description .</td></tr></table></td></tr>
<tr><td></td><td>Step 12: The Operations Manager clicks the “Yes” Button. [ALT]</td><td>Step 13: The system saves the site progress description to the relevant tables associated. A new entry in the following tables will be created.
Site Progress Description table:
Site Progress description ID
Construction Site ID
Title
Description
Date Uploaded
Construction Photo table:
Construction Photo ID
Site Progress Description ID
Construction Photo
The following attribute within the Construction Site table will be updated:
Date Uploaded (Current date the use case is executed)</td></tr>
<tr><td></td><td></td><td>Step 15: The System displays an alert to notify on the successful creation of the site progress description.</td></tr>
<tr><td></td><td></td><td>Step 16: The system displays the View Site Progress description page to the Operations manager with the new entry visible.</td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 9: The entered information is invalid; the system displays a red border around the search textbox indicating that the entered search criteria is invalid. Return to step 8</td></tr>
</table>

	ALT Step 10: The Operations Manager clicks on the “Cancel” button. The create site progress description page is closed. **The Use case is terminated**
	ALT Step 12: The Operation Manager clicks on the “No” button. The confirmation modal is closed. **Return to step 7**
CONCLUSION:	This use case is concluded when the View Site Progress description is displayed to the Operations manager with the new Site progress description visible.
POST-CONDITION:	The new site progress description is saved in the database.
BUSINESS RULES:	• Site Manger has to provide the Operations Manager with photos on the progress made to the construction site. • Only the Operations Manager can update the site progress description on the system.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.4.11 View Site Progress description.

Gohvan Construction			
Author (s): **Michael Vosloo**	**Date: 28/06/2021**	**Version: 1.0**	
USE CASE NAME:	**View Site Progress description**	**USE CASE TYPE**	
USE CASE ID:	**4.11**	**Business Requirements:** ☐	
PRIORITY:	High	**System Analysis:** o	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	User		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	• Director • Quality control officer • Client • Procurement and Logistics Officer • Electrician • Operations Manager		
DESCRIPTION:	This use case describes the process the system will go through when primarily the Operations manager or any interested stake holder would like to view the progression of the construction site relating to a specific project.		
PRE-CONDITION:	• The user must be logged into the system. • A construction site must have Site-Go-Ahead		
TRIGGER:	The user wishes to view site progress.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The user wants to view the construction site progress status.	**Step 2:** The user Hovers over the Construction menu strip.	**Step 3:** The system will display the dropdown with the following items: • Projects • Construction Sites

			• Allocated teams • Compliance Documents • Radio Frequency reports • Snags • Acceptance Documents • Site sign-offs • Commission reports • Site types
		Step 4: The user clicks on the "Projects" dropdown menu item.	**Step 5:** The system invokes use case 4.9 "Search Project"
		Step 6: The user clicks on the "Construction Progress Description" card.	**Step 7:** The system will capture the data within the **Site Progress description** and **Construction Photo** tables to then be displayed on the Site progress description page. The **Project** is linked to the **Site Progress description** and **Construction Photo** tables via a foreign project ID key entered into the **Construction Site** table and then the matching Construction Site ID primary key that is entered into the **Site Progress description** table as a foreign key. The tables are joined via inner join function within SQL to obtain the relative foreign keys mentioned. The Project ID is submitted via a Typescript component and API HttpGet controller method. The data being obtained from the aforementioned tables are as follows: **Site Progress Description** table: • Title • Description • Date Uploaded **Construction Photo** table: • Construction photo The aforementioned tables are read via a sequence of inner joins to obtain all the data entries

<table>
<tr><td></td><td></td><td></td><td>associated to the Site Progress Description and Construction Photo tables, the data is captured and stored within an object array. [ALT]</td></tr>
<tr><td></td><td></td><td></td><td>Step 8: The Site Progress Description page is displayed to the user and the page elements are populated via the data captured in the previous step. The displayed elements are as follows:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Site Progress Description</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Title</td><td>Label</td><td>The title of the site progress description is displayed to the user.</td></tr>
<tr><td>Date uploaded</td><td>Label</td><td>The date the Site progress description was uploaded it displayed to the user.</td></tr>
<tr><td>Description</td><td>Paragraph</td><td>The description for the Site progress status is displayed to the user.</td></tr>
<tr><td>Construction photos</td><td>Photo</td><td>The Progress site photos that relate to the site progress description are displayed to the user.</td></tr>
<tr><td>Create new Site progress description</td><td>Button</td><td>This button is clicked when the user wants</td></tr>
</table></td></tr>
</table>

					to create a new site progress description for the Project.

The elements are repeated according the number of objects that were stored in the captured objects array in the previous step.

ALTERNATE COURSES:	**ALT Step 7:** The system captures a null return on the API request thus there is no data entries matching the searched for parameters of the Site progress description. The system displays a modal notification to the user that there are no Site progress description entries made for the relating Construction Site. The modal prompt queries the user if they want to create a new site progress description. The modal will display the following elements:

Input name:	Control Type:	Notes:
No Site Progress descriptions found.	Heading	The Modal heading indicating there were no found Site Progress description.
Enter initial Site Progress descriptions.	Label	Prompting the user to enter an initial site progress description.
Yes	Button	If clicked on the user will be redirected to use case 4.10 “Create Site Progress description”
No	Button	If clicked on the user will be redirected to the Project page.

The use case is concluded.

CONCLUSION:	This use case concludes when the Site Progress description page is displayed.
POST-CONDITION:	The correct Site Progress description page is displayed to the user.
BUSINESS RULES:	Must be at least one entry in the Site Progress description table to view the Site Progress description page.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.4.12 Upload complete site photos

Gohvan Construction			
Author (s): **Michael Vosloo**	**Date: 28/06/2021**	**Version: 1.0**	
USE CASE NAME:	**Upload complete site photos**	**USE CASE TYPE**	
USE CASE ID:	**4.12**	**Business Requirements:** ☐	
PRIORITY:	High	**System Analysis:** o	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	Site Manager (PBA)		
PRIMARY THE SYSTEM ACTOR:	Operations Manager (PSA)		
OTHER PARTICIPATING ACTORS:	• Client • Health and safety Officer		
OTHER INTERESTED STAKEHOLDERS:	• Director • Quality control officer • suppliers • Procurement and Logistics Officer		
DESCRIPTION:	This use case describes the process the system will go through once the construction is fully complete on the construction site. The Operations manager will take photos of the fully completed site and upload it into the system. Once the final photos for the completed site are uploaded to the system the Site progress description will automatically be updated to “complete” and the use case for uploading of the Radio Frequency report (RF report) will be activated for the user to upload said report.		
PRE-CONDITION:	• The operations Manager must be logged into the system • The site Manager must let the Operations manager know that site construction has been completed.		
TRIGGER:	The Operations Manager receives a notification from the Site Manager that construction has been completed.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The Site Manager sends the Operations	**Step 2:** The Operations manager receives the	**Step 3:** The system invokes use case 4.9 “Search Project”

<table>
<tr><td></td><td>Manager the completed site photos</td><td>Complete Site photos and wants to upload the Complete site photos to the system.</td><td></td></tr>
<tr><td></td><td></td><td>Step 4: The Operations Manager clicks the “Complete Site Photos” card displayed.</td><td>Step 5: The system reads from the Completion Photo to determine if there are no complete site photos uploaded to the table relating to the searched project. [ALT]</td></tr>
<tr><td></td><td></td><td></td><td>Step 6: The system displays a modal to the Operations Manager prompting to begin the process to upload Complete site photos for the project. The modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>No Complete site photos detected.</td><td>heading</td><td>Modal heading</td></tr>
<tr><td>Upload complete site photos?</td><td>Paragraph</td><td>Queries the user to upload complete site photos.</td></tr>
<tr><td>Yes</td><td>Button</td><td>Clicked to confirm to upload complete site photos.</td></tr>
<tr><td>No</td><td>Button</td><td>Clicked to reject the uploading of complete site photos.</td></tr>
</table></td></tr>
<tr><td></td><td></td><td>Step 7: The Operations Manager clicks the “Yes” button</td><td>Step 8: The system displays a modal to the user to select or take the Complete site photos to the system. The modal contains the following elements:</td></tr>
</table>

<table>
<tr>
<td></td>
<td></td>
<td>displayed.
[ALT]</td>
<td>
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Select Complete Site Photos.</td><td>Paragraph</td><td>Queries the user to confirm to upload the Radio Frequency report.</td></tr>
<tr><td>Select</td><td>Button</td><td>Clicked to select photos stored in gallery/folder</td></tr>
<tr><td>Camera</td><td>Button</td><td>Selected to use the camera on the device being used.</td></tr>
<tr><td>Photos</td><td>Photo</td><td>These elements will be repeated for the amount of selected photos to be uploaded to the system when the photos are selected.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>Clicked to cancel the process of uploading the complete site photos</td></tr>
<tr><td>Upload</td><td>Button (disabled)</td><td>Clicked to upload the selected photos to the system.</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Step 9: The Operations Manager clicks the “Select” button displayed.
[ALT]</td>
<td>Step 10: The system opens the devices gallery/folder for the user to be able to select the desired complete site photos to be uploaded to the system.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Step 11: The Operations Manager selects the complete site photos.</td>
<td>Step 12: The system captures and validates the selected photos to be stored on the system.
Upload Button enabled.</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td></td>
<td>Step 13: The Operations manager clicks the “Upload” button displayed. [ALT]</td>
<td>Step 14: The system displays a confirmation modal to the Operations manager to query to upload the selected complete site photos to the system. The modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Confirmation required</td><td>heading</td><td>Modal heading</td></tr>
<tr><td>Upload complete site photos?</td><td>Paragraph</td><td>Queries the user to upload complete site photos.</td></tr>
<tr><td>Yes</td><td>Button</td><td>Clicked to confirm to upload complete site photos.</td></tr>
<tr><td>No</td><td>Button</td><td>Clicked to reject the uploading of complete site photos.</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Step 15: The Operations manager clicks the “yes” button displayed. [ALT]</td>
<td>Step 16: The system uploads and creates multiple entries in the Completion Photo table according to how many photos were selected to be uploaded. The system captures and stores the photos in the Completion photo table via an HttpPost method called within the Typescript component. The following attributes are entered in the Completion photo table:

Completion Photo ID
Project ID
Completion Photo
Date uploaded

</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Step 17: The system updates the Project Status ID in the Project table to indicate that the Complete Site photos have been uploaded by the Operations</td>
</tr>
</table>

			Manager. The following attributes are updated in the Project table: • Project Status ID • Date Updated (takes the current date the Use case is executed.)
			Step 18: The System generates an email that will notify the Health and Safety Officer that the Complete site photos have been uploaded.
			Step 19: The system reads the employee email address attribute stored in the Employee table to be able to send the email to the appropriate employee. The following employees are emailed the generated email: • Health and Safety Officer
			Step 20: The system displays a notification to the Operations manager indicating the Site photos have been successfully uploaded.
ALTERNATE COURSES:	**ALT Step 5:** The System reads that there are compete site photos stored in the **Completion Photo** table. The system directs the user to the View Complete Site Photos modal. **The use case is terminated**		
	ALT Step 7: The Operations manager clicks the “No” button displayed, indicating the Operations manager does not want to upload the complete site photos to the system. **The use case is terminated**		
	ALT Step 9: The Operations manager selects the “Camera” button displayed indicating he wants to take the photos of the competed site directly. The system opens and uses the camera app of the device being used. **Skip to step 11**		
	ALT Step 13: The Operations manager selects the “Cancel” button displayed, indicating the Operations manager does not want to upload the completed site photos to the system.		
	ALT Step 14: The Operations Manager clicks the “No” button displayed, indicating the operations manager does not want to upload the selected site photos displayed. **Return to step 8**		
CONCLUSION:	This use case is concluded when the Operations manager is notified that the Complete site photos have been successfully uploaded to the system.		

POST-CONDITION:	• The completion construction site photos have been uploaded to the **Completion Photo** table
BUSINESS RULES:	• Operations Manager can either take photos personally or upload phots from an existing gallery.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.4.13 Search complete site photos

Gohvan Construction			
Author (s): **Michael Vosloo**	**Date: 28/06/2021**		**Version: 1.0**
USE CASE NAME:	**Search complete site photos**		**USE CASE TYPE**
USE CASE ID:	**4.13**		**Business Requirements:** ☐
PRIORITY:	Medium		**System Analysis:** o
SOURCE:	Gohvan Construction System		**System Design:** ☑
PRIMARY BUSINESS ACTOR:	User (PBA)		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	• Operations Manager • Director • Quality Control Officer		
DESCRIPTION:	This use case entails the process the system will go through when searching for the specific completed site photos that has been captured and stored on the system relating to the project associated to the completed site photos. The User will enter the desired search criteria into the system and the corresponding completed site photos matching the search criteria will be displayed to the user.		
PRE-CONDITION:	The Project page is being displayed to the User		
TRIGGER:	The user wants to search for a completed site photo.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The Operations Manager wants to search and view the complete site photos.		**Step 2:** The System invokes use case 4.9 “Search Project”
			Step 3: The system reads from the Completion Photo to

<table>
<tr><td rowspan="3"></td><td></td><td></td><td>determine if there are complete site photos uploaded to the table relating to the searched project.
[ALT]</td></tr>
<tr><td></td><td></td><td>Step 4: The System reads the Complete site photos stored in the Completion Photo table relating to the searched project.
The attributes are read and saved into an array of object via an HttpGet method performed in the API which was called within the typescript component.
The following attributes are called and stored in the array of objects:
Completion photo ID
Completion Photo
Date Uploaded</td></tr>
<tr><td></td><td></td><td>Step 5: The system displays a modal to the user with the following elements:<table><tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr><tr><td>Complete Site Photos</td><td>heading</td><td>Modal Heading</td></tr><tr><td>Photos</td><td>Photo</td><td>These elements will be repeated for the amount of complete site photos are stored on the system</td></tr><tr><td>Close</td><td>Button</td><td>Clicked to close the modal displaying the complete Site photos</td></tr></table></td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 3: The System reads that there are no Complete site photos stored within the Completion Photo table relating to the</td></tr>
</table>

	Project. The system displays the modal to upload complete site photos. **The use case is terminated**
CONCLUSION:	This use case is concluded when the User views the Complete site photos.
POST-CONDITION:	Complete Site photos for the Project are displayed.
BUSINESS RULES:	None
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.4.14 Upload Radio Frequency report

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Michael Vosloo</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Upload Radio Frequency report</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">4.14</td><td rowspan="3">Business Requirements: ☐
System Analysis: o
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Operations Manager (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">• Financial Officer
• Client</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• Director</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the process the Operations Manager will go through once the Construction photos are uploaded to the system. The system will capture the already created Radio Frequency report and store the report in the system. Once the report has been captured and the necessary actors are notified. The ninety percent claim can then be request from the client, as well as to request the site inspection from the quality control officer.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• Site completion photos must be uploaded to the system.
• 100% completion on construction site.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The Operations Manager requests to upload the Radio Frequency report</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The Operations Manager wants to upload the Radio</td><td></td><td></td></tr>
</table>

<table>
<tr><td></td><td>Frequency report to the system.</td><td></td><td></td></tr>
<tr><td></td><td>Step 2: The Operations Manager hovers over the Construction Menu strip.</td><td></td><td>Step 3: The system invokes use case 4.9 “Search Project”</td></tr>
<tr><td></td><td>Step 4: The Operations Manager clicks on the “Radio Frequency Report” card displayed.</td><td></td><td>Step 5: The system reads from the Radio Frequency Report table to determine if there is an entry already associated with the Searched for project.
The system searched for a relating foreign key of “Project ID” in the Radio Frequency report table. [ALT]</td></tr>
<tr><td></td><td></td><td></td><td>Step 6: The system displays the Upload Radio Frequency report page to the Operations Manager. The page contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Upload Radio Frequency Report</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Name:</td><td>Label</td><td>None</td></tr>
<tr><td></td><td>Input</td><td>User enters the name of the radio frequency report.</td></tr>
<tr><td>Radio Frequency Report</td><td>Label</td><td>None</td></tr>
<tr><td>Upload</td><td>Button</td><td>User clicks this to select the Radio Frequency report to be uploaded.</td></tr>
<tr><td>Item Code</td><td>Label</td><td>None</td></tr>
<tr><td></td><td>Input</td><td>User enters the</td></tr>
</table></td></tr>
</table>

					Item Code of the cooling system.
			Serial Number	Label	None
				Input	User enters the Serial Number of the cooling system.
			Air Conditioner Size	Label	None
				Input	User enters the Air Conditioner Size of the cooling system.
			Description	Label	None
				Input	User enters the Description of the cooling system document.
			Cooling system Document	Label	None
			Upload		User clicks this to select the Cooling system document to be uploaded.
			Linked Project	Label	Displays the name of the linked project.
			Submit	button (disabled)	User clicks this to upload the

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td></td><td></td><td>Radio Frequency report.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>User clicks this to cancel the process of creating the radio frequency report</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 7: The Operations Manager enters the details for the Radio Frequency report</td><td></td><td>Step 8: The system captures and validates the provided information entered into the fields.
Submit Button is enabled. [ALT]</td></tr>
<tr><td></td><td>Step 9: The Operations Manage clicks the “Submit” button displayed. [ALT]</td><td></td><td>Step 10: The system loads the confirmation modal and displays it to the Operations Manager to prompt the confirmation of uploading the Radio Frequency report. The Modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Confirmation required</td><td>heading</td><td>Modal heading</td></tr>
<tr><td>Are you sure you want to upload the Radio Frequency report?</td><td>Paragraph</td><td>Queries the user to confirm to upload the Radio Frequency report.</td></tr>
<tr><td>Yes</td><td>Button</td><td>Clicked to confirm to upload the Radio Frequency report.</td></tr>
<tr><td>No</td><td>Button</td><td>Clicked to reject the upload of the Radio Frequency report.</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 11: The Operations Manager clicks the</td><td></td><td>Step 12: The system saves the Radio Frequency report details to the relevant tables. The following attributes will be entered and updated in the following tables:</td></tr>
</table>

	"Yes" button **[ALT]**		**Radio Frequency report** table: • Radio Frequency report ID • Construction Site ID (relating to the searched project invoke) • Name • Date Uploaded (current date the use case is executed) • Radio Frequency report The following table is updated as the **construction site** table already exists: **Construction Site** table: • Item Code • Serial Number • Air Conditioner Description • Air conditioner size • Cooling system document
			Step 13: The system will update the Project status Via an HttpPut method performed in the API which is called within the Typescript component. The Project table will be updated and the following attribute will be updated: • Project Status ID • Date Updated (current Date use case is executed) The Project status ID will be read from the **Project Status** table to reference the correct project status relating to the uploading of the radio Frequency report.
			Step 14: The System displays an alert to notify on the successful uploading of the Radio Frequency report
			Step 15: The system Generates an email that indicates the Radio Frequency report has been captured and uploaded into the system
			Step 16: The system reads the employee email address attribute stored in the **Employee** table to be able to send the email to the appropriate employees. The following employees are emailed the generated email: • Financial Officer

			• Quality Control Officer
			Step 17: The System displays an alert to notify the Operations Manager that the Financial Officer and the Quality Control Officer have been emailed on the uploading of the Radio Frequency report.
ALTERNATE COURSES:	**ALT Step 5:** The system captures that there is already a Radio Frequency report uploaded to the system. The system notifies to the Operations manager that if a new one is created the old Radio Frequency Report will be over ridden with the new one to be uploaded. **Skips to step 6**		
	ALT Step 8: The entered information is invalid; the system displays a red border around the incorrect input fields indicating that the entered data is invalid. **Return to step 7**		
	ALT Step 9: The Operations Manager clicks on the "No" button displayed on the modal. Indicating the user does not want to upload the Radio Frequency Report. **The use case is terminated.**		
	ALT Step 11: The Operations Manager clicks on the "No" button displayed on the modal. Indicating the Operations manager does not want to upload the Radio Frequency report.		
CONCLUSION:	This use case concludes once the Operational Manager, Quality Control Officer and Financial Officer are notified on the uploading of the Radio frequency report.		
POST-CONDITION:	• Operations Manager is notified on Radio Frequency report upload. • Financial Manager is notified on Radio Frequency report upload. • Quality Control Officer is notified on Radio Frequency report upload.		
BUSINESS RULES:	• Only the Operations Manager can upload the Radio Frequency report. • The client has to be aware of Radio Frequency report once construction site is completed. • Radio Frequency report has to be uploaded before generating of invoice may begin. • Radio Frequency report has to be uploaded before Site inspection from the Quality Control Officer can take place.		
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none		

ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.4.15 Search Radio Frequency report

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Michael Vosloo</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Search Radio Frequency report</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">4.15</td><td rowspan="3">Business Requirements: ☐
System Analysis: o
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">User</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">• Operations Manager
• Health and Safety Officer
• Director
• Quality Control Officer
• Financial Officer</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• Client</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the process the User and system will go through when the user desires to search and view the Radio Frequency report. The Radio Frequency report is searched from the system and then displayed to the user in a pdf format.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• Radio Frequency report is uploaded to the Report table.
• Financial Officer was notified on upload of Radio Frequency report.
• Quality Control Officer was notified on upload of Radio Frequency reports.
• User must be logged into the system</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The user requests to view Radio Frequency Report.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The user wants to</td><td>Step 2: The user hovers over</td><td>Step 3: The system will display the dropdown with the following items:
• Projects</td></tr>
</table>

<table>
<tr>
<td rowspan="2"></td>
<td>search for a Radio Frequency report.</td>
<td>the construction menu strip.</td>
<td>

Construction Sites
Allocated teams
Compliance Documents
Radio Frequency reports
Snags
Acceptance Documents
Site sign-offs
Commission reports
Site types

</td>
</tr>
<tr>
<td></td>
<td>Step 4: The user clicks on the Radio Frequency report dropdown menu item. [ALT]</td>
<td>Step 5: The system displays the search Radio Frequency report page with the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Radio Frequency Reports</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Search label</td><td>Label</td><td>This is the label for the search filter function.</td></tr>
<tr><td>Keyword</td><td>Textbox</td><td>To enter a Radio Frequency report name to search for.</td></tr>
<tr><td>Search</td><td>Button</td><td>This button will filter the table for the Searched for Radio Frequency report details in the table.</td></tr>
<tr><td>Table</td><td>Table</td><td>This table will hold the information about the Radio Frequency reports on the system.</td></tr>
<tr><td>Radio Frequency report ID</td><td>Label</td><td>Part of the table: Displays the ID of the radio frequency report.</td></tr>
<tr><td>Radio Frequency Report name</td><td>Label</td><td>Part of the table: Displays the name of the radio</td></tr>
</table>
</td>
</tr>
</table>

					frequency report.
			Date uploaded	Label	Part of the table: Displays the date the radio frequency report has been uploaded.
			Project Name	Label	Part of the table: Displays the project name that the Radio Frequency report is associated to.
			View	button	The user clicks this button to view the Radio Frequency report details and document.
			Upload	button	This button is clicked when the user wants to upload a new Radio Frequency report to the system.

the page will display a table of all the Radio Frequency report entries that are stored on the system. The table data is read from the **Radio Frequency Report** and **Project** tables. The data obtained and displayed attributes that are prevalent to the table listed above are:

Radio Frequency Report table:

- Radio Frequency report ID
- Name
- Date uploaded

Project Table:

- Project Name

		Step 6: The user enters the	**Step 7:** The system will capture and validate the entered information that was typed into the search textbox. The

		Radio Frequency report search criteria in the search text box. **[ALT]**	field is validated and displays a green border around the textbox. **[ALT]**
			Step 8: The system will submit the captured search criteria and search for the Radio Frequency report details in the **Radio Frequency report** table using a parametrised HttpGet request in the typescript component and API request. The system captures and saves the searched result to a list object and displays the search result in the table components, replacing the previous Radio Frequency report table items as they do not follow the searched criteria. **[ALT]**
		Step 9: The user clicks on the desired Radio Frequency report details displayed on the table.	**Step 10:** The system displays the Radio Frequency report page that is attributed by the **Radio Frequency Report, Construction Site** and **Project** tables. The aforementioned tables are inner joined together via their respective primary and foreign keys within the relevant tables. The data being pulled from each table via the inner join functions are as follows: **Project** table: • Project Name **Construction Site** table: • Site Name • Item Code • Serial Number • Air Conditioner Description • Air Conditioner Size • Cooling system document **Radio Frequency Report** table: • Name • Date Uploaded • Radio Frequency report The read data from the aforementioned tables are read via inner join functions that are performed on the API via an HttpGet method, the API is called from

type script component and is returned as an object. The Object returned stores all the details that are then read out on the following html elements:

Input name:	Control Type:	Notes:
Radio Frequency Report	Heading	Page heading
Name	label	The name of the Radio Frequency report
Radio Frequency Report	label	Indicates that the following button is to view the radio frequency report
View	button	Button is clicked to view the radio frequency report document.
Item Code	Label	Indicates the item code of the air conditioner.
Serial Number	Label	Indicates the serial number of the air conditioner
Air conditioner description	Paragraph	Displays the air conditioners description.
Air conditioner size	Label	Displays the air conditioners size

			Cooling system document	label	Indicates that the following button is to view the Cooling system document.
			View	button	Button is clicked to view the cooling system document.
			Project Name	Label	Indicates the linked project to the Radio Frequency report.
ALTERNATE COURSES:	**ALT Step 4:** The user selects the “Radio Frequency” elements on the “Project” page, the system will skip to step 5 but only display the Radio Frequency report details associated to that project. The project is linked to the construction site which entails the Construction site has the Project table has a foreign key within the Construction site table to then identify the Construction site allocated with the Radio Frequency report. The Project and radio Frequency report tables will be joined for this step via an inner join function performed in SQL and the API. **Skip to step 5:**				
	ALT Step 6: The user selects a “radio frequency report” from the displayed list present on screen. **Skip to step 9**				
	ALT Step 7: The entered search criteria is invalid; the system displays a red border around the search textbox indicating that the entered search criteria is invalid. **Return to step 6**				
	ALT Step 8: The search criteria entered into the search textbox does not match any existing records in the database. The system notifies and displays a modal to the user to indicate that the searched for Radio frequency report details do not exist within the database as a null value was obtained from the HttpGet request. **Return to step 6**				
CONCLUSION:	This use case concludes when the Radio Frequency report details are displayed.				
POST-CONDITION:	The correct radio frequency report details page is displayed to the user.				

BUSINESS RULES:	• Radio Frequency report has to be used for invoicing to begin
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.4.16 Request Site Visit

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Michael Vosloo</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Request Site Visit</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">4.16</td><td>Business Requirements: ☐</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td><td>System Analysis: o</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td><td>System Design: ☑</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Site Manager (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">Operations Manager (PSA)</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">• Quality Control Officer</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• Client
• Director
• Financial Officer</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the process the Operations Manager will go through once the Site Manager notifies the Operations manager that construction has been fully completed, the Complete construction photos have been uploaded and the Radio Frequency report has been uploaded. The Operations manager will request the system to notify the Quality Control Officer to come and request the site inspection, to then start the process to start creating final documents for the future use cases.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• Radio Frequency report has been uploaded
• Complete Construction Photos have been uploaded.
• Site Manager notifies Operations Manager that construction is fully complete.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The Operations Manager wishes to request site inspection by Quality Control Officer.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The Site Manager Notifies the Operations Manager that</td><td>Step 2: The Operations Manager wants to request the</td><td>Step 3: The system invokes use case 4.9 “Search Project”</td></tr>
</table>

<table>
<tr><td></td><td>the Construction has been completed.</td><td>Quality Control Officer to visit and inspect the Construction site.</td><td></td></tr>
<tr><td></td><td></td><td>Step 4: The Operations Manager clicks on the “Request Site Visit” card displayed.</td><td>Step 5: The System reads the Project Status ID from the Project table to identify what the project status is currently at by relating the Project Status ID to the Project Status table. The system determines that the Project is eligible for the Site visit to be requested. [ALT]</td></tr>
<tr><td></td><td></td><td></td><td>Step 6: The System displays a modal to the Operations Manager to confirm to request the Site visit. The modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Confirmation required</td><td>heading</td><td>Modal heading</td></tr>
<tr><td>Are you sure you want to request the Site visit from the Quality Control Officer?</td><td>Paragraph</td><td>Queries the user to confirm to request to Site visit.</td></tr>
<tr><td>Yes</td><td>Button</td><td>Clicked to confirm to request the site visit.</td></tr>
<tr><td>No</td><td>Button</td><td>Clicked to reject the request for the Quality Control Officer to visit the site.</td></tr>
</table></td></tr>
<tr><td></td><td></td><td>Step 7: The Operations Manager clicks the “Yes” button</td><td>Step 8: The system invokes use case 11.5 “Generate Site Acceptance Document”</td></tr>
</table>

		displayed on the modal. **[ALT]**	
			Step 9: The System generates an email that will display a request for the Quality Control Officer to visit the Construction site and inspect it with the Generated Acceptance Document provided.
			Step 10: The system reads the employee email address attribute stored in the Employee table to be able to send the email to the appropriate employee. The following employees are emailed the generated email: • Quality Control Officer
			Step 11: The system updates the Project Status ID in the **Project** table to indicate that the Site visit has been requested by the Operations Manager. The following attributes are updated in the **Project** table: • Project Status ID • Date Updated (takes the current date the Use case is executed.)
			Step 12: The system displays a notification to the Operations Manager indicating that the Quality Control Officer has been requested and emailed to visit the Construction Site.
ALTERNATE COURSES:	ALT Step 5: The System identifies that the Project is not eligible to request the site visit as the Project Status has not yet reached the required status to be able to request to visit the site. The system displays a notification to the Operations Manager indicating the Project is not yet eligible to request the Site visit from the Quality Control Officer. **The use case is terminated**		
	ALT Step 7: The Operations Manager clicks the “No” button indicating he does not want to request the Site visit from the Quality Control Officer. **The use case is terminated**		

CONCLUSION:	This use case concludes when the Quality Control Officer is notified via email that a site visit has been requested for inspection and to list snags on the construction site.
POST-CONDITION:	• The basic Acceptance document is created and populated with the necessary and relevant project details. • The Quality Control Officer is notified via email on request for site visit.
BUSINESS RULES:	• Only the Operations Manager can request a site visit from the Quality control officer.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.4.17 Create Snag

Gohvan Construction			
Author (s): **Michael Vosloo**	**Date: 28/06/2021**	**Version: 1.0**	
USE CASE NAME:	**Create Snag**	**USE CASE TYPE**	
USE CASE ID:	**4.17**	**Business Requirements:** ☐	
PRIORITY:	Medium	**System Analysis:** o	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	Quality Control Officer (PBA)		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	• Client • Director • Operations Manager • Construction team • Health and Safety Officer		
DESCRIPTION:	This use case entails the process the system will go through when the Quality Control Officer wants to create a new snag on the system for future use on construction sites. The System will present a form to the Quality Control Officer and prompt the Quality Control Officer to enter the details of the snag. The system will validate that the snag details entered are of the correct types and is valid. The details will then be capture and saved into the system.		
PRE-CONDITION:	None		
TRIGGER:	The Quality Control Officer wants to create a new snag.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The Quality Control Officer wants to create a new snag on the system.		

<table>
<tr>
<td></td>
<td>Step 2: The Quality Control Officer hovers over the Construction menu strip.</td>
<td></td>
<td>Step 3: The system displays the dropdown menu with the following items:
Projects
Construction Sites
Allocated teams
Compliance Documents
Radio Frequency Reports
Snags
Acceptance Documents
Site Sign-offs
Commission reports
Site types</td>
</tr>
<tr>
<td></td>
<td>Step 4: The Quality Control Officer clicks on the “Snags” dropdown menu item.</td>
<td></td>
<td>Step 5: The system displays the Snag details page with the followings page elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Snags</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Search</td><td>Label</td><td>This is the label for the search filter function.</td></tr>
<tr><td>Keyword</td><td>Textbox</td><td>To enter a Snag search criterion to search for.</td></tr>
<tr><td>Search</td><td>Button</td><td>This button will filter the table for the Searched for Snag details in the table.</td></tr>
<tr><td>Table</td><td>table</td><td>This table will hold the information about the Snag details.</td></tr>
<tr><td>Snag ID</td><td>Label</td><td>Part of the table: Displays the snag ID.</td></tr>
<tr><td>Snag item</td><td>Label</td><td>Part of the table: Displays the snag item.</td></tr>
<tr><td>View</td><td>Button</td><td>Part of the table: the user clicks this button to view the details of the</td></tr>
</table>
</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td></td>
<td></td>
<td>
<table>
<tr><td></td><td></td><td>corresponding snag.</td></tr>
<tr><td>New snag</td><td>Button</td><td>This button is clicked when the user wants to create a new snag.</td></tr>
</table>
This page will display a table of all the snags that are stored in the system. The table data is read from the Snags table. The data obtained and displayed attributes from the tables that are prevalent to the table listed above are:

Snag ID
Snag item

The table attributes are read via an HttpGet method in the typescript component and an API request.
</td>
</tr>
<tr>
<td></td>
<td>Step 6: The Quality Control Officer clicks on the “New snag” button.</td>
<td></td>
<td>Step 7: The system displays the create snag page. The create snag page contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Snag</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Snag item</td><td>label</td><td>Indication to the user that the snag item must be inputted.</td></tr>
<tr><td></td><td>input</td><td></td></tr>
<tr><td>Date created</td><td>label</td><td>Displays the date the snag was created</td></tr>
<tr><td></td><td>Input (Disabled)</td><td>This input contains the current date the use case has been executed on.</td></tr>
<tr><td>Description</td><td>paragraph</td><td>Indication to the user that the snag</td></tr>
</table>
</td>
</tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td></td><td></td><td>description item must be inputted.</td></tr>
<tr><td></td><td>input</td><td></td></tr>
<tr><td>Create</td><td>Button (disabled)</td><td>Button is clicked to create the snag.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>Button is clicked to cancel the process of creating a new snag.</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 8: The Quality Control Officer enters the snag details in the relevant input fields.</td><td></td><td>Step 9: The system captures and validates the provided information entered into the fields:
Snag item
Snag description
Create Button is enabled.
[ALT]</td></tr>
<tr><td></td><td>Step 10: The Quality Control Officer clicks the “Create” button [ALT]</td><td></td><td>Step 11: The system loads the confirmation modal and displays it to the Quality Control Officer to prompt the confirmation of creating the new snag. The Modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Confirmation Snag creation?</td><td>Label</td><td></td></tr>
<tr><td>Yes</td><td>Button</td><td>This button is clicked to create the new snag.</td></tr>
<tr><td>No</td><td>Button</td><td>This button is clicked to cancel the creation of the new snag.</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 12: The Quality Control Officer clicks on the “Yes” Button
[ALT]</td><td></td><td>Step 13: The system saves the snag details to the Snag table. The snag details are created in the snag tables via an HttpPost method sent in the API which is called through the Typescript</td></tr>
</table>

			component. The following attributes are created in the S**nag** table: • Snag ID • Snag Item • Snag Description • Date Created (current date the use case is executed on)
			Step 14: The System displays an alert to notify on the successful creation of the snag.
ALTERNATE COURSES:	**ALT Step 9:** The entered information is invalid; the system displays a red border around the search textbox indicating that the entered search criteria is invalid. **Return to step 8**		
	ALT Step 10: The Quality Control Officer clicks on the “Cancel” button. **The Use case is terminated**		
	ALT Step 12: The Quality Control Officer clicks on the “No” button**.** The confirmation modal is closed. **Return to step 8**		
CONCLUSION:	This use case is concluded when the Quality Control Officer is alerted that the snag was successfully created.		
POST-CONDITION:	The new snag details are stored in the database.		
BUSINESS RULES:	Only the Quality Control Officer can create a new snag.		
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none		
ASSUMPTIONS:	none		
OPEN ISSUES:	none		

1.2.4.18 Search Snag

Gohvan Construction			
Author (s): **Michael Vosloo**	**Date: 28/06/2021**	**Version: 1.0**	
USE CASE NAME:	**Search Snag**	**USE CASE TYPE**	
USE CASE ID:	**4.18**	**Business Requirements:** ☐	
PRIORITY:	Medium	**System Analysis:** o	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	User (PBA)		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	Quality Control Officer Operations Manager		
DESCRIPTION:	This use case entails the process the system will go through when searching for a specific Snag that has been captured and stored on the system. The User will enter the desired search criteria into the system and the corresponding Snags matching the search criteria will be displayed to the user.		
PRE-CONDITION:	The Snag exits on the system.		
TRIGGER:	The user wants to search for a Snag.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: user wants to search for snags on the system	Step 2: The user hovers over the Construction menu strip.	Step 3: The system will display the dropdown with the following items: • Projects • Construction Sites • Allocated teams • Compliance Documents • Radio Frequency reports • Snags • Acceptance Documents • Site sign-offs

<table>
<tr><td rowspan="2"></td><td></td><td></td><td>

Commission reports
Site types

</td></tr>
<tr><td></td><td>Step 4: The user clicks on the “Snags” dropdown menu item.</td><td>Step 5: the system displays the search Snags page with the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Snags</td><td>heading</td><td>Page heading</td></tr>
<tr><td>Search label</td><td>Label</td><td>This is the label for the search filter function.</td></tr>
<tr><td>Keyword</td><td>Textbox</td><td>To enter a snag search criterion to search for.</td></tr>
<tr><td>Search</td><td>button</td><td>This button will filter the table for the Searched for snag details in the table.</td></tr>
<tr><td>Table</td><td>Table</td><td>This table will hold the information about the snag details.</td></tr>
<tr><td>Snag ID</td><td></td><td>Part of the table: Displays the snag ID.</td></tr>
<tr><td>Snag Name</td><td></td><td>Part of the table: Displays the snag name.</td></tr>
<tr><td>Date created</td><td></td><td>Part of the table: Displays the date the snag was created.</td></tr>
<tr><td>New snag</td><td>Button</td><td>This button is clicked when the user wants to enter a new snag to the system</td></tr>
</table>
This page will display a table of all the snags that are stored in the system. The table data is read from the Snag table. The data obtained and displayed attributes from the tables that are prevalent to the table listed above are:

Snag ID

</td></tr>
</table>

			• Snag item (name) • Date created The table attributes are read via an HttpGet method in the typescript component and an API request.
		Step 6: The user enters the snag search criteria in the search textbox. **[ALT]**	Step 7: The system will capture and validate the entered information that was typed into the search textbox. The field is validated and displays a green border around the textbox. **[ALT]**
			Step 8: The system will submit the captured search criteria and search for the snag details in the **snag** table using a parameterised HttpGet request in the typescript component and API request. The system captures and saves the searched result to a list object and displays the search result in the table components, replacing the previous Snag table items as they do not follow the search criteria. **[ALT]**
		Step 9: The use clicks on the desired Snag details displayed on the table.	Step 10: The system displays the snag item page that is attributed by the **Snag** table. The data that is read from the snag table is as follows: **Snag** table: • Snag ID • Snag item • Snag description • Date created The read data from the aforementioned table are a basic select function that is performed on the API via an HttpGet method, the API is called from type script component and is returned as an object. The Object returned stores all the

<table>
<tr><td></td><td></td><td></td><td>details that are then read out on the following html elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Snag</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Snag item</td><td>label</td><td>Displays the snag item</td></tr>
<tr><td>Date created</td><td>label</td><td>Displays the date the snag was created</td></tr>
<tr><td>Description</td><td>paragraph</td><td>Displays the description of the snag.</td></tr>
<tr><td>Edit</td><td>button</td><td>This button is clicked when the user wishes to create a new snag item.</td></tr>
</table></td></tr>
<tr><td rowspan="3">ALTERNATE COURSES:</td><td colspan="3">ALT Step 6: The user selects a snag from the displayed list present on screen. Skip to step 9</td></tr>
<tr><td colspan="3">ALT Step 7: The entered information is invalid; the system displays a red border around the search textbox indicating that the entered search criteria is invalid.
Return to step 6</td></tr>
<tr><td colspan="3">ALT Step 8: The search criteria entered into the search textbox does not match any existing records in the database. The system notifies and displays a modal to the user to indicate that the searched for snag details do not exist within the database as a null value was obtained from the HttpGet request.
Return to step 6</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">This use case concludes when the snag details page is displayed to the user.</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">The correct snag details page is displayed to the user.</td></tr>
<tr><td>BUSINESS RULES:</td><td colspan="3">All users can search for a snag.</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td colspan="3">none</td></tr>
<tr><td>ASSUMPTIONS:</td><td colspan="3">none</td></tr>
</table>

OPEN ISSUES:	none

1.2.4.19 Edit Snag

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Michael Vosloo</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Edit Snag</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">4.19</td><td rowspan="3">Business Requirements: ☐
System Analysis: o
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Low</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Quality Control Officer (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• Director
• Client
• Construction team
• Operations Manager
• Health and Safety Officer</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case entails the process the system will go through when the Quality Control Officer wants to edit the details of an existing snag on the system. The system will prompt the user to search for a snag that exists on the system. The system will display the details of the searched snag to the Quality Control Officer. The Quality Control Officer will then enter the new details of the snag, the system then captures and validates that the snags details are correct and valid. The edited snag details will be saved to the system and the Quality Control Officer.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">The snag must exist on the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The Quality Control Officer wants to edit a snag on the system.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td>Step 1: The Quality Control Officer wants</td><td></td><td></td></tr>
</table>

<table>
<tr><td></td><td>to edit snag details.</td><td></td><td></td></tr>
<tr><td></td><td>Step 2: The Quality Control Officer hovers over the Construction menu strip.</td><td></td><td>Step 3: The system will display the dropdown with the following items:
Projects
Construction Sites
Allocated teams
Compliance Documents
Radio Frequency reports
Snags
Acceptance Documents
Site Sign-offs
Commission reports
Site types</td></tr>
<tr><td></td><td>Step 4: The Quality Control Officer clicks on the “Snags” dropdown menu item.</td><td></td><td>Step 5: The system invokes use case 4.19 “Search snag”</td></tr>
<tr><td></td><td>Step 6: The Quality Control Officer clicks on the “Edit” button displayed.</td><td></td><td>Step 7: The System enabled all the inputs that are displayed on the page. The following elements are present on the modal with pre-populated information due to the data obtained from the “search snag” invoke:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Edit Snag details</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Snag item</td><td>label</td><td>None</td></tr>
<tr><td></td><td>Input</td><td>Corresponding input for the user to input the snag item.</td></tr>
<tr><td>Date created</td><td>label</td><td>Displays the date the snag was created</td></tr>
<tr><td>Description</td><td>Label</td><td>None</td></tr>
<tr><td></td><td>Input</td><td>Corresponding input for the user to input the snag description</td></tr>
<tr><td>Confirm Edit</td><td>Button (disabled)</td><td>This button is changed once</td></tr>
</table></td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td></td><td></td><td>the edit button has been clicked to indicate that the user must confirm to capture the new details of the edited snag.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>User clicks this to cancel the editing of the snag details.</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 8: The Quality Control Officer enters the new snag details</td><td></td><td>Step 9: The system captures and validates the provided information entered into the fields:
Snag item
Snag description
Create Button is enabled.
[ALT]</td></tr>
<tr><td></td><td>Step 10: The Quality Control Officer clicks the “Confirm Edit” button
[ALT]</td><td></td><td>Step 11: The system loads the confirmation modal and displays it to the Quality Control Officer to prompt the confirmation of editing the new snag details. The Modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Confirmation to edit snag.</td><td>Label</td><td></td></tr>
<tr><td>Yes</td><td>Button</td><td>This button is clicked to edit the details of the snag.</td></tr>
<tr><td>No</td><td>Button</td><td>This button is clicked to cancel the editing of the new snag details.</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 12: The Quality Control Officer clicks the “yes” button displayed.
[ALT]</td><td></td><td>Step 13: The system edits the snag details to Snag table. The snag details are edited in the Snag table via an HttpPut method sent in the API which is called through the Typescript component. The following attributes are edited in the Snag table:
Snag item
Snag Description</td></tr>
</table>

			Step 14: The System displays an alert to notify on the successful editing of the snag details.
ALTERNATE COURSES:	**ALT Step 9:** The entered information is invalid; the system displays a red border around the search textbox indicating that the entered search criteria is invalid. **Return to step 8**		
	ALT Step 10: The Quality Control Officer clicks on the “Cancel” button. **The Use case is terminated**		
	ALT Step 12: The Quality Control Officer clicks on the “No” button**.** The confirmation modal is closed. **Return to step 7**		
CONCLUSION:	This use case is concluded when the Quality Control Officer is alerted that the snag details were successfully edited.		
POST-CONDITION:	The snag details were successfully updated.		
BUSINESS RULES:	Only the Quality Control Officer can edit the snag details.		
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none		
ASSUMPTIONS:	none		
OPEN ISSUES:	none		

1.2.4.20 Edit Acceptance Document

Gohvan Construction		
Author (s): **Michael Vosloo**	**Date: 28/06/2021**	**Version: 1.0**
USE CASE NAME:	**Edit Acceptance Document**	**USE CASE TYPE**
USE CASE ID:	**4.20**	**Business Requirements:** ☐
PRIORITY:	High	**System Analysis:** o
SOURCE:	Gohvan Construction System	**System Design:** ☑
PRIMARY BUSINESS ACTOR:	Quality Control Officer (PBA)	
PRIMARY THE SYSTEM ACTOR:	None	
OTHER PARTICIPATING ACTORS:	• Operations Manager (ERA)	
OTHER INTERESTED STAKEHOLDERS:	• Client • Financial Officer • Director • Site Manger • Construction team	
DESCRIPTION:	This use case describes the process the Quality Control Officer will go through when the Site visit is taking place, the Quality Control Officer will fill out the form in the Acceptance Document to log down the snags notified on the site visit that was requested by the Operations Manager. The quality Control Officer can either select a snag that has a common occurrence through a list of common snags or the Quality Control officer can create a new snag specific to that site problem. All snags listed on the Acceptance document will be specific to the construction site. Once all snags have been listed on the Acceptance document, the Quality Control Officer will notify the Construction teams on site to fix the snags. The Director will then be notified on the acceptance document.	
PRE-CONDITION:	• The acceptance document is created on the system • The Quality Control Officer is on site performing the site inspection.	
TRIGGER:	The Quality Control Officer requests to log a new snag onto the acceptance document.	
TYPICAL COURSE		**SYSTEM RESPONSE:**

<table>
<tr><td rowspan="3">OF EVENTS:</td><td>ACTOR ACTION:</td><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The Quality Control Officer wants to edit the acceptance document information.</td><td></td><td>Step 2: The system invokes use case 4.21 “Search Acceptance Document”</td></tr>
<tr><td>Step 3: The Quality Control Officer clicks the “Edit” button displayed</td><td></td><td>Step 4: The system displays the Edit Acceptance Document page. The page contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Acceptance Document</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Name:</td><td>Label</td><td>Displays the Name of the Acceptance Document.</td></tr>
<tr><td></td><td>Input</td><td>Corresponding input for the user to enter the updated Acceptance Document Name</td></tr>
<tr><td>Date Created:</td><td>Label</td><td>Displays the date the Acceptance Document was created</td></tr>
<tr><td>Accepted Date:</td><td>Label</td><td>Displays the date the acceptance document was accepted.</td></tr>
<tr><td>View Acceptance Document</td><td>Button (disabled)</td><td>Clicked to view the Acceptance Document.</td></tr>
<tr><td>Note:</td><td>Label</td><td>Indicates the notes for the Acceptance Document.</td></tr>
<tr><td></td><td>Input</td><td>Corresponding input for the user to enter the note for the Acceptance Document</td></tr>
</table></td></tr>
</table>

			Snags	Label	None
			Table	Table	This holds all the snags caught on the construction site and which are listed on the acceptance document.
			Snag Item	Label	Part of the table: Displays the snag Item on the acceptance document
			Snag Description	Label	Part of the table: Displays the snag description on the acceptance document
			Date caught	Label	Part of the table: Displays the date the snag was uploaded to the acceptance document.
			Completed	Button	This button is clicked to mark the snag complete on the Acceptance Document
			Remove	Button	This button is clicked to remove the snag from the Acceptance Document
			Add New Snag	Button	This button is clicked to add a snag to the Acceptance Document
			Confirm	Button	This button is clicked to begin the process of saving the updated Acceptance Document information.

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>Cancel</td><td>Button</td><td>This button is clicked to cancel the process of editing the Acceptance Document.</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 5: The Quality Control enters the updated Acceptance document details.</td><td></td><td>Step 6: The system captures and validates the provided information entered into the fields: [ALT]</td></tr>
<tr><td></td><td>Step 7: The Quality Control Officer clicks the “Add new snag” button. [ALT]</td><td></td><td>Step 8: The system displays the Add Snag Model. The modal invokes use case 4.18 “Search Snag” to obtain the snag items stored.
The modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Snag Detail</td><td>Heading</td><td>Modal Heading</td></tr>
<tr><td>Snag item</td><td>label</td><td>Indicates the snag item.</td></tr>
<tr><td></td><td>Dropdown text box</td><td>The user can enter a custom name for the snag to be added or the user can select an existing snag on the system that was captured from the Invoke “Search Snag”</td></tr>
<tr><td>Sang Description</td><td>Label</td><td>Indicates the snag description</td></tr>
<tr><td></td><td>Textbox</td><td>User enters the Snag description that was caught on the construction site.</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>Date Caught</td><td>Label</td><td>Indicates the date the Snag was caught on the Construction site.</td></tr>
<tr><td></td><td>Input (disabled)</td><td>Displays the date the snag was caught.</td></tr>
<tr><td>Add</td><td>Button</td><td>Button is clicked to add the sang to the acceptance document.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>Button is clicked to cancel the process of adding the snag to the acceptance document.</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 9: The Quality Control Officer enters the snag details.</td><td></td><td>Step 10: The system captures and validates the provided information entered into the fields: [ALT]</td></tr>
<tr><td></td><td>Step 11: The Quality Control Officer clicks the “Add” button. [ALT]</td><td></td><td>Step 12: The System loads the confirmations to prompt to the user to confirm to add the snag to the acceptance document. The Modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Confirmation to Add snag to Acceptance Documents.</td><td>Label</td><td></td></tr>
<tr><td>Yes</td><td>Button</td><td>This button is clicked to add the snag to the acceptance document.</td></tr>
<tr><td>No</td><td>Button</td><td>This button is clicked to cancel the adding of the snag to the acceptance document.</td></tr>
</table>
</td></tr>
</table>

	Step 13: The Quality Control Officer clicks the “Yes” button displayed. **[ALT]**		**Step 14:** The System adds and saves the updated details to the **Project Snag List** table, the following attributes are added to the Project Snag table via an HttpPost method performed on the API which was called in the Typescript component: • Snag ID • Acceptance Document ID • Date Uploaded • Snag Description
			Step 15: The System closes the opened Modals and displays the Edit Acceptance Document page. The sang table present on the Edit Acceptance Document page is updated accordingly to the present snags on the Acceptance Document.
	Step 16: The Quality Control Officer clicks the “Confirm” button displayed **[ALT]**		**Step 17:** The System updates the Acceptance Document stored in the **Acceptance Document** table, by adding in the updated snags and notes now associated to the acceptance document. The following attributes will be added to the Acceptance document: • Snag Item • Snag Description • Notes
			Step 18: The System generates and email that displays the updated Acceptance document that has been captured to the system.
			Step 19: The system reads the employee email address attribute stored in the Employee table to be able to send the email to the appropriate employees. The following employees are emailed the generated email: • Operations Manager
			Step 20: The system displays a notification to indicate that the Operations Manager has been emailed on the updated Acceptance Document.

ALTERNATE COURSES:	**ALT step 6:** The entered information is invalid; the system displays a red border around the incorrect input fields indicating that the entered data is invalid. **Return to step 5**
	ALT Step 7a: The Quality Control Officer clicks the remove snag icon next to the relevant snag displayed on the snags table. The Snag is removed from the list of snags on the Acceptance Document and updates the snags associated to the Acceptance Document. **Skip to step 14**
	ALT Step 7b: The Quality Control Officer clicks the complete snag icon next to the relevant snag displayed on the snags table. The Snag is updated from the list of snags on the Acceptance Document and updates the snags associated to the Acceptance Document. **Skip to step 14**
	ALT Step 7c: The Quality Control Officer does not wish to update snags associated to the Acceptance document. The Quality Control Officer ignores the updating section of snags on the system. **Skip to step 16**
	ALT Step 10: The entered information is invalid; the system displays a red border around the incorrect input fields indicating that the entered data is invalid. **Return to step 9**
	ALT Step 11: The Quality Control Officer clicks the “Cancel” button, the modal is closed. **Skip to step 15**
	ALT Step 13: The Quality Control Officer clicks on the “No” button for the confirmation modal. **Return to step 9**
	ALT Step 16: The Quality Control Officer clicks the “Cancel” button displayed on the Edit Acceptance Document page. **Terminate Use case**
CONCLUSION:	This use case concludes when the Operations Manager and Site Manager are notified that the snags have been successfully added and updated to the Acceptance Document.
POST-CONDITION:	The Acceptance Document is updated with the correct snags and Acceptance document information.
BUSINESS RULES:	• Only the Quality Control Officer can update and create snags on the Acceptance document • The Quality Control officer must be on site to log snags onto the acceptance document.
IMPLEMENTATION CONSTRAINTS	none

AND SPECIFICATIONS:	
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.4.21 Search Acceptance document

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Michael Vosloo</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Search Acceptance document</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">4.21</td><td rowspan="3">Business Requirements: ☐
System Analysis: o
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">User (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• Quality Control Officer
• Director
• Operations Manager
• Client
• Construction Team
• Health and Safety
• Procurement and logistics officer</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the process the user and system will go through when the user requests to view the Acceptance document to view details on the construction site and possible snags on the construction site that have been identified and logged by the quality Control officer.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The user must be logged into the system
• The Acceptance document is created and loaded onto the system in the Acceptance_Document table.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The user requests to search and view the Acceptance Document</td></tr>
<tr><td rowspan="3">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The User wants to search for an</td><td></td><td></td></tr>
</table>

<table>
<tr><td></td><td>Acceptance Document</td><td></td><td></td></tr>
<tr><td></td><td>Step 2: The User hovers over the Construction menu strip.</td><td></td><td>Step 3: The system displays the dropdown with the following items:
Projects
Construction Sites
Allocated teams
Compliance Documents
Radio Frequency reports
Snags
Acceptance Documents
Site sign-offs
Commission reports
Site types</td></tr>
<tr><td></td><td>Step 4: The User clicks on the “Acceptance Documents” dropdown menu item.</td><td></td><td>Step 5: The System displays the Acceptance Documents page with the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Acceptance Documents</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Search</td><td>Label</td><td></td></tr>
<tr><td>Keyword</td><td>textbox</td><td>To enter a Acceptance Document search criteria</td></tr>
<tr><td>Search</td><td>Button</td><td>Clicked to search for Acceptance Document.</td></tr>
<tr><td>Table</td><td>Table</td><td>This table will hold the information about the Acceptance documents on the system.</td></tr>
<tr><td>ID</td><td>Label</td><td>Part of the table: Displays the Acceptance Documents ID</td></tr>
<tr><td>Name</td><td>Label</td><td>Part of the table: Displays the Acceptance Documents Name</td></tr>
</table></td></tr>
</table>

<table>
<tr>
<td></td>
<td></td>
<td></td>
<td>
<table>
<tr><td>Date Created</td><td>Label</td><td>Part of the table: Displays the date the acceptance document was created.</td></tr>
<tr><td>Construction Site Name</td><td>Label</td><td>Part of the table: Displays the Construction sites name that the acceptance document is associated to.</td></tr>
</table>
This page will display a list of all the currently active and existing acceptance documents stored on the system. The list is read from the stored acceptance document entries on the Acceptance Document and Construction Site tables in the database and the relevant attributes previously mentioned are populated with the following table attributes:

Acceptance Document ID
Acceptance Document Name
Date Created
Construction Site Name (inner joined with the Construction Site table)

The table attributes are done via an HttpGet method in the Typescript component and an API request.
</td>
</tr>
<tr>
<td></td>
<td>Step 6: The user enters the Acceptance Document name as a search criteria in the search text box. [ALT]</td>
<td></td>
<td>Step 7: The system will capture and validate the entered search criteria that was typed into the search textbox. The field is validated and displays a green border around the textbox. [ALT]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Step 8: The system will submit the captured search criteria and</td>
</tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>search for the acceptance Document in the Acceptance Document table using a parametrised HttpGet request in the typescript component and API request. The system captures and saves the searched result to a list object and displays the search result in the list components, replacing the previous acceptance Document list items as they do not follow the searched criteria. [ALT]</td></tr>
<tr><td></td><td>Step 9: The user clicks on the list object displayed via the list component.</td><td></td><td>Step 10: The system displays the Acceptance Document page where the elements are attributed by the Acceptance Document, Snag and Project Snag List tables. The following attributes that are present from the relative tables are as follows:
Acceptance Document table:
Acceptance Document Name
Date Created
Accepted Date
Acceptance Document
Project Snag List table:
Snag Description
Date uploaded
Snag table:
Snag Item
The following page elements are visible and attributed by the attributes previously mentioned:<table><tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr><tr><td>Acceptance Document</td><td>Heading</td><td>Page heading</td></tr><tr><td>Name:</td><td>Label</td><td>Displays the Name of the Acceptance Document.</td></tr><tr><td>Date Created:</td><td>Label</td><td>Displays the date the Acceptance Document was created</td></tr></table></td></tr>
</table>

			Accepted Date:	Label	Displays the date the acceptance document was accepted.
			View Acceptance Document	Button	Clicked to view the Acceptance Document.
			Note:	Label	Displays the Acceptance Document Note.
			Snags	Label	None
			Table	Table	This holds all the snags caught on the construction site and which are listed on the acceptance document.
			Snag Item	Label	Part of the table: Displays the snag Item on the acceptance document
			Snag Description	Label	Part of the table: Displays the snag description on the acceptance document
			Date caught	Label	Part of the table: Displays the date the snag was uploaded to the acceptance document.
			Edit	Button	This button is clicked to edit the details of the Acceptance document.
			Confirm	Button	This button is clicked when the Quality

					Control Officer confirms the Site acceptance Document to begin the process of the client visit.
			Return	Button	This button is clicked to return to the Acceptance Documents page.
ALTERNATE COURSES:	**ALT Step 6:** The user selects a Acceptance Document from the displayed list present on screen. **Skip to step 9**				
	ALT step 7: The entered information is invalid; the system displays a red border around the search textbox indicating that the entered search criteria is invalid. **Return to step 6**				
	ALT Step 8: The search criteria entered into the search textbox does not match any existing records in the database. The system notifies and displays a modal to the user to indicate that the searched for Acceptance Document details do not exist within the database as a null value was obtained from the HttpGet request. **Return to step 6**				
CONCLUSION:	This use case concludes when the Acceptance Document page is displayed to the user.				
POST-CONDITION:	The Acceptance Document page is displayed to the user with the correct project details				
BUSINESS RULES:	All users can search for project details.				
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none				
ASSUMPTIONS:	none				
OPEN ISSUES:	none				

1.2.4.22 Confirm acceptance document

Gohvan Construction		
Author (s): **Michael Vosloo**	**Date: 28/06/2021**	**Version: 1.0**
USE CASE NAME:	**Confirm acceptance document**	**USE CASE TYPE**
USE CASE ID:	**4.22**	**Business Requirements:** ☐
PRIORITY:	High	**System Analysis:** o
SOURCE:	Gohvan Construction System	**System Design:** ☑
PRIMARY BUSINESS ACTOR:	Site Manager (PBA)	
PRIMARY THE SYSTEM ACTOR:	Quality Control Officer (PSA)	
OTHER PARTICIPATING ACTORS:	• Client	
OTHER INTERESTED STAKEHOLDERS:	• Director • Operations Manager • Construction Team • Financial Officer • Health and Safety Officer	
DESCRIPTION:	This use case describes the process the Quality Control Officer and the system will go through when the snags have been completed and the Quality Control officer has confirmed that the snags have been completed. The Quality Control Officer will confirm on the system that the snags have been completed and that the Construction Site is ready for the client visit accompanied by the Quality Control Officer. The system will capture the Confirmation of the Acceptance document and notify the Quality Control officer to request a site visit by the Client for site sign off.	
PRE-CONDITION:	• The Acceptance Document must be uploaded to the system. • The Quality Control Officer must confirm snags have been completed. • The Site Manager informs the Quality Control Officer that snags have been completed.	
TRIGGER:	The Quality control officer wants to confirm that the snags have been completed.	
TYPICAL COURSE		**SYSTEM RESPONSE:**

<table>
<tr><td rowspan="4">OF EVENTS:</td><td>ACTOR ACTION:</td><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The Site Manager notifies the Quality Control Officer that the snags mentioned on the Acceptance Document for the Construction Site have been fixed.</td><td>Step 2: The Quality Control Officer wants to confirm the Acceptance Document and check all snags have been completed.</td><td>Step 3: The System invokes use case 4.21 “Search Acceptance Document”</td></tr>
<tr><td></td><td>Step 4: The Quality Control Officer clicks the “Confirm” button displayed on the page.</td><td>Step 5: The reads all the snags relating to the Acceptance Document from the Project Snag List table. The read entries associated to the Acceptance Document will be saved to an array of objects. The following attributes will be saved to the array of objects:
Snag Item
Snag Description
Date caught</td></tr>
<tr><td></td><td></td><td>Step 6: The system displays a modal to the Quality Control Officer indicating the snags on the Acceptance Document. The following elements are present on the modal: (the elements on the table are linked to the array of objects saved in the previous step)<table><tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr><tr><td>Confirm Acceptance Document</td><td>Heading</td><td>Modal heading</td></tr><tr><td>Snags on the Acceptance Document</td><td>label</td><td>Indicating the following table present the information of all the snags associated to the</td></tr></table></td></tr>
</table>

					Acceptance Document.
			Table	table	This holds all the snags caught on the construction site and which are listed on the acceptance document
			Snag Item	Label	Part of the table: Displays the snag Item on the acceptance document.
			Snag Description	Label	Part of the table: Displays the snag description on the acceptance document.
			Date Caught	Label	Part of the table: Displays the date the snag was uploaded to the acceptance document.
			Completed	Checkbox	Part of the table: Displays if the snag is completed or not.
			Confirm	Button (Disabled)	Clicked to confirm the Acceptance Document
			Cancel	Button	Clicked to cancel the confirming of the Acceptance Document
		Step 7: The Quality Control Officer marks the checkbox of	**Step 8:** The system validates and check that all snags associated to the system are marked complete. Confirm button is enabled. **[ALT]**		

<table>
<tr><td rowspan="4"></td><td></td><td>the completed snags on acceptance document. [ALT]</td><td></td></tr>
<tr><td></td><td>Step 9: The Quality Control Officer clicks the “Confirm” button displayed.</td><td>Step 10: The system displays a modal the Quality Control Officer to confirm to accept the Acceptance Document. The modal Contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Confirm the Acceptance Document?</td><td>heading</td><td>Modal heading</td></tr>
<tr><td>Yes</td><td>Button</td><td>This button is clicked to confirm the Acceptance Document</td></tr>
<tr><td>No</td><td>Button</td><td>This button is clicked to reject the confirmation of the acceptance document.</td></tr>
</table></td></tr>
<tr><td></td><td></td><td>Step 11: The system saves the acceptance document and updates the Acceptance Document table. The Acceptance Document table is updated via an HttpPut method performed within the API which was called from the Typescript component. The following attributes in the Acceptance Document table is updated:
Accepted Date</td></tr>
<tr><td></td><td></td><td>Step 12: The system updates the Project Status ID in the Project table to indicate that the Acceptance Documents has been confirmed by the Quality Control Officer. The following attributes are updated in the Project table:
Project Status ID
Date Updated (takes the current date the Use case is executed.)</td></tr>
</table>

			Step 13: The System generates an email that will notify Operations Manager that the Acceptance Document has been confirmed, indicating that the Construction site can be visited by the client.
			Step 14: The system reads the employee email address attribute stored in the Employee table to be able to send the email to the appropriate employee. The following employees are emailed the generated email: • Operations Manager
			Step 15: The System notifies the Quality Control Officer that the Acceptance Document has been successfully accepted and the Operations Manager has been notified on the confirmed Acceptance Document.
ALTERNATE COURSES:	**ALT Step 7:** The Quality Control Officer clicks the "Cancel" button present on the modal. Indication The Quality Control Officer does not want to confirm the Acceptance Document. **The use case is terminated**		
	ALT Step 8: The system still reads that not all the snags on the Acceptance Document have been marked complete. **Return the step 7**		
CONCLUSION:	This use case will conclude once the Project Status has been updated and the Operations Manager has been notified to request a site visit from the client.		
POST-CONDITION:	• The Project Status has been updated. • Snags have been fully completed. • Client site visit has been requested.		
BUSINESS RULES:	• Only the Quality Control Officer can confirm the acceptance document. • An inspection must take place to confirm the snags caught on the Acceptance Documents.		
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none		
ASSUMPTIONS:	none		
OPEN ISSUES:	none		

1.2.4.23 Capture Site Sign-off

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Michael Vosloo</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Capture Site Sign-off</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">4.23</td><td rowspan="3">Business Requirements: ☐
System Analysis: o
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Client (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">Quality Control Officer (PSA)</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">• Financial Officer (ERA)</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• Director
• Operations Manager</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the process the Quality Control Officer and System will go through once the client site visit has happened (Accompanied by the quality control officer).
The Quality Control Officer will upload the Site Acceptance Schedule to the system indicating that the client has signed off on the construction site and the final 10% invoice may be claimed. The Financial Officer will be notified that Site sign off has taken place and can begin the final claim process.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The Client site visit has taken place.
• The client has sent the Quality Control Officer the Site Acceptance Schedule (Site Sign off)</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The Quality Control Officer requests to capture the site sign off.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The Client sends the Quality Control Officer the Site</td><td>Step 2: The Quality Control Officer receives the Site Acceptance</td><td>Step 3: The System invokes use case 4.9 “Search Project”</td></tr>
</table>

<table>
<tr><td rowspan="4"></td><td>Acceptance Schedule via an email.</td><td>Schedule and requests to upload the Site Acceptance Schedule to the system.</td><td></td></tr>
<tr><td></td><td>Step 4: The Quality Control Officer clicks the “Site Sign-off” card displayed.</td><td>Step 5: The system reads from the Acceptance Schedule table to determine if there already is an entry in the table already allocated to the Project/Construction Site. [ALT]</td></tr>
<tr><td></td><td></td><td>Step 6: The displays a modal to the user to confirm to begin the site sign-off process. The Modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Confirmation to upload the site sign-off.</td><td>Heading</td><td>Modal Heading</td></tr>
<tr><td>Yes</td><td>Button</td><td>Clicked to confirm to upload site sign-off</td></tr>
<tr><td>No</td><td>Button</td><td>Clicked to cancel the uploading of the site sign-off.</td></tr>
</table></td></tr>
<tr><td></td><td>Step 7: The Quality Control Officer clicks the “Yes” Button [ALT]</td><td>Step 8: The system displays the Capture Site Sign-off page. The page contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Site Sign-off</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Name:</td><td>Label</td><td>None</td></tr>
<tr><td></td><td>Input</td><td>Corresponding input for the user to input the Name of the Acceptance Schedule.</td></tr>
</table></td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>Description:</td><td>Label</td><td>None</td></tr>
<tr><td></td><td>Input</td><td>Corresponding input for the user to enter the description of the Acceptance Schedule.</td></tr>
<tr><td>Construction Site Name</td><td>Label</td><td>None</td></tr>
<tr><td></td><td>Input (Disabled)</td><td>Autogenerates the associated Construction site name the Site Acceptance Document is associated to.</td></tr>
<tr><td>Date Uploaded:</td><td>Label</td><td>None</td></tr>
<tr><td></td><td>Input (disabled)</td><td>Autogenerates the current date the use case is being executed.</td></tr>
<tr><td>Upload Acceptance Schedule</td><td>label</td><td>None</td></tr>
<tr><td></td><td>Button</td><td>Clicked to select the Acceptance Schedule to be uploaded.</td></tr>
<tr><td>Upload</td><td>Button (disabled)</td><td>Clicked to upload the Acceptance Schedule</td></tr>
<tr><td>Cancel</td><td>Button</td><td>Clicks to cancel the uploading of the Acceptance Schedule</td></tr>
</table>
</td></tr>
<tr><td></td><td></td><td>Step 9: The Quality Control Officer enters the Acceptance Schedule information.</td><td>Step 10: The system will capture and validate the entered information. The field is validated and displays a green border around the textbox.
Upload Button is enabled. [ALT]</td></tr>
<tr><td></td><td></td><td>Step 11: The Quality Control Office will click the</td><td>Step 12: The system displays a confirmation modal to the Quality Control Office r to confirm to upload</td></tr>
</table>

<table>
<tr>
<td rowspan="3"></td>
<td></td>
<td>“Upload” button displayed [ALT]</td>
<td>the Acceptance Schedule. The modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Confirmation required</td><td>heading</td><td>Modal heading</td></tr>
<tr><td>Are you sure you want to upload the Acceptance Schedule</td><td>Paragraph</td><td>Queries the user to confirm to upload the Acceptance Schedule.</td></tr>
<tr><td>Yes</td><td>Button</td><td>Clicked to confirm to upload the Acceptance Schedule.</td></tr>
<tr><td>No</td><td>Button</td><td>Clicked to reject the upload of the Acceptance Schedule.</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 13: The Quality Control Officer clicks the “Yes” button. [ALT]</td>
<td>Step 14: The system uses an SQL insert query performed via an HttpPost request made within the Typescript component to create a new entry within the Acceptance Schedule table. The system will enter the following attributes obtained from the entered inputs to create the Acceptance Schedule entry.
Acceptance Schedule table:
• Acceptance Schedule ID
• Construction Site ID
• Name
• Description
• Acceptance Schedule
• Date Uploaded</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Step 15: The system updates the Project status in the Project table. The following attributes in the Project table are affected with the update performed in the API via an HttpPut request made in the Typescript component:
Project table:
• Project status_ID (read from the Project Status table)</td>
</tr>
</table>

			• Date updated (updates to the current date the use case was performed on)
			Step 16: The system invokes use case 4.21 “Search Acceptance Document” to retrieve the physical acceptance document.
			Step 17: The system then merges and joins the Acceptance Documents and Acceptance Schedule via the Typescript component will making use of the JSPDF JavaScript library to merge the two documents together.
			Step 18: The System then updates the Acceptance document via an HttpPut method performed in the API which ahs been called from the Typescript component. The following attribute in the Acceptance Document will be updated. • Acceptance Document
			Step 19: The system displays a notification to the Quality Control Officer that the Acceptance Schedule has been successfully captured, The Acceptance Document has been updated and the Site Sign-Off has been marked on the project status.
			Step 20: The System generates an email that displays the Site has been signed off.
			Step 21: The system reads the employee email address attribute stored in the Employee table to be able to send the email to the appropriate employees. The following employees are emailed the generated email: • Financial Officer
			Step 22: The system displays a notification to the Quality Control Officer that the Financial Officer has been notified and emailed on the Site sign-off.
ALTERNATE	**ALT Step 5:** The System captures that there already is an Acceptance Schedule associated to the Project which indicates		

COURSES:	that the Site Sign-Off has already taken place. The System directs the user to the Site Sign-off Page.
	ALT Step 7: The Quality Control Officer clicks the “No” button displayed on the modal. **The use case is terminated**
	ALT Step 10: The entered information is invalid; the system displays a red border around the incorrect input fields indicating that the entered data is invalid. **Return to step 9**
	ALT Step 11: The Quality Control Officer clicks the “Cancel” button displayed. Indicating the Quality Control Officer does not want to capture the Site Sign-Off details. **The use case is terminated**
	ALT Step 13: The Quality Control Officer clicks on the “No” button displayed on the modal. Indicating the Quality Control Officer does not want to upload the Site Sign-Off details. **Return to step 8**
CONCLUSION:	The use case concludes when the site acceptance schedule is uploaded and the Financial Officer is updated on the project status being that the Construction site has been sign-off by the Client.
POST-CONDITION:	• The Project status is updated • The Construction site is completely finished. • The Financial Officer is updated on site sign-off.
BUSINESS RULES:	• Only the Quality Control officer can upload the site acceptance schedule. • The financial Officer must be updated on the site sign-off • The client has to send the Quality Control officer the Site Acceptance schedule.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.4.24 Search Site Sign-off

Gohvan Construction			
Author (s): Michael Vosloo	**Date: 28/06/2021**	**Version: 1.0**	
USE CASE NAME:	**Search Site Sign-off**	**USE CASE TYPE**	
USE CASE ID:	**4.24**	**Business Requirements:** ☐	
PRIORITY:	Medium	**System Analysis:** o	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	User (PBA)		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	• Quality Control Officer • Director • Operations Manager • Health and Safety Officer		
DESCRIPTION:	This use case entails the process the system will go through when searching for a specific Site Sign-off (acceptance schedule) that has been captured and stored on the system relating to the project associated to the Site Sign-off (acceptance schedule) that. The User will enter the desired search criteria into the system and the corresponding Site Sign-off (acceptance schedule) that matching the search criteria will be displayed to the user.		
PRE-CONDITION:	The Project page is being displayed to the User		
TRIGGER:	The user wants to search for a projects site sign-off (Acceptance Schedule).		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The User wants to Search for a site sign-off.		
	Step 2: The User hovers over the		**Step 3:** The system displays the dropdown with the following items: • Projects

<table>
<tr>
<td></td>
<td>Construction menu strip.</td>
<td></td>
<td>

Construction Sites
Allocated teams
Compliance Documents
Radio Frequency reports
Snags
Acceptance Documents
Site sign-offs
Commission reports
Site types

</td>
</tr>
<tr>
<td></td>
<td>Step 4: The User clicks on the “Site Sign-offs” dropdown menu item.</td>
<td></td>
<td>Step 5: The System Displays the Site Sign-offs page with the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Site Sign-offs</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Search</td><td>Label</td><td>none</td></tr>
<tr><td>Keyword</td><td>textbox</td><td>To enter a Site Sign-off search criteria</td></tr>
<tr><td>Search</td><td>Button</td><td>Clicked to search for Site Sign-off.</td></tr>
<tr><td>Table</td><td>Table</td><td>This table will hold the information about the Site sign-off on the system.</td></tr>
<tr><td>ID</td><td>Label</td><td>Part of the table: Displays the Acceptance Schedule ID</td></tr>
<tr><td>Name</td><td>Label</td><td>Part of the table: Displays the Site sign-off Name</td></tr>
<tr><td>Construction Site Name</td><td>Label</td><td>Part of the table: Displays the Construction sites name that the site sign-off is associated to.</td></tr>
</table>
</td>
</tr>
</table>

			<table><tr><td>Date Uploaded</td><td>Label</td><td>Part of the table: Displays the date the site sign-off took place and was uploaded.</td></tr></table> This page will display a list of all the Site sign-offs stored on the system. The list is read from the stored Acceptance Schedule entries in the **Acceptance Schedule** and **Construction Site** tables in the database and the relevant attributes previously mentioned are populated with the following table attributes: • Acceptance Schedule ID • Name • Date Uploaded • Construction Site Name (inner joined with the Construction Site table) The table attributes are done via an HttpGet method in the Typescript component and an API request
	Step 6: The User enters the Site Sign-off search criteria. [ALT]		**Step 7:** The system will capture and validate the entered search criteria that was typed into the search textbox. The field is validated and displays a green border around the textbox. **[ALT]**
			Step 8: The system will submit the captured search criteria and search for the site sign-off in the **Acceptance Schedule** table using a parametrised HttpGet request in the typescript component and API request. The system captures and saves the searched result to a list object and displays the search result in the list components, replacing the previous site sign-off list items as they do not follow the searched criteria. [**ALT]**
	Step 9: The user clicks on		**Step 10:** The system displays the Site Sign-off Page where the

<table>
<tr><td></td><td>the list object displayed via the list component.</td><td></td><td>elements are attributed by the Acceptance Schedule and Construction Site tables. The following attributes that are present from the relative tables are as follows:
Acceptance Schedule table:
Name
Description
Date Uploaded
Acceptance Document
Construction Site table:
Site Name
The following page elements are visible and attributed by the attributes previously mentioned:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Site Sign-Off</td><td>Heading</td><td>Page Heading</td></tr>
<tr><td>Name:</td><td>Label</td><td>Displays the name of the Site Sign-off.</td></tr>
<tr><td>Description</td><td>Label</td><td>Displays the description of the site Sign-off.</td></tr>
<tr><td>Date Uploaded</td><td>Label</td><td>Displays the date the site was signed-off on.</td></tr>
<tr><td>Acceptance Schedule</td><td>Button</td><td>Clicked to view the Acceptance Schedule Document.</td></tr>
<tr><td>Return</td><td>Button</td><td>Clicked to return to the Site sign-offs page.</td></tr>
</table></td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 6: The user selects a Site Sign-off from the displayed list present on screen.
Skip to step 9</td></tr>
<tr><td></td><td colspan="3">ALT step 7: The entered information is invalid; the system displays a red border around the search textbox indicating that the entered search criteria is invalid. Return to step 6</td></tr>
<tr><td></td><td colspan="3">ALT Step 8: The search criteria entered into the search textbox does not match any existing records in the database. The system</td></tr>
</table>

	notifies and displays a modal to the user to indicate that the searched for Site Sign-off details do not exist within the database as a null value was obtained from the HttpGet request. **Return to step 6**
CONCLUSION:	This use case concludes when the Acceptance Document page is displayed to the user.
POST-CONDITION:	The Acceptance Document page is displayed to the user with the correct project details
BUSINESS RULES:	All users can search for project details.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.4.25 Capture Commission report

Gohvan Construction			
Author (s): **Michael Vosloo**	**Date: 28/06/2021**	**Version: 1.0**	
USE CASE NAME:	**Capture Commission report**	**USE CASE TYPE**	
USE CASE ID:	**4.25**	**Business Requirements:** ☐	
PRIORITY:	High	**System Analysis:** o	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	Client (PBA)		
PRIMARY THE SYSTEM ACTOR:	Operations Manager (PSA)		
OTHER PARTICIPATING ACTORS:	• None		
OTHER INTERESTED STAKEHOLDERS:	• Director • Financial manager • Quality Control Officer		
DESCRIPTION:	This use case describes the process the Operations Manager will go through when the Commission document is received from the client. The user will navigate to the desired project for the commission report to be uploaded to. The user will then select to upload the commission report via a card that is displayed on the project page. The use case will then prompt the user to select the desired commission report to be uploaded to the system. The system will then capture the commission report and save the report to the database and then display the report to the user that has just been uploaded.		
PRE-CONDITION:	• The Client sends the Commission report to the Operations Manager • The Site sign off has been completed.		
TRIGGER:	The Operations Manager receives the Commission report for the project.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The client sends	**Step 2:** The Operations	

<table>
<tr><td></td><td>the Operations Manager the Commission report via email for the project.</td><td>Manager receives the commission report via email from the client.</td><td></td></tr>
<tr><td></td><td></td><td>Step 3: The Operations Manager wants to upload the commission report to the system for the project.</td><td></td></tr>
<tr><td></td><td></td><td>Step 4: The Operations Manager hovers over Construction menu strip.</td><td>Step 5: The system invokes use case 4.9 “Search Project”</td></tr>
<tr><td></td><td></td><td>Step 6: The Operations Manager clicks on the “Commission Report” card displayed on the project page.</td><td>Step 7: The System displays the Commissions report page with a list of commission reports specific to the searched for project. The page contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Commission Reports</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Search</td><td>Label</td><td>This is the label for the search filter function.</td></tr>
<tr><td>keyword</td><td>Textbox</td><td>To enter a Commission reports name to search for.</td></tr>
<tr><td>Search</td><td>Button</td><td>This button will filter the table for the Searched for Commission report details in the table.</td></tr>
<tr><td>Table</td><td>Table</td><td>This table will hold the information about the Commission</td></tr>
</table></td></tr>
</table>

		report Details.
Commission Report ID	label	Part of the table: Displays the Commission reports ID
Commission Report Name	label	Part of the table: Displays the Commission report name
Date Uploaded	label	Part of the table: Displays the date the Commission report was uploaded.
Project Name	label	Part of the table: Displays the project name that the Commission report is associated to.
Upload	button	This button is clicked when the user wants to upload a new Commission report.

This page will display a table of all the Commission reports stored on the system specific to the data that was obtained in the Search project use case invoke. The table data is read from the **Commission report** table. The data obtained and displayed attributes from the tables that are prevalent to the table listed above are:

- Commission report ID
- Name
- Date uploaded
- Project Name (inner joined via a foreign key)

The table attributes are read via an HttpGet method in the Typescript

<table>
<tr><td></td><td></td><td></td><td>component and an API request.
[ALT]</td></tr>
<tr><td></td><td></td><td>Step 8: Operations Manager clicks the “Upload” button to upload a new Commission report.</td><td>Step 9: The system displays the “Upload Commission report” page with the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Upload Commission report</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Name</td><td>Label</td><td>None</td></tr>
<tr><td></td><td>input</td><td>This input is used to capture the Commission report name.</td></tr>
<tr><td>Date uploaded</td><td>label</td><td>None</td></tr>
<tr><td></td><td>Input (disabled)</td><td>Takes the current date the use case takes place on</td></tr>
<tr><td>Commission report</td><td>Label</td><td>Displays to the user that the following button after this label will be clicked to capture the Commission report.</td></tr>
<tr><td>Upload</td><td>button</td><td>None</td></tr>
<tr><td>Project Name</td><td>Label</td><td>None</td></tr>
<tr><td></td><td>Input (disabled)</td><td>Auto selects the project Name as the search project use case was invoked.</td></tr>
<tr><td>Create</td><td>Button (disabled)</td><td>This button is clicked to create the new Commission report.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>This button is clicked to cancel the creation of the new</td></tr>
</table></td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td></td><td></td><td>Commission report.</td></tr>
</table>
</td></tr>
<tr><td></td><td></td><td>Step 10: The Operations Manager enters the Commission report information in the correct inputs.</td><td>Step 11: The system will capture and validate the entered information. The field is validated and displays a green border around the textbox.
Create Button is enabled.
[ALT]</td></tr>
<tr><td></td><td></td><td>Step 12: The Operations Manager clicks the “Create” button displayed.
[ALT]</td><td>Step 13: The system displays a confirmation modal to the Operations Manager to confirm to upload Commission report. The modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Confirmation required</td><td>heading</td><td>Modal heading</td></tr>
<tr><td>Are you sure you want to upload the Commission report?</td><td>Paragraph</td><td>Queries the user to confirm to upload the Commission report.</td></tr>
<tr><td>Yes</td><td>Button</td><td>Clicked to confirm to upload the Commission report.</td></tr>
<tr><td>No</td><td>Button</td><td>Clicked to reject the upload of the Commission report.</td></tr>
</table>
</td></tr>
<tr><td></td><td></td><td>Step 14: The Operations Manager clicks the “yes” button displayed.
[ALT]</td><td>Step 15: The system uses an SQL insert query performed via an HttpPost request made within the Typescript component to create a new entry within Commission Report table. The system will enter the following attributes obtained from the entered inputs to create the Commission report entry.
Commission Report table:
• Commission Report ID
• Project ID
• Name</td></tr>
</table>

<table>
<tr><td rowspan="4"></td><td></td><td></td><td>Date uploaded (takes the current date the use case takes place on)
Commission Report</td></tr>
<tr><td></td><td></td><td>Step 16: The system displays a modal to the Operations Manager indicating the Commission report has been saved to the system. The modal contains the following elements:<table><tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr><tr><td>Commission Report has been successfully uploaded to the system</td><td>heading</td><td>Modal heading</td></tr><tr><td>ok</td><td>Button</td><td>None</td></tr></table></td></tr>
<tr><td></td><td>Step 17: The Operations Manager clicks the “Ok” button.</td><td></td></tr>
<tr><td colspan="3"></td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 7: The system reads that no Commission Report is associated with the Project.
Skip to step 9</td></tr>
<tr><td></td><td colspan="3">ALT Step 11: Operations Manager clicks the “No” button.
The Use case is terminated</td></tr>
<tr><td></td><td colspan="3">ALT Step 11: The entered information is invalid; the system displays a red border around the incorrect input fields indicating that the entered data is invalid.
Return to step 10</td></tr>
<tr><td></td><td colspan="3">ALT Step 14: The Operations Manager clicks the “No” button displayed.
Return to step 9</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">This use case is concluded when the Operations Manager is notified on the success of the Commission Report being uploaded.</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">The Operations Manager is notified that the Commission report is uploaded to the system.</td></tr>
<tr><td>BUSINESS RULES:</td><td colspan="3">Only the Operations Manager can upload the commission report to the system.</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS</td><td colspan="3">none</td></tr>
</table>

AND SPECIFICATIONS:	
ASSUMPTIONS:	• The client has sent the commission report to the Operations Manager
OPEN ISSUES:	none

1.2.4.26 Search Commission report

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Michael Vosloo</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Search Commission report</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">4.26</td><td>Business Requirements: ☐</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Medium</td><td>System Analysis: o</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td><td>System Design: ☑</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">User (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• Director
• Health and Safety officer
• Quality Control Officer
• Financial Officer
• Operations Manager</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case entails the process the system will go through when searching for a specific Commission report that has been captured and stored on the system relating to the project associated to the Commission report The User will enter the desired search criteria into the system and the corresponding Commission report matching the search criteria will be displayed to the user.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The Client sends the Commission report to the Operations Manager
• The Site sign off has been completed.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The user wants to search for a Commission report.</td></tr>
<tr><td rowspan="3">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The user wants to search for a</td><td></td><td></td></tr>
</table>

<table>
<tr><td></td><td>Commission report.</td><td></td><td></td></tr>
<tr><td></td><td>Step 2: The user hovers over the Construction menu strip.</td><td></td><td>Step 3: The system will display the dropdown with the following items:
Projects
Construction Sites
Allocated teams
Compliance Documents
Radio Frequency reports
Snags
Acceptance Documents
Site sign-offs
Commission reports
Site types</td></tr>
<tr><td></td><td>Step 4: The user clicks on the “Commission reports” dropdown menu strip. [ALT]</td><td></td><td>Step 5: The system displays the Search Commission report page with the following items:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>commission reports</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Search</td><td>Label</td><td>This is the label for the search filter function.</td></tr>
<tr><td>keyword</td><td>Textbox</td><td>To enter a commission report name to search for.</td></tr>
<tr><td>Search</td><td>Button</td><td>This button will filter the table for the Searched for commission report details in the table.</td></tr>
<tr><td>Table</td><td>Table</td><td>This table will hold the information about the Project Details.</td></tr>
<tr><td>Commission report ID</td><td>label</td><td>Part of the table: Displays the commission report ID</td></tr>
</table></td></tr>
</table>

Commission report Name	label	Part of the table: Displays the commission report name
Date Uploaded	label	Part of the table: Displays the date the commission report was uploaded.
Project Name	label	Part of the table: Displays the project name that the commission report is associated to.
Upload	button	This button is clicked when the user wants to upload a new commission report.

This page will display a table of all the commission reports that are stored within the system. The table data is read from the **Commission report** and **project** tables. The data obtained and displayed attributes that are prevalent to the table listed above are:

Commission Report Table:

- Commission report ID
- Name
- Date Uploaded

Project Table:

- Project Name

The table attributes are read Via an HttpGet method in the Typescript component and an API request.

<table>
<tr><td></td><td>Step 6: The user enters the Commission report search criteria in the search text box. [ALT]</td><td></td><td>Step 7: The system will capture and validate the entered information that was typed into the search textbox. The field is validated and displays a green border around the textbox. [ALT]</td></tr>
<tr><td></td><td></td><td></td><td>Step 8: The system will submit the captured search criteria and search for the commission report details in the Commission Report table using a parametrised HttpGet request in the typescript component and API request. The system captures and saves the searched result to a list object and displays the search result in the table components, replacing the previous Commission report table items as they do not follow the searched criteria. [ALT]</td></tr>
<tr><td></td><td>Step 9: The user clicks on the desired Commission report details displayed on the table.</td><td></td><td>Step 10: The system displays the Commission report page that is attributed by the Commission Report and Project tables according to the searched for commission report details. The commission report details are read from the Commission Report and Project tables and passed via the API control to the type script component to then be displayed on the Commission report page elements. The Commission report page has the following elements displayed:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Commission Report</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Name</td><td>Label</td><td>Displays the Commission report name.</td></tr>
<tr><td>Commission Report</td><td>Label</td><td>Displays to the user that the following button after this label will display</td></tr>
</table>
</td></tr>
</table>

					the Commission report.
			View	button	When this button is clicked the system will display the Commission report to the user.
			Project Name	Label	Displays the project name that the Commission report is linked to.
			Date Uploaded	label	Displays the date the Commission report was uploaded.
			Return	Button	This button is clicked to return back to the Commission reports page.
ALTERNATE COURSES:	**ALT Step 4**: The user selects the "Commission report" element on the "Project" page, the system will **skip to step 5** but only display the Commission report associated to that project that was being displayed. The Commission report is linked to the project which entails that the **Commission report** and **Project** tables have respective foreign keys within the **Commission report** table. The Commission report are retrieved via an inner join function in SQL for matching primary keys and the data is retrieved via an API and typescript component. **Skip to step 5**				
	ALT Step 6: The user selects a Commission report from the displayed list present on the screen. **Skip to step 9**				
	ALT Step 7: The entered search criteria is invalid; the system displays a red border around the search textbox indicating that the entered search criteria is invalid. **Return to step 6**				
	ALT Step 8: The search criteria entered into the search textbox does not match any existing records in the database. The system notifies and displays a modal to the user to indicate that the searched for Commission report details do not exist within the database as a null value was obtained from the HttpGet request. **Return to step 6**				

CONCLUSION:	This use called is concluded when the Commission report page is displayed to the user.
POST-CONDITION:	The correct Commission report is displayed to the user.
BUSINESS RULES:	All users can search for a Commission report linked to a project.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.4.27 Create Site Type

Gohvan Construction			
Author (s): Michael Vosloo	**Date: 28/06/2021**	**Version: 1.0**	
USE CASE NAME:	**Create Site Type**	**USE CASE TYPE**	
USE CASE ID:	**4.27**	**Business Requirements:** ☐	
PRIORITY:	Low	**System Analysis:** o	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	Operations Manager (PBA)		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	• Quality Control Officer • Director • Procurement and logistics officer • Financial Officer		
DESCRIPTION:	This use case entails the process the system will go through when the Operations Manager wants to create a new Site type for the system. The system will prompt the user to enter the new site type details info into the correct fields. The system I'll capture and validate the entered details, the system will then store the new site type into the system.		
PRE-CONDITION:	User must be logged in.		
TRIGGER:	The Operations Manager wants to create a new Site type.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The Operations Manager wants to create a new site type.		
	Step 2: The Operations Manager		**Step 3**: The system displays the dropdown menu with the following items:

<table>
<tr>
<td></td>
<td>hovers over the Construction menu strip.</td>
<td></td>
<td>

Projects
Construction Sites
Allocated teams
Compliance Documents
Radio Frequency Reports
Snags
Acceptance Documents
Site Sign-offs
Commission reports
Site types

</td>
</tr>
<tr>
<td></td>
<td>Step 4: The Operations Manager clicks on the “Site Types” dropdown menu item.</td>
<td></td>
<td>Step 5: The system displays the Site type details page with the followings page elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Site Types</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Search</td><td>Label</td><td>This is the label for the search filter function.</td></tr>
<tr><td>Keyword</td><td>Textbox</td><td>To enter a site type search criterion to search for.</td></tr>
<tr><td>Search</td><td>Button</td><td>This button will filter the table for the Searched for site type details in the table.</td></tr>
<tr><td>Table</td><td>table</td><td>This table will hold the information about the site type details.</td></tr>
<tr><td>Site Type ID</td><td>Label</td><td>Part of the table: Displays the site type ID.</td></tr>
<tr><td>Site Type Name</td><td>Label</td><td>Part of the table: Displays the Site Type name.</td></tr>
<tr><td>View</td><td>Button</td><td>Part of the table: the user clicks this button to view the details of the corresponding site type.</td></tr>
<tr><td>New Site Type</td><td>Button</td><td>This button is clicked when the user wants to create a new site type.</td></tr>
</table>
</td>
</tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>This page will display a table of all the site types that are stored in the system. The table data is read from the Site Type table. The data obtained and displayed attributes from the tables that are prevalent to the table listed above are:
Site Type ID
Site Type Name
The table attributes are read via an HttpGet method in the typescript component and an API request.</td></tr>
<tr><td></td><td>Step 6: The Operations Manager clicks on the “New Site Type” button.</td><td></td><td>Step 7: The system displays a modal with the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Create Site Type</td><td>Heading</td><td>Modal Heading</td></tr>
<tr><td>Site Type Name</td><td>label</td><td>Indication to the user that the site type Name must be inputted.</td></tr>
<tr><td></td><td>Input</td><td>Corresponding input for the user to input the Site type Name</td></tr>
<tr><td>Site Type Description</td><td>Label</td><td>Indication to the user that the site type description must be inputted.</td></tr>
<tr><td></td><td>input</td><td>Corresponding input for the user to input the Site type description</td></tr>
<tr><td>Create</td><td>Button (disabled)</td><td>User clicks this button to create site type.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>User clicks this to cancel the creation of the new site type.</td></tr>
</table></td></tr>
<tr><td></td><td>Step 8: The Operations Manager enters the Site Type details in the</td><td></td><td>Step 9: The system captures and validates the provided information entered into the fields:
Site Type Name
Site Type Description</td></tr>
</table>

<table>
<tr><td rowspan="5"></td><td>relevant input fields.</td><td></td><td>Create Button is enabled.
[ALT]</td></tr>
<tr><td>Step 10: The Operations Manager clicks the “Create” button [ALT]</td><td></td><td>Step 11: The system loads the confirmation modal and displays it to the user to prompt the confirmation of creating the new site type. The Modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Confirmation Site Type creation</td><td>Label</td><td></td></tr>
<tr><td>Yes</td><td>Button</td><td>This button is clicked to create the new site type.</td></tr>
<tr><td>No</td><td>Button</td><td>This button is clicked to cancel the creation of the new site type.</td></tr>
</table></td></tr>
<tr><td>Step 12: The Operations Manager clicks on the “Yes” Button
[ALT]</td><td></td><td>Step 13: The system saves the site type details to the Site Type table. The site type details are created in the Site Type tables via an HttpPost method sent in the API which is called through the Typescript component. The following attributes are created in the Site Type table:
Site Type ID
Site Type Name
Site Type Description</td></tr>
<tr><td></td><td></td><td>Step 14: The System displays an alert to notify on the successful creation of the Site type.</td></tr>
<tr><td colspan="3"></td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 9: The entered information is invalid; the system displays a red border around the search textbox indicating that the entered search criteria is invalid.
Return to step 8</td></tr>
<tr><td></td><td colspan="3">ALT Step 10: The Operations Manager clicks on the “Cancel” button. The create site type modal is closed.
The Use case is terminated</td></tr>
<tr><td></td><td colspan="3">ALT Step 12: The Operation Manager clicks on the “No” button. The confirmation modal is closed.
Return to step 8</td></tr>
</table>

CONCLUSION:	This use case is concluded when the Operations Manager is alerted that the site type was successfully created.
POST-CONDITION:	The new Site type details are stored in the database.
BUSINESS RULES:	Only the Operations Manager can create a new site type.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.4.28 Search Site type

Gohvan Construction			
Author (s): Michael Vosloo	**Date: 28/06/2021**		**Version: 1.0**
USE CASE NAME:	**Search Site type**		**USE CASE TYPE**
USE CASE ID:	**4.28**		**Business Requirements:** ☐
PRIORITY:	Medium		**System Analysis:** o
SOURCE:	Gohvan Construction System		**System Design:** ☑
PRIMARY BUSINESS ACTOR:	User (PBA)		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	• Operations Manager • Procurement and Logistics Officer • Director • Financial Officer		
DESCRIPTION:	This use case entails the process the system will go through when searching for a specific site type that has been captured and stored on the system. The User will enter the desired search criteria into the system and the corresponding site type matching the search criteria will be displayed to the user.		
PRE-CONDITION:	The site type must exist on the system.		
TRIGGER:	The user wants to search for a site type.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The User wants to search for a site type.		
	Step 2: The user hovers over the Construction Menu strip.		**Step 3**: The system displays the dropdown menu with the following items: • Projects • Construction Sites • Allocated teams

<table>
<tr>
<td></td>
<td></td>
<td></td>
<td>

Compliance Documents
Radio Frequency Reports
Snags
Acceptance Documents
Site Sign-offs
Commission reports
Site types

</td>
</tr>
<tr>
<td></td>
<td>Step 4: The user clicks on the “Site Types” dropdown menu item.</td>
<td></td>
<td>Step 5: The system displays the Site type details page with the followings page elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Site Types</td><td>Heading</td><td>Page heading</td></tr>
<tr><td>Search</td><td>Label</td><td>This is the label for the search filter function.</td></tr>
<tr><td>Keyword</td><td>Textbox</td><td>To enter a site type search criterion to search for.</td></tr>
<tr><td>Search</td><td>Button</td><td>This button will filter the table for the Searched for site type details in the table.</td></tr>
<tr><td>Table</td><td>table</td><td>This table will hold the information about the site type details.</td></tr>
<tr><td>Site Type ID</td><td>Label</td><td>Part of the table: Displays the site type ID.</td></tr>
<tr><td>Site Type Name</td><td>Label</td><td>Part of the table: Displays the Site Type name.</td></tr>
<tr><td>View</td><td>Button</td><td>Part of the table: the user clicks this button to view the details of the corresponding site type.</td></tr>
<tr><td>New Site Type</td><td>Button</td><td>This button is clicked when</td></tr>
</table>
</td>
</tr>
</table>

			the user wants to create a new site type. This page will display a table of all the site types that are stored in the system. The table data is read from the **Site Type** table. The data obtained and displayed attributes from the tables that are prevalent to the table listed above are: • Site Type ID • Site Type Name The table attributes are read via an HttpGet method in the typescript component and an API request.
	Step 6: The user enters the Site Type search criteria in the search textbox. **[ALT]**		**Step 7:** The system will capture and validate the entered information that was typed into the search textbox. The field is validated and displays a green border around the textbox. **[ALT]**
			Step 8: The system will submit the captured search criteria and search for the site type details in **the Site Type** table using a parameterised HttpGet request in the typescript component and API request. The system captures and saves the searched result to a list object and displays the search result in the table components, replacing the previous Site type table items as they do not follow the search criteria. **[ALT]**
	Step 9: The user clicks on the desired Site Type details displayed on the table.		**Step 10:** The system displays a modal to the user that displays the Site type details. The modal elements are attributed by the selected site type information that is pulled from the **Site Type** table. The data is read via an HttpGet request done in the API which is called within the Typescript component. The following data is read from the **Site Type** table: • Site Type ID

<table>
<tr><td></td><td></td><td></td><td>

Site Type Name
Site Type Description

The following modal elements are attributed with the read data and is displayed to the user:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Create Site Type</td><td>Heading</td><td>Modal Heading</td></tr>
<tr><td>Site Type Name</td><td>label</td><td>None</td></tr>
<tr><td></td><td>Input (disabled)</td><td>Displays the Site type name.</td></tr>
<tr><td>Site Type Description</td><td>Label</td><td>None</td></tr>
<tr><td></td><td>Input (disabled)</td><td>Displays the Site type description.</td></tr>
<tr><td>Edit</td><td>Edit</td><td>User clicks this button to edit the site type.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>User clicks this to close the view modal.</td></tr>
</table>
</td></tr>
<tr><td rowspan="3">ALTERNATE COURSES:</td><td colspan="3">ALT Step 6: The user selects a site type from the displayed list present on screen.
Skip to step 9</td></tr>
<tr><td colspan="3">ALT Step 7: The entered information is invalid; the system displays a red border around the search textbox indicating that the entered search criteria is invalid.
Return to step 6</td></tr>
<tr><td colspan="3">ALT Step 8: The search criteria entered into the search textbox does not match any existing records in the database. The system notifies and displays a modal to the user to indicate that the searched for site type details do not exist within the database as a null value was obtained from the HttpGet request.
Return to step 6</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">This use case concludes when the site type details modal is displayed to the user.</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">The correct site type details are displayed to the user.</td></tr>
<tr><td>BUSINESS RULES:</td><td colspan="3">All users can search for a site type.</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS</td><td colspan="3">none</td></tr>
</table>

AND SPECIFICATIONS:	
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.4.29 Edit Site Type

Gohvan Construction			
Author (s): Michael Vosloo	**Date: 28/06/2021**	**Version: 1.0**	
USE CASE NAME:	**Edit Site Type**	**USE CASE TYPE**	
USE CASE ID:	**4.29**	**Business Requirements:** ☐	
PRIORITY:	Low	**System Analysis:** o	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	Operations Manager (PBA)		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	• Operations Manager • Procurement and Logistics Officer • Director • Financial Officer		
DESCRIPTION:	This use case entails the process the system will go through when the Operations Manager wants to edit the details of an existing site type on the system. The system will prompt the user to search for a site type that exists on the system. The system will display the details of the searched site type to the Operations Manager. The Operations Manager will then enter the new details of the site type, the system then captures and validates that the site type details are correct and valid. The edited site type details will be saved to the system and the Operations Manager.		
PRE-CONDITION:	The user must be logged in. An existing site type must exist on the system.		
TRIGGER:	The Operations Manager wants to edit a site type.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The Operations Manager wants to edit		

<table>
<tr><td></td><td>a site type's details.</td><td></td><td></td></tr>
<tr><td></td><td>Step 2: The Operations Manager Hovers over the Construction menu strip.</td><td></td><td>Step 3: The system will display the dropdown with the following items:
Projects
Construction Sites
Allocated teams
Compliance Documents
Radio Frequency reports
Snags
Acceptance Documents
Site Sign-offs
Commission reports
Site types</td></tr>
<tr><td></td><td>Step 4: The Operations Manager clicks on the "Site types" dropdown menu item.</td><td></td><td>Step 5: The system invokes use case 4.29 "Search Site type"</td></tr>
<tr><td></td><td>Step 6: The Operations Manage clicks on the "Edit" button displayed</td><td></td><td>Step 7: The System enabled all the inputs that are displayed on the modal. The following elements are present on the modal with pre-populated information due to the data obtained from the "search Site type" invoke:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Edit Site Type</td><td>Heading</td><td>Modal Heading</td></tr>
<tr><td>Site Type Name</td><td>label</td><td>None</td></tr>
<tr><td></td><td>Input</td><td>Corresponding input for the user to input the Site type Name</td></tr>
<tr><td>Site Type Description</td><td>Label</td><td>None</td></tr>
<tr><td></td><td>input</td><td>Corresponding input for the user to input the Site type description</td></tr>
<tr><td>Confirm edit</td><td>Button (disabled)</td><td>This button is changed once the edit button has been</td></tr>
</table></td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td></td><td></td><td>clicked to indicate that the user must confirm to capture the new details of the edited site type.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>User clicks this to cancel the editing of the new site type.</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 8: The Operations Manager enters the new site type details.</td><td></td><td>Step 9: The system captures and validates the provided information entered into the fields:
Site type name
Site type description
Confirm edit Button is enabled.
[ALT]</td></tr>
<tr><td></td><td>Step 10: The Operations Manager clicks the “Confirm Edit” button
[ALT]</td><td></td><td>Step 11: The system loads the confirmation modal and displays it to the Operations Manager to prompt the confirmation of editing the new site type details. The Modal contains the following elements:
<table>
<tr><th>Input name:</th><th>Control Type:</th><th>Notes:</th></tr>
<tr><td>Confirmation to edit Site type.</td><td>Label</td><td></td></tr>
<tr><td>Yes</td><td>Button</td><td>This button is clicked to edit the details of the site type.</td></tr>
<tr><td>No</td><td>Button</td><td>This button is clicked to cancel the editing of the new site type details.</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 12: The Operations Manager clicks the “yes” button displayed.
[ALT]</td><td></td><td>Step 13: The system edits the site type details to the Site Type table. The site type details are edited in the Site Type table via an HttpPut method sent in the API which is called through the Typescript component. The following attributes are edited in the Site Type table:
Site Type Name
Site Type Description</td></tr>
</table>

			Step 14: The System displays an alert to notify on the successful editing of the Site type.
ALTERNATE COURSES:	**ALT Step 9:** The entered information is invalid; the system displays a red border around the search textbox indicating that the entered search criteria is invalid. **Return to step 8**		
	ALT Step 10: The Operations Manger clicks on the “Cancel” button. **The Use case is terminated**		
	ALT Step 12: The Operations Manger clicks on the “No” button**.** The confirmation modal is closed. **Return to step 7**		
CONCLUSION:	This use case is concluded when the Operations Manager is alerted that the site type details were successfully edited.		
POST-CONDITION:	The site type details were successfully updated.		
BUSINESS RULES:	Only the Operations Manager can edit the site type details.		
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none		
ASSUMPTIONS:	none		
OPEN ISSUES:	none		

1.2.5. Invoicing Subsystem

1.2.5.1 Search Invoice

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><th>Author (s): Vincent Yu</th><th colspan="2">Date: 28/06/2021</th><th>Version: 3.0</th></tr>
<tr><th>USE CASE NAME:</th><th colspan="2">Search Invoice</th><th>USE CASE TYPE</th></tr>
<tr><td>USE CASE ID:</td><td colspan="2">5.1</td><td rowspan="3">Business Requirements: □
System Analysis: □
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Financial Officer (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">Managing Director</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the event when the Financial Officer wants to view an Invoice associated to a project. The system will display the Invoice and the details associated with the Invoice.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The Financial Officer needs to be logged onto the system.
• Invoice already generated and stored in the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">. The Financial Officer wants to search for a specific Invoice.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td>Step 1:
The Financial Officer clicks on the “Menu” to navigate to the click on the Invoicing tab and then clicking</td><td></td><td>Step 2:
The system responds by loading the “Search Invoice” screen already populated with the Invoice table information using the MySql Server in the .NET Controller to read the Invoice table using LINQ and LAMBDA Queries.
<table><tr><th>Table</th><th>Attribute</th></tr><tr><td>Invoice</td><td>Invoice Number</td></tr><tr><td></td><td>Date Sent</td></tr><tr><td></td><td>Date paid</td></tr><tr><td></td><td>Invoice Receipt Number</td></tr></table></td></tr>
</table>

<table>
<tr>
<td></td>
<td>“Search Invoice”</td>
<td></td>
<td>
The “Search Invoice” screen has the following controls:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Search Invoice</td><td>Heading</td><td>None</td></tr>
<tr><td>Search Bar</td><td>Textbox</td><td>This is used to get the search parameter from the Financial Officer.</td></tr>
<tr><td>Sort</td><td>Drop down</td><td>This is used to get the filter option from the Financial Officer.</td></tr>
<tr><td>Results Table</td><td>Table</td><td>This table will contain the saved Invoice information per row.</td></tr>
<tr><td>Download Invoice</td><td>Button</td><td>Each record in the Results Table will have a Download Invoice button. This allows the Financial Officer to download the Invoice.</td></tr>
<tr><td>Back</td><td>Button</td><td>This is used to help the Financial Officer navigate back.</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 3:
The Financial Officer enters the search parameter [Project ID or Project Name] in the Search Bar control.</td>
<td></td>
<td>
Step 4:
The system captures and validates the search parameter [Project ID or Project Name] supplied by the Financial Officer, against the Project table. [ALT]
<table>
<tr><th>Input Name</th><th>Attribute in Table</th><th>Notes</th></tr>
<tr><td>Search Parameter [Project ID]</td><td>Project_ID</td><td>Required, maximum of 13 characters.</td></tr>
</table>
</td>
</tr>
</table>

<table>
<tr><td rowspan="5"></td><td></td><td></td><td>
<table>
<tr><td>Search Parameter [Project Name]</td><td>Project_ Name</td><td>Required, maximum of 128 characters</td></tr>
</table>
</td></tr>
<tr><td></td><td></td><td>Step 5:
The system Invokes use case 4.9 Search Project.</td></tr>
<tr><td></td><td></td><td>Step 6:
The system retrieves the Invoice record result based on the search parameter supplier by the Financial Officer from the Invoice table:
• Invoice Number
• Invoice
• Date sent
• Date paid
The Invoice record is displayed in the Results Table control.</td></tr>
<tr><td>Step 7:
The Financial Officer clicks the “Download Invoice” button.</td><td></td><td>Step 8:
The system retrieves the following attribute(s) from the Invoice table:
<table>
<tr><th>Table</th><th>Attribute</th><th>Notes</th></tr>
<tr><td>Invoice</td><td>Invoice</td><td>This is the stored PDF Invoice document</td></tr>
</table>
This document is then sent to the Financial Officer as requested.</td></tr>
<tr><td colspan="3"></td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 4:
The entered search parameter is not in the correct format or not completed.

The system displays a Dialog of the validation error to the financial officer.
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Validation Error</td><td>Label</td><td>None</td></tr>
<tr><td>Error Message</td><td>Label</td><td>This is used to inform the Financial Officer about the validation error.</td></tr>
<tr><td>OK</td><td>Button</td><td>This is used by the Financial Officer to acknowledge the error.</td></tr>
</table>
The Financial Officer clicks the “OK” button
Return to Step 2</td></tr>
<tr><td colspan="4">ALT Step 4:</td></tr>
</table>

<table>
<tr><td></td><td>There is no invoice in the system that corresponds to the entered search parameter.

The system displays the following control to inform the financial officer that no record was found based on the provided search parameter:
<table><tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>No Results Found</td><td>Label</td><td>This is used to inform the financial officer no record was found.</td></tr></table>
The financial officer clears the Search Bar control
Return to Step 2</td></tr>
<tr><td>CONCLUSION:</td><td>The use case concludes when the Invoice is displayed.</td></tr>
<tr><td>POST-CONDITION:</td><td>• The Invoice table was searched for an entry that corresponds with the search parameter
• The invoice is displayed</td></tr>
<tr><td>BUSINESS RULES:</td><td>• Only the Financial Officer has access to searching invoices.</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td>None</td></tr>
<tr><td>ASSUMPTIONS:</td><td>None</td></tr>
<tr><td>OPEN ISSUES:</td><td>None</td></tr>
</table>

1.2.5.2 Send Invoice

Gohvan Construction			
Author (s): Vincent Yu	**Date: 28/06/2021**	**Version: 3.0**	
USE CASE NAME:	**Send Invoice**		
USE CASE ID:	**5.2**	**Business Requirements:** □	
PRIORITY:	High	**System Analysis:** □	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	Financial Officer (PBA)		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	Managing Director		
DESCRIPTION:	This use case describes the event when The Financial Officer wants to send an invoice to the client. The system will generate an invoice, associated to a specific project and send it to the Financial Officer. The Financial Officer will check the invoice to ensure the invoice is accurate. Once the check is complete the Invoice is sent to the client for approval.		
PRE-CONDITION:	• Financial Officer needs to be logged on to the system. • Financial Officer needs to be in the **specific project** to download invoices. • **An RF Report needs to be present** to generate the initial claim invoice (90%). • **Both RF Report and Client site sign-off need to be present** to generate the final claim invoice (10%).		
TRIGGER:	. The Financial offer would like to send an invoice to the client.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The Financial Officer clicks on the "**Invoicing**" card on the		**Step 2:** The system responds by loading the "**Select Invoice Type**" modal. This modal prompts the financial officer to choose the type of Invoice to be generated/sent. This screen has the following controls:

<table>
<tr>
<td></td>
<td>project's home screen. And then clicks the "Send Invoice" button</td>
<td></td>
<td>
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Send Invoice</td><td>Heading</td><td>None</td></tr>
<tr><td>Send Invoice Description</td><td>Label</td><td>This is used to inform the financial officer about the use case instructions</td></tr>
<tr><td>Initial Invoice</td><td>Button</td><td>This is used to select the type of invoice to be sent.</td></tr>
<tr><td>Final Invoice</td><td>Button</td><td>This is used to select the type of invoice to be sent.</td></tr>
<tr><td>Back</td><td>Button</td><td>This is used to help the User navigate back.</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 3:
The financial officer clicks the Initial Invoice button on the Invoice Type screen.
[ALT]</td>
<td></td>
<td>Step 4:
The system reads the client's Purchase Order (PO) PO Amount, Client details, Site details. associated with the project from the Client Purchase Order, Client and Site table with the following attributes:
• PO Amount
• Client Name
• Client Address
• Vat no.
• Registration no.
• Site name
Using these attributes, the system generates an Initial Claim Invoice. This Initial Claim caters for 90% of the total invoice amount.
[ALT]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Step 5:
The system responds by loading the "Confirm Invoice" modal The system prompts the financial officer to confirm the invoice.
The system displays the generated Invoice using the following controls:</td>
</tr>
</table>

			<table><tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Send Initial/ Final Invoice</td><td>Heading</td><td>Heading displays the option selected.</td></tr><tr><td>Request Confirmation</td><td>Label</td><td>This is used to inform the financial to confirm the displayed invoice.</td></tr><tr><td>PDF Viewer</td><td>PDF Viewer (<iframe> or link <a>)</td><td>This is used to select the type of invoice to be sent.</td></tr><tr><td>Confirm</td><td>Button</td><td>This is used to confirm that the invoice is correct.</td></tr><tr><td>Download</td><td>Button</td><td>This is used to download the displayed invoice pdf. This button is disabled until the “Confirm” button is clicked.</td></tr><tr><td>Cancel</td><td>Button</td><td>This is used to help the financial officer confirm that the invoice is not correct. And it allows navigate back.</td></tr></table>
	Step 6: The financial officer clicks the “**Confirm**” button. **[ALT]**		**Step 7:** The system saves the invoice in the Invoice table: <table><tr><th>Table</th><th>Attribute</th><th>Notes</th></tr><tr><td>Invoice</td><td>Invoice Number</td><td>This the primary key used to track invoices</td></tr><tr><td></td><td>Date Sent</td><td>This is the date the invoice was sent to the client.</td></tr></table>

			<table><tr><td></td><td>**Invoice Type**</td><td>This indicates what type of invoice the invoice is.</td></tr><tr><td></td><td>**Invoice**</td><td>This is the stored PDF Invoice document</td></tr></table> The system displays an alert informing the financial officer that the invoice has been confirmed and successfully saved. <table><tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>**Alert**</td><td>**Alert**</td><td>None</td></tr><tr><td>**Invoice Confirmed**</td><td>**Label**</td><td>This is used to inform the financial officer that the invoice was confirmed.</td></tr><tr><td>**OK**</td><td>**Button**</td><td>This is used to acknowledge the alert.</td></tr></table> After this “Invoice Confirmed” alert the Download button should be enabled.
	Step 8: The financial Officer clicks on the “**OK**” button.		
	Step 9: The financial Officer clicks on the “**Download**” button.		**Step 10:** The system retrieves the following attribute(s) from the **Invoice** table: <table><tr><th>Table</th><th>Attribute</th><th>Notes</th></tr><tr><td>**Invoice**</td><td>**Invoice**</td><td>This is the stored PDF Invoice document</td></tr></table> This document is then sent to the Financial Officer as requested.

ALTERNATE COURSES:	**ALT Step 3:** The financial officer clicks the "**Final Invoice**" button on the Select Invoice Type modal. **ALT Step 4:** The system reads the client's Purchase Order (PO) PO Amount, Client details, Site details. associated with the project from the **Client Purchase Order, Client and Site** table with the following attributes: • PO Amount • Client Name • Client Address • Vat no. • Registration no. • Site name Using these attributes, the system generates a **Final** Claim Invoice. This **Final** Claim caters for the remaining **10%** of the total invoice amount. **Continue to Step 5**
	ALT Step 6: The financial officer confirms that the invoice is not correct. **ALT Step 7:** The financial officer clicks the cancel button. **Terminate Use Case.**
CONCLUSION:	This use case concludes when the invoice is sent to the financial officer.
POST-CONDITION:	• Invoice is sent to the client for approval and acceptance.
BUSINESS RULES:	• Every project should only have two invoices (Initial 90% and Final 10% Claims). • RF Report for the project needs to be uploaded, before Invoicing can begin. • Only the Financial Officer has access to generating invoices.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.5.3 Capture Proof of Payment

Gohvan Construction			
Author (s): Vincent Yu	**Date: 28/06/2021**	**Version: 3.0**	
USE CASE NAME:	**Capture Proof of Payment**	**USE CASE TYPE**	
USE CASE ID:	**5.3**	**Business Requirements:** □	
PRIORITY:	High	**System Analysis:** □	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	Financial Officer (PBA)		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	Managing Director		
DESCRIPTION:	This use case describes the event where the Financial Officer is notified on the Taulia's Portal that an Invoice has been paid to the company. The Financial Officer needs to capture this status on the Taulia Portal as proof of payment (POP).		
PRE-CONDITION:	• Financial Officer has to be logged into the system. • Financial Officer needs to be in the **specific project** to capture proofs of payment. • The Invoice has been approved by client and the receipt number is associated to the corresponding invoice.		
TRIGGER:	The Financial Officer wants to capture the status of an already processed invoice.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:** Manual Action	Automated Action
	Step 1: The Financial Officer clicks on the "**Invoicing**" card on the project's home		**Step 2:** The system invokes use case 5.1 "**Search Invoice**" To retrieve the Invoices associated with the specific project.

<table>
<tr><td></td><td>screen. And then clicks the “Capture Proof of Payment” button</td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td>
Step 3:
The system responds by loading the “Select an Invoice” screen already populated with the Invoice table information using the MySql Server in the .NET Controller to read the Invoice table using LINQ and LAMBDA Queries.
<table>
<tr><th>Table</th><th>Attribute</th></tr>
<tr><td rowspan="4">Invoice</td><td>Invoice Number</td></tr>
<tr><td>Date Sent</td></tr>
<tr><td>Date paid</td></tr>
<tr><td>Invoice Receipt Number</td></tr>
</table>
The system prompts the financial officer to select an invoice they want to capture a POP for.
The “Select an Invoice” screen has the following controls:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Select Invoice</td><td>Heading</td><td>None</td></tr>
<tr><td>Please Select</td><td>Label</td><td>This is used to prompt the financial officer to select an invoice.</td></tr>
<tr><td>Results Table</td><td>Table</td><td>This table will contain the saved Invoice information per row.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>This is used to help the Financial Officer navigate back.</td></tr>
</table>
[ALT]
</td></tr>
<tr><td></td><td>Step 4:</td><td></td><td>Step 5:</td></tr>
</table>

<table>
<tr>
<td></td>
<td>The Financial Officer clicks on a record in the Results Table.</td>
<td></td>
<td>The system responds by loading the “Capture Proof of Payment” screen which has the following controls:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Capture Proof of Payment</td><td>Heading</td><td>None</td></tr>
<tr><td>Description</td><td>Label</td><td>This is used to inform the financial officer about what is required.</td></tr>
<tr><td>Upload</td><td>File Upload</td><td>This is used to upload the POP.</td></tr>
<tr><td>POP Details</td><td>Label</td><td>None</td></tr>
<tr><td>POP Date</td><td>Label & Date Input</td><td>These controls are used to get the date of the POP</td></tr>
<tr><td>POP Amount</td><td>Label & Textbox</td><td>These controls are used to get the POP’s Amount.</td></tr>
<tr><td>Submit</td><td>Button</td><td>This is used to submit the provided details.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>This is used to help the Financial Officer navigate back.</td></tr>
</table>
The system prompts the financial officer to upload the corresponding POP and provide its details for the selected Invoice record.</td>
</tr>
<tr>
<td></td>
<td>Step 6:
The financial officer uploads the POP by using the File Upload Control; and provides the POP date</td>
<td></td>
<td>Step 7:
The system captures and validates the provided details supplied by the Financial Officer. [ALT]
<table>
<tr><th>Input Name</th><th>Attribute in Table</th><th>Notes</th></tr>
<tr><td>Proof of</td><td>POP</td><td>Required, of type image.</td></tr>
</table></td>
</tr>
</table>

and Amount using the **Date** and **Textbox** controls. And then clicks the Submit button to submit the information.		**Payment**		
		Proof of Payment Date	**POP_Date**	Required, of type date
		Proof of Payment Amount	**POP_Amount**	Required, maximum of 128 characters
		Step 8: The system prompts the financial to confirm the provided information, using the "**Proof of Payment Confirmation**" modal which has the following controls:		
		Control Name	**Control Type**	**Notes**
		Confirm Proof of Payment	**Heading**	None
		Proof of Payment	**Image**	This is the screenshot uploaded by the financial officer.
		POP_ID	**Label**	This is used to display the POP ID info.
		POP_Date	**Label**	This is used to display the POP Date info.
		POP_Amount	**Label**	This is used to display the POP Amount info.
		Confirm	**Button**	This is used to confirm the POP.
		Cancel	**Button**	This is used to help the Financial Officer navigate back.
	Step 9: The Financial Officer clicks the "**Confirm**"		**Step 10:** The system stores the following attribute(s) in the **Proof of Payment** table:	
			Table	**Attribute**	**Notes**

<table>
<tr>
<td></td>
<td>button.
[ALT]</td>
<td></td>
<td>
<table>
<tr><td rowspan="4">Proof of Payment</td><td>POP_ID</td><td>This is the primary key used to track Proofs of payments.</td></tr>
<tr><td>POP_Date</td><td>This is the date the POP was captured.</td></tr>
<tr><td>POP_Amount</td><td>This the POP’s amount</td></tr>
<tr><td>Proof of Payment</td><td>This is the stored screenshot image taken from the Taulia payment portal.</td></tr>
</table>
The system prompts the financial officer that the POP was captured successfully with an alert:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Alert</td><td>Alert</td><td>None</td></tr>
<tr><td>Proof of Payment Captured</td><td>Label</td><td>This is used to inform the financial officer that the POP was captured.</td></tr>
<tr><td>OK</td><td>Button</td><td>This is used to acknowledge the alert.</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 11:
The financial officer clicks the “OK” button.</td>
<td></td>
<td></td>
</tr>
</table>

<table>
<tr><td>ALTERNATE COURSES:</td><td>
ALT Step 3:
There is no invoice in the system that corresponds to the entered search parameter.

The system displays the following control to inform the financial officer that no record was found based on the provided search parameter:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>No Results Found</td><td>Label</td><td>This is used to inform the financial officer no record was found.</td></tr>
</table>
Terminate Use Case.
</td></tr>
<tr><td></td><td>
ALT Step 7:
The provided details are not in the correct format or not completed.

The system displays a Dialog of the validation error to the financial officer.
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Validation Error</td><td>Label</td><td>None</td></tr>
<tr><td>Error Message</td><td>Label</td><td>This is used to inform the Financial Officer about the validation error.</td></tr>
<tr><td>OK</td><td>Button</td><td>This is used by the Financial Officer to acknowledge the error.</td></tr>
</table>
The Financial Officer clicks the “OK” button
Return to Step 5
</td></tr>
<tr><td></td><td>
ALT Step 9:
The financial officer clicks the cancel button.
Return to Step 5
</td></tr>
<tr><td>CONCLUSION:</td><td>The use case concludes when the Proof of Payment is displayed.</td></tr>
<tr><td>POST-CONDITION:</td><td>

The Proof of Payment table was searched for an entry that corresponds with the search parameter
The proof of payment is displayed

</td></tr>
</table>

BUSINESS RULES:	• Only the Financial Officer has access to searching for proof of payments.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None
ASSUMPTIONS:	None
OPEN ISSUES:	None

1.2.5.4 Capture Claim

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s): Vincent Yu</td><td colspan="2">Date: 28/06/2021</td><td>Version: 3.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Capture Claim</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">5.4</td><td rowspan="3">Business Requirements: □
System Analysis: □
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Financial Officer (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">Managing Director</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">The Client has approved the received invoice and sends a receipt number associated to the invoice. The financial officer needs to log this receipt number on Gohvan system. The financial officer will also use this receipt number to process the claim on Taulia.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The Financial Officer needs to be logged onto the system.
• Invoice already generated and sent to client.
• Financial Officer needs to be in the specific project to capture claims.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The Client has approved the invoice and sent the Receipt number.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td>Step 1:
The Financial Officer clicks on the “Invoicing” card on the project’s home</td><td></td><td>Step 2:
The system invokes use case 5.1 “Search Invoice” To retrieve the Invoices associated with the specific project.</td></tr>
</table>

<table>
<tr><td></td><td>screen. And then clicks the “Capture Claim” button</td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td>
Step 3:
The system responds by loading the “Select an Invoice” screen already populated with the Invoice table information using the MySql Server in the .NET Controller to read the Invoice table using LINQ and LAMBDA Queries.
<table>
<tr><th>Table</th><th>Attribute</th></tr>
<tr><td rowspan="4">Invoice</td><td>Invoice Number</td></tr>
<tr><td>Date Sent</td></tr>
<tr><td>Date paid</td></tr>
<tr><td>Invoice Receipt Number</td></tr>
</table>
The system prompts the financial officer to select an invoice they want to capture a claim for.
The “Select an Invoice” screen has the following controls:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Select Invoice</td><td>Heading</td><td>None</td></tr>
<tr><td>Please Select</td><td>Label</td><td>This is used to prompt the financial officer to select an invoice.</td></tr>
<tr><td>Results Table</td><td>Table</td><td>This table will contain the saved Invoice information per row.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>This is used to help the Financial Officer navigate back.</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>[ALT]</td></tr>
<tr><td></td><td>Step 4:
The Financial Officer clicks on a record in the Results Table.</td><td></td><td>Step 5:
The system responds by loading the “Capture Claim” modal which has the following controls:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Capture Claim</td><td>Heading</td><td>None</td></tr>
<tr><td>Description</td><td>Label</td><td>This is used to inform the financial officer about what is required.</td></tr>
<tr><td>Receipt Number</td><td>Label & Textbox</td><td>These controls are used to get the date of the receipt Number.</td></tr>
<tr><td>Submit</td><td>Button</td><td>This is used to submit the provided details.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>This is used to help the Financial Officer navigate back.</td></tr>
</table>
The system prompts the financial officer to provide the claim details for the selected Invoice record.</td></tr>
<tr><td></td><td>Step 6:
The financial officer provides the Receipt Number, using the Textbox control.
And then clicks the Submit button to submit the receipt number.</td><td></td><td>Step 7:
The system captures and validates the provided details supplied by the Financial Officer. [ALT]
<table>
<tr><th>Input Name</th><th>Attribute in Table</th><th>Notes</th></tr>
<tr><td>Receipt Number</td><td>Receipt Number</td><td>Required, maximum of 128 characters</td></tr>
</table></td></tr>
<tr><td></td><td></td><td></td><td>Step 8:</td></tr>
</table>

The system prompts the financial to confirm the provided information, using the "**Claim Confirmation**" modal which has the following controls:

Control Name	Control Type	Notes
Confirm Receipt Number	**Heading**	None
Receipt Number	**Label**	This is used to display the provided Receipt number.
Confirm	**Button**	This is used to confirm the receipt number.
Cancel	**Button**	This is used to help the Financial Officer navigate back.

Step 9:

The Financial Officer clicks the "**Confirm**" button.
[ALT]

Step 10:

The system stores the following attribute(s) in the **Invoice** table:

Table	Attribute	Notes
Invoice	**Receipt Number**	This is the number used to track claims on the 3rd party payment portal (Taulia)

The system prompts the financial officer that the receipt number was captured successfully with an alert:

Control Name	Control Type	Notes
Alert	**Alert**	None
Claim Captured	**Label**	This is used to inform the financial officer that the claim was captured.

<table>
<tr><td></td><td></td><td></td><td><table><tr><td>OK</td><td>Button</td><td>This is used to acknowledge the alert.</td></tr></table></td></tr>
<tr><td></td><td>Step 11:
The financial officer clicks the “OK” button.</td><td></td><td></td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 3:
There is no invoice in the system that corresponds to the entered search parameter.

The system displays the following control to inform the financial officer that no record was found based on the provided search parameter:<table><tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>No Results Found</td><td>Label</td><td>This is used to inform the financial officer no record was found.</td></tr></table>Terminate Use Case.</td></tr>
<tr><td></td><td colspan="3">ALT Step 7:
The provided details are not in the correct format or not completed.
The system displays a Dialog of the validation error to the financial officer.<table><tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Validation Error</td><td>Label</td><td>None</td></tr><tr><td>Error Message</td><td>Label</td><td>This is used to inform the Financial Officer about the validation error.</td></tr><tr><td>OK</td><td>Button</td><td>This is used by the Financial Officer to acknowledge the error.</td></tr></table>The Financial Officer clicks the “OK” button
Return to Step 5</td></tr>
<tr><td></td><td colspan="3">ALT Step 9:
The financial officer clicks the cancel button.
Return to Step 5</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">The use case concludes once the receipt number is captured and the invoice’s status is updated.</td></tr>
</table>

POST-CONDITION:	• The Receipt Number is stored in the invoice table
BUSINESS RULES:	• Only the Financial Officer has access to capturing claims.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None
ASSUMPTIONS:	None
OPEN ISSUES:	None

1.2.5.5 Search Proof of Payment

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s): Vincent Yu</td><td colspan="2">Date: 28/06/2021</td><td>Version: 3.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Search Proof of Payment</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">5.5</td><td rowspan="3">Business Requirements: □
System Analysis: □
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Medium</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Financial Officer (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">Managing Director</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the event when the Financial Officer wants to view a Proof of Payment associated to an Invoice. The system will display the Proof of Payment and the details associated with the Proof of Payment.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The Financial Officer needs to be logged onto the system.
• Proof of Payment is already stored in the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The Financial Officer wants to search for a specific Invoice’s Proof of Payment.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1:
The Financial Officer clicks on the “Menu” to navigate to the click on the Invoicing tab and then clicking</td><td></td><td>Step 2:
The system responds by loading the “Search Proof of Payment” screen already populated with the Proof of Payment table information:<table><tr><th>Table</th><th>Attribute</th></tr><tr><td rowspan="3">Proof of Payment</td><td>Proof of Payment ID</td></tr><tr><td>Proof of Payment</td></tr><tr><td>Date</td></tr></table></td></tr>
</table>

	“**Search Proof of Payment**”		**Invoice ID** using the MySql Server in the .NET Controller to read the **Proof of Payment** table using LINQ and LAMBDA Queries. The “**Search Proof of Payment**” screen has the following controls:

Control Name	Control Type	Notes
Search Proof of Payment	**Heading**	None
Search Bar	**Textbox**	This is used to get the search parameter from the Financial Officer.
Sort	**Drop down**	This is used to get the filter option from the Financial Officer.
Results Table	**Table**	This table will contain the saved **Proof of Payment** information per row.
View Proof of Payment	**Button**	Each record in the Results Table will have a “View Proof of Payment” button. This allows the Financial Officer to view the stored Proof of payment.
Back	**Button**	This is used to help the Financial Officer

<table>
<tr><td></td><td></td><td></td><td><table><tr><td></td><td></td><td>navigate back.</td></tr></table></td></tr>
<tr><td rowspan="4"></td><td>Step 3:
The Financial Officer enters the search parameter [Invoice ID] in the Search Bar control.</td><td></td><td>Step 4:
The system captures and validates the search parameter [Invoice ID] supplied by the Financial Office, against the Invoice table. [ALT]<table><tr><th>Input Name</th><th>Attribute in Table</th><th>Notes</th></tr><tr><td>Search Parameter [Invoice ID]</td><td>Invoice_ID</td><td>Required, maximum of 13 characters.</td></tr></table></td></tr>
<tr><td></td><td></td><td>Step 5:
The system Invokes use case 5.1 Search Invoice.</td></tr>
<tr><td></td><td></td><td>Step 6:
The system retrieves the Proof of Payment record result based on the search parameter supplier by the Financial Officer from the Proof of Payment table:
Invoice ID
Proof of Payment ID
Proof of Payment
Date captured
The Proof of Payment record is displayed in the Results Table control.</td></tr>
<tr><td>Step 7:
The Financial Officer clicks the “View Proof of Payment” button.</td><td></td><td>Step 8:
The system retrieves the following attribute(s) from the Proof of Payment table:<table><tr><th>Table</th><th>Attribute</th><th>Notes</th></tr><tr><td>Proof of Payment</td><td>Proof of Payment</td><td>This is the stored screenshot image taken from the Taulia payment portal.</td></tr></table>This screenshot image is then displayed in the “View Proof of Payment” modal with the following controls:</td></tr>
</table>

Control Name	Control Type	Notes
Proof of Payment	Heading	None
Date Captured	Label	This is used to display the date details of the Proof of Payment.
Proof of Payment	Image	This is the screenshot of the POP
Back	Button	This is used to navigate back.

ALTERNATE COURSES:

ALT Step 4:

The entered search parameter is not in the correct format or not completed.

The system displays a **Dialog** of the validation error to the financial officer.

Control Name	Control Type	Notes
Validation Error	Label	None
Error Message	Label	This is used to inform the Financial Officer about the validation error.
OK	Button	This is used by the Financial Officer to acknowledge the error.

The Financial Officer clicks the “OK” button

Return to Step 2

ALT Step 4:

There is no invoice in the system that corresponds to the entered search parameter.

The system displays the following control to inform the financial officer that no record was found based on the provided search parameter:

Control Name	Control Type	Notes
No Results Found	Label	This is used to inform the financial officer no record was found.

The financial officer clears the Search Bar control.

	Return to Step 2
CONCLUSION:	The use case concludes when the Proof of Payment is displayed.
POST-CONDITION:	• The Proof of Payment table was searched for an entry that corresponds with the search parameter • The proof of payment is displayed
BUSINESS RULES:	• Only the Financial Officer has access to searching for proof of payments.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None
ASSUMPTIONS:	None
OPEN ISSUES:	None

1.2.5.6 Search Invoice Type

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><th>Author (s): Vincent Yu</th><th>Date: 28/06/2021</th><th colspan="2">Version: 3.0</th></tr>
<tr><th>USE CASE NAME:</th><th>Search Invoice Type</th><th colspan="2">USE CASE TYPE</th></tr>
<tr><td>USE CASE ID:</td><td>5.6</td><td colspan="2" rowspan="3">Business Requirements: □
System Analysis: □
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td>Low</td></tr>
<tr><td>SOURCE:</td><td>Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Financial Officer (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">Managing Director</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the event when the Financial Officer wants to search for a specific invoice type. The system will display the invoice type with its description.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The Financial Officer needs to be logged onto the system.
• Invoice type is already stored in the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The Financial Officer wants to search for a specific invoice type.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td>Step 1:
The Financial Officer clicks on the “Menu” to navigate to the click on the Invoicing tab and then clicking “Search</td><td></td><td>Step 2:
The system responds by loading the “Search Invoice Type” screen already populated with the Invoice Type table information:
<table><tr><th>Table</th><th>Attribute</th></tr><tr><td rowspan="2">Invoice Type</td><td>Invoice Type ID</td></tr><tr><td>Description</td></tr></table>using the MySql Server in the .NET Controller to read the Invoice Type table using LINQ and LAMBDA Queries.</td></tr>
</table>

<table>
<tr>
<td></td>
<td>Invoice Type”</td>
<td></td>
<td>The “Search Invoice Type” screen has the following controls:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Search Invoice Type</td><td>Heading</td><td>None</td></tr>
<tr><td>Search Bar</td><td>Textbox</td><td>This is used to get the search parameter from the Financial Officer.</td></tr>
<tr><td>Results Table</td><td>Table</td><td>This table will contain the saved Invoice Type information per row.</td></tr>
<tr><td>Back</td><td>Button</td><td>This is used to help the Financial Officer navigate back.</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 3:
The Financial Officer enters the search parameter [Invoice type description] in the Search Bar control.</td>
<td></td>
<td>Step 4:
The system captures and validates the search parameter [Invoice_type_description] supplied by the Financial Office, against the Invoice type table. [ALT]
<table>
<tr><th>Input Name</th><th>Attribute in Table</th><th>Notes</th></tr>
<tr><td>Search Parameter [Invoice_Type_Description]</td><td>Invoice_Type_Description</td><td>Required, maximum of 256 characters.</td></tr>
</table>
</td>
</tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>Step 5:
The system retrieves the Invoice Type record result based on the search parameter supplier by the Financial Officer from the Invoice Type table:
Invoice Type ID
Invoice Type Description
The Invoice type record is displayed in the Results Table control.</td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 4:
There is no invoice type in the system that corresponds to the entered search parameter.

The system displays the following control to inform the financial officer that no record was found based on the provided search parameter:<table><tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>No Results Found</td><td>Label</td><td>This is used to inform the financial officer no record was found.</td></tr></table>Terminate Use Case.</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">The use case concludes when the Invoice type record is displayed.</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">The Invoice Type table was searched for an entry that corresponds with the search parameter
The Invoice type is displayed</td></tr>
<tr><td>BUSINESS RULES:</td><td colspan="3">Only the Financial Officer has access to searching for invoice types.</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td colspan="3">None</td></tr>
<tr><td>ASSUMPTIONS:</td><td colspan="3">None</td></tr>
<tr><td>OPEN ISSUES:</td><td colspan="3">None</td></tr>
</table>

1.2.5.7 Create Invoice Type

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s): Vincent Yu</td><td colspan="2">Date: 28/06/2021</td><td>Version: 3.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Create Invoice Type</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">5.7</td><td rowspan="3">Business Requirements: □
System Analysis: □
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Low</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Financial Officer (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">Managing Director</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the event when the Financial Officer wants to create a new invoice type. The system will capture and validate the invoice type information. If the information was validated successfully the system will store it in the Invoice Type table.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The Financial Officer needs to be logged onto the system.
• The invoice type has not been created before (No duplication).</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The Financial Officer wants to create a new invoice type.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td>Step 1:
The Financial Officer clicks on the “Menu” to navigate to the click on the Invoicing tab and then</td><td></td><td>Step 2:
The system invokes use case 5.6 Search Invoice Type.</td></tr>
</table>

<table>
<tr><td></td><td>clicking “Add Invoice Type”</td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td>
Step 3:
The system responds by loading the “Create Invoice Type” screen already populated with the Invoice Type table information:
<table>
<tr><th>Table</th><th>Attribute</th></tr>
<tr><td rowspan="2">Invoice Type</td><td>Invoice Type ID</td></tr>
<tr><td>Invoice Type Description</td></tr>
</table>
using the MySql Server in the .NET Controller to read the Invoice Type table using LINQ and LAMBDA Queries.
The “Create Invoice Type” screen has the following controls:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Create Invoice Type</td><td>Heading</td><td>None</td></tr>
<tr><td>Results Table</td><td>Table</td><td>This table will contain the saved Invoice Type information per row.</td></tr>
<tr><td>Add Invoice Type</td><td>Button</td><td>This is used when the Financial Officer wishes to add a new invoice type to the system.</td></tr>
<tr><td>Back</td><td>Button</td><td>This is used to help the Financial Officer navigate back.</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 4:
The Financial Officer clicks the “Add</td><td></td><td>Step 5:
The system responds by loading the “Add Invoice Type” modal with the following controls:</td></tr>
</table>

<table>
<tr>
<td></td>
<td>Invoice Type” button.</td>
<td></td>
<td>
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Add Invoice Type</td><td>Heading</td><td>None</td></tr>
<tr><td>Invoice Type Description</td><td>Label and Textbox</td><td>This is used to display the date details of the Proof of Payment.</td></tr>
<tr><td>Submit</td><td>Button</td><td>This is used when the financial officer has provided the required details of the new Invoice Type.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>This used to cancel the creation or used to navigate back.</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 6:
The Financial officer uses the “Invoice Type Description” controls to enter the required information. Then clicks the “Submit” button</td>
<td></td>
<td>Step 7:
The system captures and validates the provided parameters [Invoice type description] supplied by the Financial Officer, against the Invoice Type table. [ALT]
<table>
<tr><th>Input Name</th><th>Attribute in Table</th><th>Notes</th></tr>
<tr><td>Search Parameter [Invoice Type Description]</td><td>Invoice_Type_Description</td><td>Required, maximum of 128 characters. Type String</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Step 8:
The system Invokes use case 5.6 Search Invoice Type.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Step 9:
The system cannot retrieve the search result based on the newly validated Invoice Type details. (As the invoice type is not present on the system and needs to be created). The system</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td></td>
<td></td>
<td>responds by presenting the “Invoice Type Confirmation” modal below controls to the financial officer on the “Create Invoice Type” screen.
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Confirm New Invoice Type</td><td>Heading</td><td>none</td></tr>
<tr><td>New Invoice Type Description</td><td>Label</td><td>This is used to display the validated information provided by the financial officer.</td></tr>
<tr><td>Confirm</td><td>Button</td><td>This is used to confirm the provided details.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>This is used to cancel or navigate back.</td></tr>
</table>
[ALT]</td>
</tr>
<tr>
<td></td>
<td>Step 10:
The financial officer clicks the “Confirm” button.</td>
<td></td>
<td>Step 11:
The system uses a SQL insert query to save the captured information in the Invoice Type table as follows:
<table>
<tr><th>Table</th><th>Attribute</th></tr>
<tr><td rowspan="2">Invoice Type</td><td>Invoice Type ID (generated)</td></tr>
<tr><td>Invoice Type Description</td></tr>
</table>
The system prompts the financial officer that the invoice type was created successfully with an alert:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Alert</td><td>Alert</td><td>None</td></tr>
<tr><td>Invoice Type Created</td><td>Label</td><td>This is used to inform the financial officer that the invoice was created.</td></tr>
</table>
</td>
</tr>
</table>

<table>
<tr><td></td><td></td><td></td><td><table><tr><td>OK</td><td>Button</td><td>This is used to acknowledge the alert.</td></tr></table></td></tr>
<tr><td></td><td>Step 12:
The financial officer clicks the “OK” button.</td><td></td><td></td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 7:
The entered parameters are not in the correct format or not completed.
The system displays a Dialog of the validation error to the financial officer.
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Validation Error</td><td>Label</td><td>None</td></tr>
<tr><td>Error Message</td><td>Label</td><td>This is used to inform the Financial Officer about the validation error.</td></tr>
<tr><td>OK</td><td>Button</td><td>This is used by the Financial Officer to acknowledge the error.</td></tr>
</table>
The Financial Officer clicks the “OK” button
Return to Step 3</td></tr>
<tr><td></td><td colspan="3">ALT Step 9:
There is an existing invoice type on the system with the same provided Invoice Type Description.
The system displays the following control to inform the financial officer that invoice type was already created and is stored on the system:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Invoice Type already created</td><td>Label</td><td>This is used to inform the financial officer that the invoice type already exists on the system.</td></tr>
<tr><td>OK</td><td>Button</td><td>This is used by the Financial Officer to acknowledge the alert.</td></tr>
</table>
Return to Step 3</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">The use case concludes when the new invoice type is created and store in the system.</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">• The Invoice type is created and stored in the Invoice Type table.</td></tr>
</table>

BUSINESS RULES:	• Only the Financial Officer has access to creating new invoice types
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None
ASSUMPTIONS:	None
OPEN ISSUES:	None

1.2.6. Human Resource Subsystem

1.2.6.1 Add employee

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Tshepang Mashao</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Add Employee</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">6.1</td><td rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Critical</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Human Resource Officer</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">The use describes the event where the officer fills out a form that requires all employee details, including name, surname, cell number, ID/Passport number, tax number, home address/postal address and email address information that has already been given by the employee. The officer then uses that information to add the employee to the system, and the use case concludes.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">The officer must be logged onto the system.
The employee form with all relevant details must be filled in</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The officer wants to add a new employee to the system</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The officer requests to add the employee.</td><td>Step 2: The officer selects the add button in the “Employee” screen.</td><td>Step 3: The system uses a SQL read query to read the EmployeeSkills table, in order to get the following attributes for the “Select Employee Skills” dropdown:
EmployeeSkillsID
EmployeeSkillsDescription
The system uses a SQL read query to read the EmployeeType table, in order to get the following attributes for the “Select Employee Type” dropdown:</td></tr>
</table>

- EmployeeTypeID
- EmployeeTypeDescription

Step 4: The system responds by loading the "Add Employee" **modal** with the following controls:

Input Name	Control Name	Notes
Add Employee	Heading	None
Name	Label	This is the label for the name of the employee
Name	Textbox	None
Surname	Name	This is the label for the surname of the employee
Surname	Textbox	None
ID/ Passport Number	Label	This is the label for the Tax Number of the employee
ID Number	Textbox	None
Tax Number	Label	This is the label for the Tax Number of the employee
Tax Number	Textbox	None
Home Address	Label	This is the label for the home address of the employee
Home Address	Text Area	None
Postal Address	Label	This is the label for the postal address of the employee
Postal Address	Text Area	None
Email Address	Label	This is the label for the email address of the employee

			Email Address | Textbox | None **Employee Skills** | Label | This is the label for the skills of the employee **Employee Skills dropdown** | Dropdown | Populated with the information retrieved from the **Employee Skills** table **Employee Type** | Label | This is the label for the type of employee. **Employee Type Status** | Dropdown | Populated with the information retrieved from the **Employee Type** table **Add** | Button | None **Cancel** | Button | This button is used when the officer wishes to terminate the process
			Step 5: The system then prompts the officer to capture the following details
	Step 6: The officer fills in all the relevant details mentioned in Step 4 from the filled-in employee form		
	Step 7: The officer selects the option to save the employee on the system by		**Step 8:** The system verifies all the required field have been filled in, and that all the fields have the correct format and contains valid details. **[ALT]** **Input Name** | **Attribute in Table** | **Validation Requirements** **Name** | Emp_Name | Required

clicking on the "Add" button. **[ALT]**

Surname	Emp_Surname	Required
Cell Number	Emp_Cell	Required
ID Number	Emp_IDNumber	Required
Tax Number	Emp_Tax	Required
Home Address	Emp_Home Address	Required
Postal Address	Emp_Postal Address	Required
Email Address	Emp_Email Address	Required
Employee Skills	EmpSkill_ Description	Required
Employee Type	EmpType_ Description	Required

Step 9: The system will prompt the officer to confirm the details entered by displaying a Confirmation **Modal**

with the following controls:

Input Name	Control Type	Notes
Confirmation Required	Label	This is the header for the modal
Are you sure the entered details are correct?	Label	This is the information used to prompt confirmation from the officer
Yes	Button	Selected when the officer accepts confirmation
No	Button	Selected when the officer denies confirmation

Step 10: The

Step 11: The system confirms that the employee has no record on the system. **[ALT]**

officer selects the “Yes” button to confirm all details. **[ALT]**

Step 12: The system uses a SQL insert query to save the captured information in the Employee table as follows:

Details to be saved	Attribute in Table	Details
Employee ID	Employee ID	Generated by the system by reading the last EmployeeID from the Employee table and incrementing it by one.
Name	Emp_Name	None
Surname	Emp_Surname	None
Cell Number	Emp_Cell	None
ID Number	Emp_IDNumber	None
Tax Number	Emp_Tax	None
Home Address	Emp_Home Address	None
Postal Address	Emp_Postal Address	None
Email Address	Emp_Email Address	None
Employee Skills	EmpSkill_ Description	None
Employee Type	EmpType_ Description	None

Step 13: The system notifies the officer that the employee was added successfully on the system, by displaying a “Add Employee Success

Notification" Modal with the following controls: **[ALT]**

Input Name	Control Type	Notes
Result	Label	This is the header for the modal
The employee has been added successfully	Label	This is the information that indicates that the employee was added successfully.
OK	Button	Selected when the officer accepts confirmation

ALTERNATE COURSES:

Step 7: The officer wishes to terminate this use case by clicking on the "Cancel" button.

The system then displays a Modal with the following controls, to request confirmation from the officer:

Input Name	Control Type	Notes
Confirmation Required	Label	This is the header for the modal
Are you sure you want to cancel this process?	Label	This is the information used to prompt confirmation from the officer
Yes	Button	Selected when the officer accepts confirmation
No	Button	Selected when the officer denies confirmation

The officer confirms the termination of this use case by clicking on the "Yes" button. The system will respond by terminating the use case.

ALT: The officer denies confirmation for the termination of this use case by clicking on the "No" button. Return to Step 5.

Step 8: The details provided by the officer is not valid.

	<table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Validation Error</td><td>Label</td><td>This is the header for the modal</td></tr><tr><td>The entered details are not in the correct format. Please try again.</td><td>Label</td><td>This is the information used to indicate that the entered details are not valid.</td></tr><tr><td>OK</td><td>Button</td><td>None</td></tr></table> The officer clicks the “OK” button. Return to step 6.
	Step 10: The officer does not confirm the details. Return to Step 6.
	Step 11: The employee already has a record on the system. Display error notification. <table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Duplicate Error</td><td>Label</td><td>This is the header for the modal</td></tr><tr><td>This employee already exists on the system</td><td>Label</td><td>This is the information used to indicate that the employee already exists on the system</td></tr></table> Terminate use case.
	Step 13: The employee was not added successfully. <table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Result</td><td>Label</td><td>This is the header for the modal</td></tr><tr><td>An unexpected error has been encountered. Please try again later.</td><td>Label</td><td>This is the information that indicates that the employee was not added successfully.</td></tr><tr><td>OK</td><td>Button</td><td>Selected when the officer accepts confirmation</td></tr></table> The officer clicks the “OK” button Terminate use case.
CONCLUSION:	The use case concludes when a notification that the employee has been successfully added is displayed
POST-CONDITION:	The officer has successfully added as an employee
BUSINESS RULES:	• Only the officer may add an employee on the system • An employee cannot be added to the system twice
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none

ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.6.2 Edit employee

Gohvan Construction			
Author (s): Tshepang Mashao	**Date: 28/06/2021**	**Version: 1.0**	
USE CASE NAME:	**Edit Employee**	**USE CASE TYPE**	
USE CASE ID:	**6.2**	**Business Requirements:** ☐	
PRIORITY:	Moderate	**System Analysis:** ☐	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	Human Resource Officer		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	None		
DESCRIPTION:	The use describes the event where the officer fills out a form that requires all employee details, including name, surname, cell number, ID/Passport number, tax number, home address/postal address and email address information that has already been given by the employee. The officer then uses that information to add the employee to the system, and the use case concludes.		
PRE-CONDITION:	• The officer is logged in		
TRIGGER:	The officer would like to update an employee on the system.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The officer would like to update an existing employee.		

<table>
<tr><td rowspan="3"></td><td>Step 2: The officer provides info to search for the employee .</td><td></td><td>Step 3: The system invokes Use Case 6.3 Search Employee. [ALT]</td></tr>
<tr><td></td><td></td><td>Step 4: The system uses a SQL read query to read the EmployeeSkills table, in order to get the following attributes for the “Select Employee Skills” dropdown:
EmployeeSkillsID
EmployeeSkillsDescription
The system uses a SQL read query to read the EmployeeType table, in order to get the following attributes for the “Select Employee Type” dropdown:
EmployeeTypeID
EmployeeTypeDescription</td></tr>
<tr><td></td><td></td><td>Step 5: The system responds by loading the “Update Employee” Modal with the following controls:
<table>
<tr><th>Input Name</th><th>Control Name</th><th>Notes</th></tr>
<tr><td>Update Employee</td><td>Heading</td><td>None</td></tr>
<tr><td>Name</td><td>Label</td><td>This is the label for the name of the employee</td></tr>
<tr><td>Name</td><td>Textbox</td><td>None</td></tr>
<tr><td>Surname</td><td>Name</td><td>This is the label for the surname of the employee</td></tr>
<tr><td>Surname</td><td>Textbox</td><td>None</td></tr>
<tr><td>ID/ Passport Number</td><td>Label</td><td>This is the label for the Tax Number of the employee</td></tr>
<tr><td>ID Number</td><td>Textbox</td><td>None</td></tr>
</table></td></tr>
</table>

			Tax Number	Label	This is the label for the Tax Number of the employee
			Tax Number	Textbox	None
			Home Address	Label	This is the label for the home address of the employee
			Home Address	Text Area	None
			Postal Address	Label	This is the label for the postal address of the employee
			Postal Address	Text Area	None
			Email Address	Label	This is the label for the email address of the employee
			Email Address	Textbox	None
			Employee Skills	Label	This is the label for the skills of the employee
			Employee Skills dropdown	Dropdown	Populated with the information retrieved from the **Employee Skills** table
			Employee Type	Label	This is the label for the type of employee.
			Employee Type Status	Dropdown	Populated with the information retrieved from the **Employee**

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th></th><th></th><th>Type table</th></tr>
<tr><td>Add</td><td>Button</td><td>None</td></tr>
<tr><td>Cancel</td><td>Button</td><td>This button is used when the officer wishes to terminate the process</td></tr>
</table>
</td></tr>
<tr><td></td><td></td><td></td><td>Step 6: The system then prompts the officer to capture the following details</td></tr>
<tr><td></td><td>Step 7: The officer fills in all the relevant details mentioned in Step 4 from the filled-in employee form</td><td></td><td></td></tr>
<tr><td></td><td>Step 8: The officer selects the option to update the employee on the system by clicking on the “Update” button. [ALT]</td><td></td><td>Step 9: The system verifies all the required field have been filled in, and that all the fields have the correct format and contains valid details. [ALT]
<table>
<tr><th>Input Name</th><th>Attribute in Table</th><th>Validation Requirements</th></tr>
<tr><td>Name</td><td></td><td>Required</td></tr>
<tr><td>Surname</td><td></td><td>Required</td></tr>
<tr><td>Cell Number</td><td></td><td>Required</td></tr>
<tr><td>ID Number</td><td></td><td>Required</td></tr>
<tr><td>Tax Number</td><td></td><td>Required</td></tr>
<tr><td>Home Address</td><td></td><td>Required</td></tr>
<tr><td>Postal Address</td><td></td><td>Required</td></tr>
<tr><td>Email Address</td><td></td><td>Required</td></tr>
<tr><td>Employee Skills</td><td></td><td>Required</td></tr>
</table>
</td></tr>
</table>

			Employee Type | | Required
			Step 10: The system will prompt the officer to confirm the details entered by displaying a Modal with the following controls: **Input Name** | **Control Type** | **Notes** **Confirmation Required** | Label | This is the header for the modal **Are you sure the entered details are correct?** | Label | This is the information used to prompt confirmation from the officer **Yes** | Button | Selected when the officer accepts confirmation **No** | Button | Selected when the officer denies confirmation
	Step 11: The officer selects the "Yes" button to confirm all details. **[ALT]**		**Step 12:** The system uses a SQL insert query to save the captured information in the Employee table as follows: **Details to be saved** | **Attribute in Table** | **Details** **Employee ID** | Employee ID | Generated by the system by reading the last EmployeeI D from the Employee table and incrementin g it by one. **Name** | Emp_Name | None **Surname** | Emp_Surname | None

			Cell Number | Emp_Cell | None **ID Number** | Emp_IDNumber | None **Tax Number** | Emp_Tax | None **Home Address** | Emp_Home Address | None **Postal Address** | Emp_Postal Address | None **Email Address** | Emp_Email Address | None **Employee Skills** | EmpSkill_ Description | None **Employee Type** | EmpType_ Description | None
			Step 13: The system notifies the officer that the employee was added successfully on the system, by displaying a Modal with the following controls: **[ALT]** **Input Name** | **Control Type** | **Notes** **Result** | Label | This is the header for the modal **The employee has been updated successfully** | Label | This is the information that indicates that the employee was updated successfully. **OK** | Button | Selected when the officer accepts confirmation
ALTERNATE COURSES:			**Step 8:** The officer wishes to terminate this use case by clicking on the "Cancel" button. The system then displays a Modal with the following controls, to request confirmation from the officer:

Input Name	Control Type	Notes
Confirmation Required	Label	This is the header for the modal
Are you sure you want to cancel this process?	Label	This is the information used to prompt confirmation from the officer
Yes	Button	Selected when the officer accepts confirmation
No	Button	Selected when the officer denies confirmation

The officer confirms the termination of this use case by clicking on the “Yes” button. The system will respond by terminating the use case.

ALT: The officer denies confirmation for the termination of this use case by clicking on the “No” button. Return to Step 5.

Step 9: The details provided by the officer is not valid.

Input Name	Control Type	Notes
Validation Error	Label	This is the header for the modal
The entered details are not in the correct format. Please try again.	Label	This is the information used to indicate that the entered details are not valid.
OK	Button	None

The officer clicks the “OK” button.

Return to step 6.

Step 11: The officer does not confirm the details. Return to Step 6.

Step 13: The employee was not updated successfully.

Input Name	Control Type	Notes
Result	Label	This is the header for the modal

	An unexpected error has been encountered. Please try again later.	Label	This is the information that indicates that the employee was not updated successfully.
	OK	Button	Selected when the officer accepts confirmation
	The officer clicks the “OK” button Terminate use case.		
CONCLUSION:	This use case concludes when the officer is notified that the updates have been made successfully.		
POST-CONDITION:	The user has successfully updated the employee’s information.		
BUSINESS RULES:	none		
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none		
ASSUMPTIONS:	none		
OPEN ISSUES:	none		

1.2.6.3 Search employee

Gohvan Construction			
Author (s): Tshepang Mashao	**Date: 28/06/2021**	**Version: 1.0**	
USE CASE NAME:	**Search Employee**	**USE CASE TYPE**	
USE CASE ID:	**6.3**	**Business Requirements:** ☐	
PRIORITY:	Moderate	**System Analysis:** ☐	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	Human Resource Officer		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	None		
DESCRIPTION:	This use case describes the event of the Human Resource Officer searching for an employee. The officer can search for an employee by entering search criteria and selecting the employee that the officer wanted to search for. This use case concludes when the system displays all the employee's details.		
PRE-CONDITION:	• The officer is logged in • The employee exists on the system.		
TRIGGER:	The officer would like to search for an employee.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The officer wants to search for an employee.		
	Step 2: The officer clicks on the		**Step 3:** The system responds by loading the "Employee" screen with the following controls:

<table>
<tr>
<td></td>
<td>“Employee ” option.</td>
<td></td>
<td>
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Search Employee</td><td>Heading</td><td>None</td></tr>
<tr><td>Insert keyword:</td><td>Label</td><td>This label is for the search parameter</td></tr>
<tr><td>Search Param-eter</td><td>Textbox</td><td>None</td></tr>
<tr><td>Search</td><td>Button</td><td>None</td></tr>
<tr><td>Search Result Table</td><td>Table</td><td>This table is not visible until the system has searched the Employee table and a match has been found</td></tr>
<tr><td>View</td><td>Button</td><td>This button is found in each row of the table to view the result.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>The button is used when the user wishes to terminate the use case</td></tr>
<tr><td>+</td><td>Button</td><td>This control is used to add an employee</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 4: The officer selects which information he wants to search the employee with.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Step 5: The officer provides the information</td>
<td></td>
<td>Step 6: The system verifies that all the required fields are filled in, contains valid details and are in the correct format. [ALT]:</td>
</tr>
</table>

<table>
<tr><td></td><td>to search by and selects the “Search” button.</td><td></td><td>
<table>
<tr><th>Input Name</th><th>Attribute in Table</th><th>Validation Requirements</th></tr>
<tr><td>Name</td><td></td><td>Required</td></tr>
<tr><td>Surname</td><td></td><td>Required</td></tr>
<tr><td>ID Number</td><td></td><td>Required, maximum of 13 characters</td></tr>
</table>
</td></tr>
<tr><td></td><td></td><td></td><td>Step 7: The SQL read query is used to read the following attributes from the following tables based on the keyword provided:
• Employee Name
• Employee Surname
• Cell Number
• ID/Passport number
• Tax number
• Home address
• Postal address
• Email address
From the Employee table
• Employee Skills
From the Employee Skills table
• Employee Type
From the Employee Type table</td></tr>
<tr><td></td><td></td><td></td><td>Step 8: The system populates the Search Result table with the data read in Step 7. [ALT]</td></tr>
<tr><td></td><td>Step 8: The officer selects the client they want to view by clicking on the “View” button</td><td></td><td>Step 9: The system displays all the relevant employee details mentioned in Step 7 by loading the “View Employee” modal with the following controls:
<table>
<tr><th>Input Name</th><th>Control Name</th><th>Notes</th></tr>
<tr><td>Search Employee</td><td>Heading</td><td>None</td></tr>
<tr><td>Name</td><td>Label</td><td>This is the label for the name of the employee</td></tr>
<tr><td>Name</td><td>Textbox</td><td>This textbox is</td></tr>
</table>
</td></tr>
</table>

					populated with the data of the attribute and is disabled.
			Surname	Name	This is the label for the surname of the employee
			Surname	Textbox	This textbox is populated with the data of the attribute and is disabled.
			ID/ Passport Number	Label	This is the label for the Tax Number of the employee
			ID Number	Textbox	This textbox is populated with the data of the attribute and is disabled.
			Tax Number	Label	This is the label for the Tax Number of the employee
			Tax Number	Textbox	This textbox is populated with the data of the attribute and is disabled.
			Home Address	Label	This is the label for

					the home address of the employee
			Home Address	Text Area	This textbox is populated with the data of the attribute and is disabled.
			Postal Address	Label	This is the label for the postal address of the employee
			Postal Address	Text Area	This textbox is populated with the data of the attribute and is disabled.
			Email Address	Label	This is the label for the email address of the employee
			Email Address	Textbox	This textbox is populated with the data of the attribute and is disabled.
			Employee Skills	Label	This is the label for the skills of the employee
			Employee Skills dropdown	Dropdown	Populated with the information retrieved

					from the **Employee Skills** table
			Employee Type	Label	This is the label for the type of employee.
			Employee Type Status	Dropdown	Populated with the information retrieved from the **Employee Type** table
			Employee Role	Label	This is the label for the role of the employee
			Employee Role	Textbox	This textbox is populated with the data of the attribute and is disabled.
			Username	Label	This is the label for the username of the employee
			Username	Textbox	This textbox is populated with the data of the attribute and is disabled.
			Update	Button	This is used to initiate the process of updating an employee

			Upload Document	Button	This is used to initiate the process of uploading a document for the employee
			X	Button	This button is used when the officer wishes to terminate the process

ALTERNATE COURSES:

Step 6: The information provided is not valid, details were not provided and/or not in the correct format.

Input Name	Control Type	Notes
Validation Error	Label	This is the header for the modal
The entered details are not in the correct format. Please try again.	Label	This is the information used to indicate that the entered details are not valid.
OK	Button	None

The officer clicks the “OK” button.

Step 7: No data is displayed because no information matched the search criteria entered. The system notifies the officer by displaying a **Modal**, with the following controls:

Input Name	Control Type	Notes
Result	Label	This is the header for the modal.
There is no employee that matches with the entered details.	Label	This is the information used to indicate that the entered details have no matches.
OK	Button	None

The officer clicks the “OK” button

Return to Step 3.

CONCLUSION:	The use case concludes when the details of the employee searched for is displayed
POST-CONDITION:	The officer has successfully searched an employee.
BUSINESS RULES:	none
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
FFASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.6.4 Upload Employee Document

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s): Tshepang Mashao</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Upload Employee Document</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">6.4</td><td rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Critical</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Human Resource Manager</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">The system must allow the user to upload employee documents to the system. The user can attach the employee’s certifications and health and safety certificates to the employee’s file.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The officer must be logged onto the system</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The officer requests to upload an employee document.</td></tr>
<tr><td rowspan="3">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The officer requests to upload an employee document.</td><td></td><td></td></tr>
<tr><td>Step 2: The officer navigates to the Home Screen and</td><td></td><td>Step 3: The system responds by displaying the Upload Employee Compliance Document Modal, with the following controls:</td></tr>
</table>

	clicks on the "Upload" button in the Employee Compliance card.		

Control Name	Control Type	Notes
Upload Employee Compliance Document	Heading	None
Expiry Date	Label	None
Expiry Date	Datepicker	This is used to indicate the date that the document expires
Employee	Label	None
Employee	Textbox	This is used for the officer to input the name of the employee
Search Employee	Button	Clicked when the officer is ready to search for the employee
Cancel	Button	This button is used when the officer wishes to terminate the process
Upload	Button	This button is used when the officer wishes to upload the document
Save	Button	This button is

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td></td><td></td><td>used when the officer wants the file to be stored on the system. Until all the details are valid, it is disabled.</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 4: The officer enters the expiry date.</td><td></td><td>Step 5: The system ensures that the expiry date is valid. [ALT]
<table>
<tr><th>Input Name</th><th>Attribute in Table</th><th>Validation Requirements</th></tr>
<tr><td>Expiry Date</td><td>Emp_ ExpiryDate</td><td>Required, must be a future date</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 6: The officer enters the details (name, surname or ID) of the employee</td><td></td><td>Step 7: The system invokes Use Case 6.3 Search Employee.</td></tr>
<tr><td></td><td>Step 8: The officer clicks the “Upload” button to upload the document. [ALT]</td><td></td><td>Step 9: The system displays an upload dialogue for the officer to choose the file to upload.</td></tr>
<tr><td></td><td>Step 10: The officer uploads the employee document by choosing the appropriate file.</td><td></td><td>Step 11: The system uploads the selected file. [ALT]</td></tr>
<tr><td></td><td>Step 12: The user clicks the “Save” button. [ALT]</td><td></td><td>Step 13: The system prompts the officer to confirm the saving of the employee document by displaying a modal with the following controls:</td></tr>
</table>

			<table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Confirmation Required</td><td>Label</td><td>This is the header for the modal</td></tr><tr><td>Are you sure the entered details and document are correct?</td><td>Label</td><td>This is the information used to prompt confirmation from the officer</td></tr><tr><td>Yes</td><td>Button</td><td>Selected when the officer accepts confirmation</td></tr><tr><td>No</td><td>Button</td><td>Selected when the officer denies confirmation</td></tr></table>
	Step 14: The officer confirms the captured information by clicking the “Yes” button. **[ALT]**		**Step 15:** The system uses an SQL insert query to save the uploaded employee document form in the **Employee Compliance Document** table and saves the Expiry Date and Employee information.**[ALT]** <table><tr><th>Details to be saved</th><th>Attribute in Table</th><th>Details</th></tr><tr><td>Employee Compliance Document</td><td>Employee ID</td><td>Generated by the system by reading the last Employee Compliance Doc_ID from the Employee Compliance Document table and incrementing it by one.</td></tr><tr><td>Title</td><td>EmpComp</td><td>None</td></tr></table>

	Doc_ Title	
Expiry Date	EmpComp Doc_ Expiry Date	None
Employee ID	Employee_ ID	None

Step 16: The system notifies the office that the document was uploaded successfully by using a Modal with the following controls: **[ALT]**

Input Name	Control Type	Notes
Result	Label	This is the header for the modal
The document has been uploaded successfully	Label	This is the information that indicates that the document was uploaded successfully.
OK	Button	Selected when the officer accepts confirmation

Step 17: The officer acknowledges the notification by clicking the “OK” button.

ALTERNATE COURSES:

Step 5: The details provided by the officer is not valid.

Input Name	Control Type	Notes

Validation Error	Label	This is the header for the modal
The entered details are not in the correct format. Please try again.	Label	This is the information used to indicate that the entered details are not valid.
OK	Button	None

The officer clicks the “OK” button.

Return to step 4.

Step 8: The officer wishes to terminate this use case by clicking on the “Cancel” button.

The system then displays a Modal with the following controls, to request confirmation from the officer:

Input Name	Control Type	Notes
Confirmation Required	Label	This is the header for the modal
Are you sure you want to cancel this process?	Label	This is the information used to prompt confirmation from the officer
Yes	Button	Selected when the officer accepts confirmation
No	Button	Selected when the officer denies confirmation

The officer confirms the termination of this use case by clicking on the “Yes” button. The system will respond by terminating the use case.

ALT: The officer denies confirmation for the termination of this use case by clicking on the “No” button. Return to Step 5.

Step 11: The officer uploads a file that is not in the correct format. The system will show an error message below the text: “This file is not in the correct format”. The “Save” button will remain disabled.

Return to Step 8.

Step 14: The officer does not confirm the upload of the Employee Compliance Document by clicking “No”. The system displays a confirmation Modal:

Input Name	Control Type	Notes

Confirmation Required	Label	This is the header for the modal
Are you sure you want to cancel this process?	Label	This is the information used to prompt confirmation from the officer
Yes	Button	Selected when the officer accepts confirmation
No	Button	Selected when the officer denies confirmation

The officer confirms the termination of this use case by clicking on the “Yes” button. The system will respond by terminating the use case.

ALT: The officer denies confirmation for the termination of this use case by clicking on the “No” button. Return to Step 12.

Step 15: A database error occurs in the upload process. Display error message, by means of a Modal, with the following controls:

Input Name	Control Type	Notes
Result	Label	This is the header for the modal
An unexpected error has been encountered. Please try again later.	Label	This is the information that indicates that the document was not uploaded successfully.
OK	Button	Selected when the officer accepts confirmation

The officer clicks the “OK” button

Terminate use case.

Step 16: The document was not uploaded successfully

Input Name	Control Type	Notes
Result	Label	This is the header for the modal
An unexpected error has been encountered. Please try again later.	Label	This is the information that indicates that the document was not uploaded successfully.
OK	Button	Selected when the officer accepts confirmation

The officer clicks the “OK” button

Terminate use case.

CONCLUSION:	The system displays a message that the document was uploaded successfully.
POST-CONDITION:	• An employee document is uploaded and stored in the Employee Compliance Document table. • The database must maintain data integrity after the employee document is uploaded.
BUSINESS RULES:	• The file must be in PDF format.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.6.5 Edit Employee Document

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Tshepang Mashao</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Edit Employee Document</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">6.5</td><td rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Critical</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Human Resource Officer</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">Certain employee certificates expire and need to be tracked and updated accordingly. The user must be able to search for specific documents/certificates and replace expired certificates with new certificates</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">The previous compliance document is expired
The officer Is logged onto the system</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The officer wishes to update the details of an employee document</td></tr>
<tr><td rowspan="2">FTYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The officer would like to update the details of an employee</td><td></td><td>Step 2: The system invokes Use Case 6.6 “Search Employee Document”</td></tr>
</table>

	Step 3: The User clicks on the "Update" button		Step 4: The system responds by loading the "Update Compliance Document" modal with the following controls:

Control Name	Control Type	Notes
Update Compliance Document	Heading	None
Expiry Date	Label	None
Expiry Date	Date-picker	This is used to indicate the date that the document expires
Employee	Label	None
Employee	Textbox	This is prepopulated with the full name of the employee
Search Employee	Button	Clicked when the officer is ready to search for the employee
Cancel	Button	This button is used when the officer wishes to terminate the process
Upload	Button	This button is used when the officer wishes to upload the document
Update	Button	This button is used when the officer wants the file to be stored on the system. Until all the details

<table>
<tr><td></td><td></td><td></td><td><table><tr><td></td><td></td><td>are valid, it is disabled.</td></tr></table></td></tr>
<tr><td></td><td>Step 4: The officer enters the expiry date.</td><td></td><td>Step 5: The system ensures that the expiry date is valid. [ALT]
<table><tr><th>Input Name</th><th>Attribute in Table</th><th>Validation Requirements</th></tr><tr><td>Expiry Date</td><td>Emp_ ExpiryDate</td><td>Required, must be a future date</td></tr></table></td></tr>
<tr><td></td><td>Step 6: The officer clicks the “Upload” button to upload the document.</td><td></td><td>Step 7: The system displays an upload dialogue for the officer to choose the file to upload.</td></tr>
<tr><td></td><td>Step 8: The officer uploads the employee document by choosing the appropriat e file.</td><td></td><td>Step 9: The system uploads the selected file. [ALT]</td></tr>
<tr><td></td><td>Step 10: The user clicks the “Update” button. [ALT]</td><td></td><td>Step 11: The system prompts the officer to confirm the saving of the employee document by displaying a modal with the following controls:
<table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Confirmation Required</td><td>Label</td><td>This is the header for the modal</td></tr><tr><td>Are you sure the entered details and document are correct?</td><td>Label</td><td>This is the information used to prompt confirmation from the officer</td></tr><tr><td>Yes</td><td>Button</td><td>Selected when the officer accepts confirmation</td></tr></table></td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>No</th><td>Button</td><td>Selected when the officer denies confirmation</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 12: The officer confirms the captured information by clicking the “Yes” button. [ALT]</td><td></td><td>Step 13: The system uses an SQL update query to save the uploaded employee document form in the Employee Compliance Document table and saves the Expiry Date and Employee information. [ALT]
<table>
<tr><th>Details to be saved</th><th>Attribute in Table</th><th>Details</th></tr>
<tr><th>Employee Compliance Document</th><td>Employee ID</td><td>Generated by the system by reading the last Employee Compliance Doc_ID from the Employee Compliance Document table and incrementing it by one.</td></tr>
<tr><th>Expiry Date</th><td></td><td>None</td></tr>
<tr><th>Employee ID</th><td></td><td>None</td></tr>
</table>
</td></tr>
<tr><td></td><td></td><td></td><td>Step 14: The system notifies the office that the document was updated successfully by using a Modal with the following controls: [ALT]
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><th>Result</th><td>Label</td><td>This is the header for the modal</td></tr>
<tr><th>The document has been</th><td>Label</td><td>This is the information that indicates that the document</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><td rowspan="4">ALTERNATE COURSES:</td><td></td><td></td><td>
<table>
<tr><td>updated successfully</td><td></td><td>was updated successfully.</td></tr>
<tr><td>OK</td><td>Button</td><td>Selected when the officer accepts confirmation</td></tr>
</table>
</td></tr>
<tr><td>Step 15: The officer clicks the “OK” button.</td><td></td><td></td></tr>
<tr><td colspan="3">Step 5: The details provided by the officer is not valid.

<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Validation Error</td><td>Label</td><td>This is the header for the modal</td></tr>
<tr><td>The entered details are not in the correct format. Please try again.</td><td>Label</td><td>This is the information used to indicate that the entered details are not valid.</td></tr>
<tr><td>OK</td><td>Button</td><td>None</td></tr>
</table>
The officer clicks the “OK” button.
Return to step 4.</td></tr>
<tr><td colspan="3">Step 9: The officer uploads a file that is not in the correct format. The system will show an error message below the text: “This file is not in the correct format”. The “Save” button will remain disabled.
Return to Step 8.</td></tr>
<tr><td colspan="3">Step 11: The officer does not confirm the upload of the Employee Compliance Document by clicking “No”. The system displays a confirmation Modal:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Confirmation Required</td><td>Label</td><td>This is the header for the modal</td></tr>
<tr><td>Are you sure you want to cancel this process?</td><td>Label</td><td>This is the information used to prompt confirmation from the officer</td></tr>
<tr><td>Yes</td><td>Button</td><td>Selected when the officer accepts confirmation</td></tr>
</table>
</td></tr>
</table>

	No | Button | Selected when the officer denies confirmation The officer confirms the termination of this use case by clicking on the "Yes" button. The system will respond by terminating the use case. **ALT:** The officer denies confirmation for the termination of this use case by clicking on the "No" button. Return to Step 12.
	Step 13: A database error occurs in the upload process. Display error message, by means of a **Modal**, with the following controls: **Input Name** | **Control Type** | **Notes** **Result** | Label | This is the header for the modal **An unexpected error has been encountered. Please try again later.** | Label | This is the information that indicates that the document was not uploaded successfully. **OK** | Button | Selected when the officer accepts confirmation The officer clicks the "OK" button Terminate use case.
	Step 14: The document was not uploaded successfully **Input Name** | **Control Type** | **Notes** **Result** | Label | This is the header for the modal **An unexpected error has been encountered. Please try again later.** | Label | This is the information that indicates that the document was not uploaded successfully. **OK** | Button | Selected when the officer accepts confirmation The officer clicks the "OK" button Terminate use case.
CONCLUSION:	
POST-CONDITION:	
BUSINESS RULES:	
IMPLEMENTATION CONSTRAINTS	none

AND SPECIFICATIONS:	
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.6.6 Search Employee document

Gohvan Construction			
Author (s): Tshepang Mashao	**Date: 28/06/2021**	**Version: 1.0**	
USE CASE NAME:	**Search Employee Document**	**USE CASE TYPE**	
USE CASE ID:	**6.6**	**Business Requirements:** ☐	
PRIORITY:	Critical	**System Analysis:** ☐	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	Human Resource Officer		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	None		
DESCRIPTION:	The Human resource manager wants to search for an employee's existing documents. The user can search for an employee document by finding the employee and searching for the document by name or document code. The documents related to search must be displayed and the system must allow the user to access the documents.		
PRE-CONDITION:			
TRIGGER:	The officer wishes to search for an employee document		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The officer requests to search for an employee document		
	Step 2: The officer clicks on the		Step 3: The system responds by displaying the "Search Compliance Document" modal, with the following controls:

<table>
<tr>
<td></td>
<td>“Search” button on the “Employee Compliance Document” card.</td>
<td></td>
<td>
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Search Compliance Document</td><td>Heading</td><td>None</td></tr>
<tr><td>Insert keyword:</td><td>Label</td><td>This label is for the search parameter</td></tr>
<tr><td>Search Parameter</td><td>Textbox</td><td>None</td></tr>
<tr><td>Search</td><td>Button</td><td>None</td></tr>
<tr><td>Search Result Table</td><td>Table</td><td>This table is not visible until the system has searched the Employee table and a match has been found</td></tr>
<tr><td>View</td><td>Button</td><td>This button is found in each row of the table to view the result.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>The button is used when the user wishes to terminate the use case</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 3: The officer provides the appropriate search parameter for the employee document and</td>
<td>**</td>
<td>Step 4: The system captures the search criteria provided and validates the information, ensures that it is filled in and in the correct format against the Employee Compliance Document table [ALT]:
<table>
<tr><th>Input Name</th><th>Attribute in Table</th><th>Validation Requirements</th></tr>
<tr><td>Title</td><td></td><td>Required</td></tr>
</table>
</td>
</tr>
</table>

<table>
<tr><td></td><td>continues by clicking the “Search” button. [ALT]</td><td></td><td>
<table>
<tr><td>Name</td><td></td><td>Required</td></tr>
<tr><td>Surname</td><td></td><td>Required</td></tr>
<tr><td>ID</td><td></td><td>Required</td></tr>
<tr><td>Expiry Date</td><td></td><td>Required</td></tr>
</table>
</td></tr>
<tr><td></td><td></td><td></td><td>Step 5: An SQL read query is used to read the following attributes from the following tables based on the parameter provided:
Employee Compliance Document table:
EmpCompDocTitle
EmployeeID
EmpCompDocExpiryDate
The Employee table is searched, based on the EmployeeID, for the:
EmployeeName
EmployeeSurname
Associated to the Employee Document</td></tr>
<tr><td></td><td></td><td></td><td>Step 6: The system populates the Search Result Table with the data received in the previous step. [ALT]</td></tr>
<tr><td></td><td>Step 7: The officer selects the employee document he requested to search for from the table by clicking the “View” button</td><td></td><td>Step 8: The system displays the selected Employee Compliance Document information in the form of a Modal: [ALT]
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>View Compliance Document</td><td>Heading</td><td>This serves as a header for the modal</td></tr>
<tr><td>Title</td><td>Label</td><td>This label shows the title of the employee compliance document</td></tr>
<tr><td>Document</td><td>Viewer</td><td>This viewer displays the employee compliance document</td></tr>
</table>
</td></tr>
</table>

			Cancel	Button	A floating action button selected when the officer wants to close the modal
			X	Button	Selected when the officer wants to close the modal
			Update	Button	A floating action button selected when the officer wants to update the employee document

ALTERNATE COURSES:

Step 4: The system failed to validate the search parameters.

Input Name	Control Type	Notes
Validation Error	Label	This is the header for the modal
The entered details are not in the correct format. Please try again.	Label	This is the information used to indicate that the entered details are not valid.
OK	Button	None

The officer clicks the “OK” button.

Return to step 3.

Step 5: The entered keyword results in no matches found. A modal will display to indicate the error, with the following controls:

Input Name	Control Type	Notes
Result	Label	This is the header for the modal.
There is no employee that	Label	This is the information used to indicate that the

	<table><tr><td>matches with the entered details.</td><td></td><td>entered details have no matches.</td></tr><tr><td>OK</td><td>Button</td><td>None</td></tr></table> The officer clicks the “OK” button Terminate use case. **Step 8:** The system failed to display the details of the selected Employee Compliance Document. A modal will display, with the following controls: <table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Result</td><td>Label</td><td>This is the header for the modal</td></tr><tr><td>An unexpected error has been encountered. Please try again later.</td><td>Label</td><td>This is the information that indicates that the employee type was not created successfully.</td></tr><tr><td>OK</td><td>Button</td><td>Selected when the officer accepts confirmation</td></tr></table> The officer clicks the “OK” button Terminate use case.
CONCLUSION:	The system retrieves and displays the employee document from the **Employee Compliance Document** table.
POST-CONDITION:	• The officer can view information of the employee document. • The searched employee document has been found.
BUSINESS RULES:	none
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.6.7 Create Account

Gohvan Construction			
Author (s): Tshepang Mashao	**Date: 28/06/2021**	**Version: 1.0**	
USE CASE NAME:	**Create Account**	**USE CASE TYPE**	
USE CASE ID:	**6.7**	**Business Requirements:** ☐	
PRIORITY:	Critical	**System Analysis:** ☐	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	Human Resource Officer		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	None		
DESCRIPTION:	The use describes the event where the officer fills out a form that requires all the user's details, including name, surname, cell number, ID/Passport number, tax number, home address/postal address and email address information that has already been given by the employee. The officer then uses that information to add the user to the system, by which a new username and password will be created, and the use case concludes.		
PRE-CONDITION:	The officer must be logged onto the system. The employee form with all relevant details must be filled in		
TRIGGER:	The officer wants to add a new employee to the system		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The officer wants to create a new account.		
	Step 2: The officer		**Step 3:** The system invokes Use Case 6.3 "Search Employee".

<table>
<tr><td></td><td>selects the add button in the Account screen.</td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td>Step 4: The system responds by loading the “Create Account” modal with the following controls:
<table>
<tr><th>Input Name</th><th>Control Name</th><th>Notes</th></tr>
<tr><td>Create Account</td><td>Heading</td><td>None</td></tr>
<tr><td>Name</td><td>Label</td><td>None</td></tr>
<tr><td>Name</td><td>Textbox</td><td>This is populated with for the name of the user</td></tr>
<tr><td>Surname</td><td>Name</td><td>None</td></tr>
<tr><td>Surname</td><td>Textbox</td><td>This is populated with the surname of the user</td></tr>
<tr><td>ID/ Passport Number</td><td>Label</td><td>None</td></tr>
<tr><td>ID/ Passport Number</td><td>Textbox</td><td>This is populated with the ID number of the user</td></tr>
<tr><td>Username</td><td>Label</td><td>This is the label for the Username of the user</td></tr>
<tr><td>Username</td><td>Textbox</td><td>None</td></tr>
<tr><td>Email</td><td>Label</td><td>This is the label for the email of the ser</td></tr>
<tr><td>Email</td><td>Textbox</td><td>None</td></tr>
<tr><td>Password</td><td>Label</td><td>This is the label for the password of the user</td></tr>
</table></td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>Password</td><td>Textbox</td><td>None</td></tr>
<tr><td>Security Question</td><td>Label</td><td>This is the label for the security question of the user</td></tr>
<tr><td>Security Question</td><td>Textbox</td><td>None</td></tr>
<tr><td>Security Answer</td><td>Label</td><td>This is the label for the security answer of the user</td></tr>
<tr><td>Security Answer</td><td>Textbox</td><td>None</td></tr>
<tr><td>User Role</td><td>Label</td><td>None</td></tr>
<tr><td>User Role</td><td>Textbox</td><td>This is populated with the Emp Type_ Description of the user</td></tr>
<tr><td>Create</td><td>Button</td><td>This button is used when the officer wishes to register the new user account</td></tr>
<tr><td>Cancel</td><td>Button</td><td>This button is used when the officer wishes to terminate the process</td></tr>
</table>
</td></tr>
<tr><td></td><td></td><td></td><td>Step 5: The system then prompts the officer to capture the following details</td></tr>
<tr><td></td><td>Step 6: The officer fills in all the relevant details mentioned in Step 4.</td><td></td><td></td></tr>
<tr><td></td><td>Step 7: The officer selects</td><td></td><td>Step 8: The system verifies all the required field have been filled in, and that</td></tr>
</table>

<table>
<tr><td></td><td>the option to save the user on the system by clicking on the “Create” button. [ALT]</td><td></td><td>all the fields have the correct format and contains valid details. [ALT]
<table>
<tr><th>Input Name</th><th>Attribute in Table</th><th>Validation Requirements</th></tr>
<tr><td>Username</td><td></td><td>Required</td></tr>
<tr><td>Password</td><td></td><td>Required</td></tr>
<tr><td>Email</td><td></td><td>Required</td></tr>
<tr><td>ID Number</td><td></td><td>Required</td></tr>
<tr><td>Security Question</td><td></td><td>Required</td></tr>
<tr><td>Security Answer</td><td></td><td>Required</td></tr>
</table></td></tr>
<tr><td></td><td></td><td></td><td>Step 9: The system will prompt the officer to confirm the details entered by displaying a Modal with the following controls:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Confirmation Required</td><td>Label</td><td>This is the header for the modal</td></tr>
<tr><td>Are you sure the entered details are correct?</td><td>Label</td><td>This is the information used to prompt confirmation from the officer</td></tr>
<tr><td>Yes</td><td>Button</td><td>Selected when the officer accepts confirmation</td></tr>
<tr><td>No</td><td>Button</td><td>Selected when the officer denies confirmation</td></tr>
</table></td></tr>
<tr><td></td><td>Step 10: The officer selects</td><td></td><td>Step 11: The system confirms that the user has no record on the system. [ALT]</td></tr>
</table>

<table>
<tr><td></td><td>the “Yes” button to confirm all details. [ALT]</td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td>Step 12: The system uses a SQL insert query to save the captured information in the User table as follows:
<table>
<tr><th>Details to be saved</th><th>Attribute in Table</th><th>Details</th></tr>
<tr><td>User ID</td><td>UserID</td><td>Generated by the system by reading the last UserID from the User table and incrementing it by one.</td></tr>
<tr><td>Username</td><td>Username</td><td>None</td></tr>
<tr><td>Password</td><td>Password</td><td>None</td></tr>
<tr><td>Email</td><td>User_Email</td><td>None</td></tr>
<tr><td>Security Question</td><td>Security Question</td><td>None</td></tr>
<tr><td>Security Answer</td><td>Security Answer</td><td>None</td></tr>
</table></td></tr>
<tr><td></td><td></td><td></td><td>Step 13: The system notifies the officer that the employee was added successfully on the system, by displaying a Modal with the following controls: [ALT]
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Result</td><td>Label</td><td>This is the header for the modal</td></tr>
<tr><td>The user has been created successfully</td><td>Label</td><td>This is the information that indicates that the user was created successfully.</td></tr>
<tr><td>OK</td><td>Button</td><td>Selected when the officer</td></tr>
</table></td></tr>
</table>

					accepts confirmation

ALTERNATE COURSES:

Step 7: The officer wishes to terminate this use case by clicking on the “Cancel” button.

The system then displays a Modal with the following controls, to request confirmation from the officer:

Input Name	Control Type	Notes
Confirmation Required	Label	This is the header for the modal
Are you sure you want to cancel this process?	Label	This is the information used to prompt confirmation from the officer
Yes	Button	Selected when the officer accepts confirmation
No	Button	Selected when the officer denies confirmation

The officer confirms the termination of this use case by clicking on the “Yes” button. The system will respond by terminating the use case.

ALT: The officer denies confirmation for the termination of this use case by clicking on the “No” button. Return to Step 5.

Step 8: The details provided by the officer is not valid.

Input Name	Control Type	Notes
Validation Error	Label	This is the header for the modal
The entered details are not in the correct format. Please try again.	Label	This is the information used to indicate that the entered details are not valid.
OK	Button	None

The officer clicks the “OK” button.

Return to step 6.

Step 10: The officer does not confirm the details. Return to Step 6.

<table>
<tr><td></td><td>
Step 11: The employee already has a record on the system. Display error notification.
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Duplicate Error</td><td>Label</td><td>This is the header for the modal</td></tr>
<tr><td>This employee already exists on the system</td><td>Label</td><td>This is the information used to indicate that the employee already exists on the system</td></tr>
<tr><td>OK</td><td>Button</td><td>None</td></tr>
</table>
The officer clicks on the “OK” button.
Terminate use case.

Step 13: The employee was not added successfully.
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Result</td><td>Label</td><td>This is the header for the modal</td></tr>
<tr><td>An unexpected error has been encountered. Please try again later.</td><td>Label</td><td>This is the information that indicates that the employee was not added successfully.</td></tr>
<tr><td>OK</td><td>Button</td><td>Selected when the officer accepts confirmation</td></tr>
</table>
The officer clicks the “OK” button
Terminate use case.
</td></tr>
<tr><td>CONCLUSION:</td><td>The use case concludes when a notification that the employee has been successfully added is displayed</td></tr>
<tr><td>POST-CONDITION:</td><td>The officer has successfully added as an employee</td></tr>
<tr><td>BUSINESS RULES:</td><td>• Only the officer may add an employee on the system
• An employee cannot be added to the system twice</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td>none</td></tr>
<tr><td>ASSUMPTIONS:</td><td>none</td></tr>
<tr><td>OPEN ISSUES:</td><td>none</td></tr>
</table>

1.2.6.8 Edit Account

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Tshepang Mashao</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Edit Account</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">6.8</td><td rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Moderate</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Human Resource Officer</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the event where the officer wants to update the details of an account on the system. The system retrieves the original information and allows the Human Resource Officer to edit the details as desired. The new details are then saved into the User table after the confirmation of the details has been accepted.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The officer is logged in</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The officer would like to update an employee on the system.</td></tr>
<tr><td rowspan="3">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The officer would like to update an existing user.</td><td></td><td></td></tr>
<tr><td>Step 2: The officer provides info to</td><td></td><td>Step 3: The system invokes Use Case 6.9 Search Account. [ALT]</td></tr>
</table>

search for the user.

Step 4: The system responds by loading the "Update Account" modal with the following controls:

Input Name	Control Name	Notes
Create Account	Heading	None
Name	Label	None
Name	Textbox	This is populated with for the name of the user
Surname	Name	None
Surname	Textbox	This is populated with the surname of the user
ID/ Passport Number	Label	None
ID/ Passport Number	Textbox	This is populated with the ID number of the user
Username	Label	This is the label for the Tax Number of the user
Username	Textbox	None
Email	Label	This is the label for the email of the ser
Email	Textbox	None
Password	Label	This is the label for the password of the user
Password	Textbox	None
Security Question	Label	This is the label for the security question of the user
Security Question	Textbox	None

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>Security Answer</td><td>Label</td><td>This is the label for the security answer of the user</td></tr>
<tr><td>Security Answer</td><td>Textbox</td><td>None</td></tr>
<tr><td>User Role</td><td>Label</td><td>None</td></tr>
<tr><td>User Role</td><td>Textbox</td><td>This is populated with the Employee Type_ Description of the employee</td></tr>
<tr><td>Update</td><td>Button</td><td>This button is used when the officer wishes to update the user details</td></tr>
<tr><td>Cancel</td><td>Button</td><td>This button is used when the officer wishes to terminate the process</td></tr>
</table>
</td></tr>
<tr><td></td><td></td><td></td><td>Step 5: The system then prompts the officer to capture the following details</td></tr>
<tr><td></td><td>Step 6: The officer fills in all the relevant details that needs to be updated</td><td></td><td></td></tr>
<tr><td></td><td>Step 7: The officer selects the option to update the user by clicking on the “Update” button. [ALT]</td><td></td><td>Step 8: The system verifies all the required field have been filled in, and that all the fields have the correct format and contains valid details. [ALT]
<table>
<tr><th>Input Name</th><th>Attribute in Table</th><th>Validation Requirements</th></tr>
<tr><td>Username</td><td>Username</td><td>Required</td></tr>
<tr><td>Password</td><td>Password</td><td>Required</td></tr>
<tr><td>Email</td><td>User_Email</td><td>Required</td></tr>
<tr><td>ID Number</td><td>Security Question</td><td>Required</td></tr>
</table>
</td></tr>
</table>

			Security Question | Security Answer | Required **Security Answer** | Username | Required
			Step 9: The system will prompt the officer to confirm the details entered by displaying a Modal with the following controls: **Input Name** | **Control Type** | **Notes** **Confirmation Required** | Label | This is the header for the modal **Are you sure the entered details are correct?** | Label | This is the information used to prompt confirmation from the officer **Yes** | Button | Selected when the officer accepts confirmation **No** | Button | Selected when the officer denies confirmation
	Step 10: The officer selects the "Yes" button to confirm all details. **[ALT]**		**Step 12:** The system uses a SQL insert query to save the captured information in the User table as follows: **Details to be saved** | **Attribute in Table** | **Details** **User ID** | UserID | Generated by the system by reading the last UserID from the User table and incrementing it by one. **Username** | Username | None **Password** | Password | None **Email** | User_Email | None **Security Question** | Security Question | None

			Security Answer | Security Answer | None
			Step 13: The system notifies the officer that the user was added successfully on the system, by displaying a Modal with the following controls: **[ALT]**

Input Name	Control Type	Notes
Result	Label	This is the header for the modal
The user has been updated successfully	Label	This is the information that indicates that the employee was updated successfully.
OK	Button	Selected when the officer accepts confirmation

ALTERNATE COURSES:

Step 7: The officer wishes to terminate this use case by clicking on the "Cancel" button.

The system then displays a Modal with the following controls, to request confirmation from the officer:

Input Name	Control Type	Notes
Confirmation Required	Label	This is the header for the modal
Are you sure you want to cancel this process?	Label	This is the information used to prompt confirmation from the officer
Yes	Button	Selected when the officer accepts confirmation
No	Button	Selected when the officer denies confirmation

The officer confirms the termination of this use case by clicking on the "Yes" button. The system will respond by terminating the use case.

<table>
<tr><td></td><td>ALT: The officer denies confirmation for the termination of this use case by clicking on the “No” button. Return to Step 5.</td></tr>
<tr><td></td><td>Step 8: The details provided by the officer is not valid.

<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Validation Error</td><td>Label</td><td>This is the header for the modal</td></tr>
<tr><td>The entered details are not in the correct format. Please try again.</td><td>Label</td><td>This is the information used to indicate that the entered details are not valid.</td></tr>
<tr><td>OK</td><td>Button</td><td>None</td></tr>
</table>
The officer clicks the “OK” button.
Return to step 6.</td></tr>
<tr><td></td><td>Step 10: The officer does not confirm the details. Return to Step 6.</td></tr>
<tr><td></td><td>Step 13: The user was not updated successfully.
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Result</td><td>Label</td><td>This is the header for the modal</td></tr>
<tr><td>An unexpected error has been encountered. Please try again later.</td><td>Label</td><td>This is the information that indicates that the user was not updated successfully.</td></tr>
<tr><td>OK</td><td>Button</td><td>Selected when the officer accepts confirmation</td></tr>
</table>
The officer clicks the “OK” button
Terminate use case.</td></tr>
<tr><td>CONCLUSION:</td><td>This use case concludes when the officer is notified that the updates have been made successfully.</td></tr>
<tr><td>POST-CONDITION:</td><td>The officer has successfully updated the user’s information.</td></tr>
<tr><td>BUSINESS RULES:</td><td>none</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td>none</td></tr>
<tr><td>ASSUMPTIONS:</td><td>none</td></tr>
<tr><td>OPEN ISSUES:</td><td>none</td></tr>
</table>

1.2.6.9 Search Account

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Tshepang Mashao</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Search Account</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">6.9</td><td rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Moderate</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Human Resource Officer</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the event of the Human Resource Officer searching for an employee. The Human Resource Officer can search for an employee by entering search criteria and selecting the employee that the Human Resource Officer wanted to search for. This use case concludes when the system displays all the employee’s details.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The Human Resource Officer is logged in
• The employee exists on the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The Human Resource Officer would like to search for an employee.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The Human Resource Officer wants to search for an employee.</td><td></td><td></td></tr>
</table>

<table>
<tr>
<td></td>
<td>Step 2: The Human Resource Officer clicks on the “Employee ” option.</td>
<td></td>
<td>Step 3: The system responds by loading the “Account” screen with the following controls:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Account</td><td>Heading</td><td>None</td></tr>
<tr><td>Insert keyword:</td><td>Label</td><td>This label is for the search parameter</td></tr>
<tr><td>Search Param-eter</td><td>Textbox</td><td>None</td></tr>
<tr><td>Search</td><td>Button</td><td>None</td></tr>
<tr><td>Search Result Table</td><td>Table</td><td>This table is not visible until the system has searched the Employee table and a match has been found</td></tr>
<tr><td>View</td><td>Button</td><td>This button is found in each row of the table to view the result.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>The button is used when the user wishes to terminate the use case</td></tr>
<tr><td>+</td><td>Button</td><td>This control is used to add an employee</td></tr>
</table></td>
</tr>
<tr>
<td></td>
<td>Step 4: The Human Resource Officer provides the information to search by and selects the “Search” button.</td>
<td></td>
<td>Step 6: The system verifies that all the required fields are filled in, contains valid details and are in the correct format. [ALT]:
<table>
<tr><th>Input Name</th><th>Attribute in Table</th><th>Validation Requirements</th></tr>
<tr><td>Username</td><td></td><td>Required</td></tr>
<tr><td>Email</td><td></td><td>Required</td></tr>
</table></td>
</tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>Step 7: The SQL read query is used to read the following attributes from the following tables based on the keyword provided:
Username
Email
From the User table</td></tr>
<tr><td></td><td></td><td></td><td>Step 8: The system populates the Search Result table with the data read in Step 7. [ALT]</td></tr>
<tr><td></td><td>Step 8: The Human Resource Officer selects the client they want to view by clicking on the “View” button.</td><td></td><td>Step 9: The system displays all the relevant user details mentioned in Step 7 by loading the “View User” modal with the following controls:
<table>
<tr><th>Input Name</th><th>Control Name</th><th>Notes</th></tr>
<tr><td>View User</td><td>Heading</td><td>None</td></tr>
<tr><td>Name</td><td>Label</td><td>None</td></tr>
<tr><td>Name</td><td>Textbox</td><td>This is populated with for the name of the user</td></tr>
<tr><td>Surname</td><td>Name</td><td>None</td></tr>
<tr><td>Surname</td><td>Textbox</td><td>This is populated with the surname of the user</td></tr>
<tr><td>ID/ Passport Number</td><td>Label</td><td>None</td></tr>
<tr><td>ID/ Passport Number</td><td>Textbox</td><td>This is populated with the ID number of the user</td></tr>
<tr><td>Username</td><td>Label</td><td>This is the label for the Tax Number of the user</td></tr>
<tr><td>Username</td><td>Textbox</td><td>None</td></tr>
<tr><td>Email</td><td>Label</td><td>This is the label for the email of the ser</td></tr>
</table></td></tr>
</table>

			Email	Textbox	None
			Password	Label	This is the label for the password of the user
			Password	Textbox	None
			Security Question	Label	This is the label for the security question of the user
			Security Question	Textbox	None
			Security Answer	Label	This is the label for the security answer of the user
			Security Answer	Textbox	None
			User Role	Label	None
			User Role	Textbox	This is populated with the Employee Type_ Description of the employee
			X	Button	This button is used when the Human Resource Officer wishes to terminate the process

ALTERNATE COURSES:

Step 6: The information provided is not valid, details were not provided and/or not in the correct format.

Input Name	Control Type	Notes
Validation Error	Label	This is the header for the modal
The entered details are not in the correct	Label	This is the information used to indicate that the entered details are not valid.

	format. Please try again. | | | **OK** | Button | None The Human Resource Officer clicks the “OK” button.
	Step 7: No data is displayed because no information matched the search criteria entered. The system notifies the Human Resource Officer by displaying a **Modal**, with the following controls: **Input Name** | **Control Type** | **Notes** **Result** | Label | This is the header for the modal. **There is no employee that matches with the entered details.** | Label | This is the information used to indicate that the entered details have no matches. **OK** | Button | None The Human Resource Officer clicks the “OK” button Return to Step 3.
CONCLUSION:	The use case concludes when the details of the employee searched for is displayed
POST-CONDITION:	The Human Resource Officer has successfully searched an employee.
BUSINESS RULES:	none
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.6.10 Create Employee Type

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Tshepang Mashao</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Create employee type</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">6.10</td><td rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Moderate</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Human Resource Officer</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the event wherein the Human Resource Officer adds an Employee Type onto the system. This use case starts when the Human Resource Officer wants to add an employee type, then navigates to the button where after clicked, the Add Employee Type modal will appear. The Human Resource Officer will then enter the appropriate details. The system will save the details once validation has been done successfully. The use case will conclude when the Employee Type is successfully saved on the system.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The Human Resource Officer must be logged into the system</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">.The Human Resource Officer wishes to add a new employee type to the system</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The Human Resource</td><td></td><td></td></tr>
</table>

<table>
<tr><td rowspan="2"></td><td>Officer wants to add an employee type</td><td></td><td></td></tr>
<tr><td>Step 2: The Human Resource Officer clicks the Employee card.</td><td></td><td>Step 3: The system responds by loading the Employee screen, with the following controls:

<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Search Employee</td><td>Heading</td><td>None</td></tr>
<tr><td>Insert keyword:</td><td>Label</td><td>This label is for the search parameter</td></tr>
<tr><td>Search Param-eter</td><td>Textbox</td><td>None</td></tr>
<tr><td>Search</td><td>Button</td><td>None</td></tr>
<tr><td>Search Result Table</td><td>Table</td><td>This table is not visible until the system has searched the Employee table and a match has been found</td></tr>
<tr><td>View</td><td>Button</td><td>This button is found in each row of the table to view the result.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>The button is used when the user wishes to terminate the use case</td></tr>
<tr><td>+</td><td>Button</td><td>This control is used to add an employee, a skill or a type</td></tr>
</table>
</td></tr>
</table>

<table>
<tr>
<td></td>
<td>Step 4: The Human Resource Officer clicks the “+” button and selects the option to create a new employee type</td>
<td></td>
<td>Step 5: The system loads the Add Employee Type Modal with the following controls:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Employee Type Name</td><td>Label</td><td>This label indicates that the employee type name must be entered</td></tr>
<tr><td>Employee Type Name</td><td>Textbox</td><td>None</td></tr>
<tr><td>Employee Type Description</td><td>Label</td><td>This label indicates that the employee type description must be entered</td></tr>
<tr><td>Employee Type Description</td><td>Textbox</td><td>None</td></tr>
<tr><td>Cancel</td><td>Button</td><td>When clicked, the Employee screen will show.</td></tr>
<tr><td>Create</td><td>Button</td><td>When clicked, the confirmation modal will appear.</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 6: The Human Resource Officer enters the required details and clicks on the “Create” button.
[ALT]</td>
<td></td>
<td>Step 7: The system captures and validates the information entered:
• EmployeeTypeName
• EmployeeTypeDescription
In the Employee Type table
[ALT]</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td></td>
<td></td>
<td>Step 8: The system loads the Employee Type Confirmation Modal with the following controls:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Confirmation Required</td><td>Label</td><td>This is the header for the modal</td></tr>
<tr><td>Are you sure the entered details are correct?</td><td>Label</td><td>This is the information used to prompt confirmation from the Human Resource Officer</td></tr>
<tr><td>Yes</td><td>Button</td><td>Selected when the Human Resource Officer accepts confirmation</td></tr>
<tr><td>No</td><td>Button</td><td>Selected when the Human Resource Officer denies confirmation</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 9: The Human Resource Officer accepts confirmation of the employee type by clicking the “Yes” button. [ALT]</td>
<td></td>
<td>Step 10: The system saves the creation of the Employee Type record using and SQL insert query and saves the following data:
• EmployeeTypeID
• EmployeeTypeName
• EmployeeTypeDescription
In the EmployeeType table.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Step 11: The system displays a modal indicating the success of creating an Employee Type, with the following controls: [ALT]</td>
</tr>
</table>

			Input Name	**Control Type**	**Notes**
			Result	Label	This is the header for the modal
			The employee type has been added successfully	Label	This is the information that indicates that the employee type was created successfully.
			OK	Button	Selected when the Human Resource Officer accepts confirmation
ALTERNATE COURSES:	Step 6: The Human Resource Officer clicks the cancel button. Terminate use case.				
	Step 8: Input validation has failed.				
	Input Name	**Control Type**	**Notes**		
	Validation Error	Label	This is the header for the modal		
	The entered details are not in the correct format. Please try again.	Label	This is the information used to indicate that the entered details are not valid.		
	OK	Button	None		
	The Human Resource Officer clicks the “OK” button. Return to step 6.				
	Step 8: The Human Resource Officer does not confirm the creation of the Employee Type by clicking “No”. The system displays a confirmation Modal:				
	Input Name	**Control Type**	**Notes**		
	Confirmation Required	Label	This is the header for the modal		

Are you sure you want to cancel this process?	Label	This is the information used to prompt confirmation from the Human Resource Officer
Yes	Button	Selected when the Human Resource Officer accepts confirmation
No	Button	Selected when the Human Resource Officer denies confirmation

The Human Resource Officer confirms the termination of this use case by clicking on the “Yes” button. The system will respond by terminating the use case.

ALT: The Human Resource Officer denies confirmation for the termination of this use case by clicking on the “No” button. Return to Step 6.

Step 11A: The Employee type already exists on the system. The system displays an error message in the form of a Modal, with the following controls:

Input Name	Control Type	Notes
Duplicate Error	Label	This is the header for the modal
This employee type already exists on the system	Label	This is the information used to indicate that the employee type already exists on the system
OK	Button	None

The Human Resource Officer clicks “OK” to acknowledge the message.

Terminate use case.

Step 11B: The system failed to successfully create an Employee Type. An error message is displayed, with the following controls:

Input Name	Control Type	Notes
Result	Label	This is the header for the modal
An unexpected error has been encountered. Please try again later.	Label	This is the information that indicates that the employee type was not created successfully.

	OK	Button	Selected when the Human Resource Officer accepts confirmation
	The Human Resource Officer clicks the “OK” button Terminate use case.		
CONCLUSION:	This use case concludes when the Human Resource Officer is informed of the successful creation of the Employee Type		
POST-CONDITION:	The Human Resource Officer has successfully added an Employee Type onto the system and saved it in the Employee Type table		
BUSINESS RULES:	• The employee type description must be concise. • Only the management team will have an account		
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none		
ASSUMPTIONS:	none		
OPEN ISSUES:	none		

1.2.6.11 Search Employee Type

Gohvan Construction			
Author (s): Tshepang Mashao	**Date: 28/06/2021**	**Version: 1.0**	
USE CASE NAME:	**Search Employee Type**	**USE CASE TYPE**	
USE CASE ID:	**6.11**	**Business Requirements:** ☐	
PRIORITY:	Moderate	**System Analysis:** ☐	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	Human Resource Officer		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	None		
DESCRIPTION:	This use case describes the event wherein the Human Resource Officer wants to search for an Employee Type on the system. This use case starts when the Human Resource Officer wants to find a certain Employee Type. This use case will conclude the Human Resource Officer has a successful search result.		
PRE-CONDITION:	• The Human Resource Officer is logged onto the system • The Employee Type must exist on the system		
TRIGGER:	.The Human Resource Officer wants to search for an Employee Type		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The Human Resource Officer wants to search for an		

<table>
<tr><td></td><td>Employee Type</td><td></td><td></td></tr>
<tr><td></td><td>Step 2: The Human Resource Officer clicks the Employee card.</td><td></td><td>Step 3: The system responds by loading the Employee screen, with the following controls:

<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Search Employee</td><td>Heading</td><td>None</td></tr>
<tr><td>Insert keyword:</td><td>Label</td><td>This label is for the search parameter</td></tr>
<tr><td>Search Parameter</td><td>Textbox</td><td>None</td></tr>
<tr><td>Search</td><td>Button</td><td>None</td></tr>
<tr><td>Search Result Table</td><td>Table</td><td>This table is not visible until the system has searched the Employee table and a match has been found</td></tr>
<tr><td>View</td><td>Button</td><td>This button is found in each row of the table to view the result.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>The button is used when the user wishes to terminate the use case</td></tr>
<tr><td>+</td><td>Button</td><td>This control is used to add an employee, a skill or a type</td></tr>
</table></td></tr>
<tr><td></td><td>Step 4: The Human Resource Officer</td><td></td><td>Step 5: The system validates the keyword phrase entered in the textbox. [ALT]</td></tr>
</table>

<table>
<tr><td></td><td>enters a keyword search phrase in the Search Parameter textbox, clicks the Search Button, then clicks the Employee Type Search button. [ALT]</td><td></td><td></td></tr>
<tr><td rowspan="3"></td><td></td><td></td><td>Step 6: The system uses the keyword phrase to retrieve the Employee Type data using a SQL Read query from the Search Result table and populates the table with the retrieved data.
<table>
<tr><th>Title</th><th>Attribute in Table</th></tr>
<tr><td>Name</td><td>EmployeeType Name</td></tr>
<tr><td>Description</td><td>EmployeeType Description</td></tr>
</table></td></tr>
<tr><td></td><td></td><td>Step 7: The system then populates the table using the retrieved information and displays it to the Employee screen. [ALT]</td></tr>
<tr><td>Step 8: The Human Resource Officer can then select an Employee type record from the table to view the details of the desired Employee Type by clicking the “View” button.</td><td></td><td>Step 9: The system displays the selected Employee Type information in the form of a Modal: [ALT]
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>View Employee Type</td><td>Heading</td><td>This serves as a header for the modal</td></tr>
<tr><td>Employee Type Name</td><td>Label</td><td>This label indicates that the name of the employee type will be shown</td></tr>
</table></td></tr>
</table>

			Employee Type Name	Textbox	This textbox is populated with the data of the attribute and is disabled.
			Description	Label	This label indicates that the description of the employee type will be shown
			Employee Type Description	Textbox	This textbox is populated with the data of the attribute and is disabled.
			X	Button	Selected when the Human Resource Officer wants to close the modal

ALTERNATE COURSES:

Step 4: The Human Resource Officer no longer wants to search for an employee type. Terminate use case. Return to Employee Screen.

Step 5: The system failed to validate the search parameters.

Input Name	Control Type	Notes
Validation Error	Label	This is the header for the modal
The entered details are not in the correct format. Please try again.	Label	This is the information used to indicate that the entered details are not valid.
OK	Button	None

The Human Resource Officer clicks the “OK” button.

Return to step 4.

<table>
<tr><td></td><td>
Step 7: The entered keyword results in no matches found. A modal will display to indicate the error, with the following controls:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Result</td><td>Label</td><td>This is the header for the modal.</td></tr>
<tr><td>There is no employee that matches with the entered details.</td><td>Label</td><td>This is the information used to indicate that the entered details have no matches.</td></tr>
<tr><td>OK</td><td>Button</td><td>None</td></tr>
</table>
The Human Resource Officer clicks the “OK” button
Terminate use case.
</td></tr>
<tr><td></td><td>
Step 9: The system failed to display the details of the selected Employee Type. A modal will display, with the following controls:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Result</td><td>Label</td><td>This is the header for the modal</td></tr>
<tr><td>An unexpected error has been encountered. Please try again later.</td><td>Label</td><td>This is the information that indicates that the employee type was not created successfully.</td></tr>
<tr><td>OK</td><td>Button</td><td>Selected when the Human Resource Officer accepts confirmation</td></tr>
</table>
The Human Resource Officer clicks the “OK” button
Terminate use case.
</td></tr>
<tr><td>CONCLUSION:</td><td>This use case concludes when the system displays the search results.</td></tr>
<tr><td>POST-CONDITION:</td><td>• The Human Resource Officer has successfully searched for an Employee Type and retrieved the details from the EmployeeType table</td></tr>
<tr><td>BUSINESS RULES:</td><td>• Only the Human Resource Officer is allowed to search for an employee type</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td>none</td></tr>
<tr><td>ASSUMPTIONS:</td><td>none</td></tr>
<tr><td>OPEN ISSUES:</td><td>none</td></tr>
</table>

1.2.6.12 Create Employee Skill

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Tshepang Mashao</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Create Employee Skill</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">6.12</td><td rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Moderate</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Human Resource Officer</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the event wherein the Human Resource Officer adds an Employee Skill onto the system. This use case starts when the Human Resource Officer wants to add an employee skill, then navigates to the button where after clicked, the Add Employee Skill modal will appear. The Human Resource Officer will then enter the appropriate details. The system will save the details once validation has been done successfully. The use case will conclude when the Employee Skill is successfully saved on the system.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The Human Resource Officer must be logged into the system</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">.The Human Resource Officer wishes to add a new employee type to the system</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The Human Resource</td><td></td><td></td></tr>
</table>

<table>
<tr><td rowspan="2"></td><td>Officer wants to add an employee skill</td><td></td><td></td></tr>
<tr><td>Step 2: The Human Resource Officer clicks the Employee card.</td><td></td><td>Step 3: The system responds by loading the Employee screen, with the following controls:

<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Search Employee</td><td>Heading</td><td>None</td></tr>
<tr><td>Insert keyword:</td><td>Label</td><td>This label is for the search parameter</td></tr>
<tr><td>Search Param-eter</td><td>Textbox</td><td>None</td></tr>
<tr><td>Search</td><td>Button</td><td>None</td></tr>
<tr><td>Search Result Table</td><td>Table</td><td>This table is not visible until the system has searched the Employee table and a match has been found</td></tr>
<tr><td>View</td><td>Button</td><td>This button is found in each row of the table to view the result.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>The button is used when the user wishes to terminate the use case</td></tr>
<tr><td>+</td><td>Button</td><td>This control is used to add an employee, a skill or a type</td></tr>
</table>
</td></tr>
</table>

<table>
<tr>
<td></td>
<td>Step 4: The Human Resource Officer clicks the “+” button and selects the option to create a new employee skill.</td>
<td></td>
<td>Step 5: The system loads the Add Employee Skill Modal with the following controls:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Employee Skill Name</td><td>Label</td><td>This label indicates that the employee type name must be entered</td></tr>
<tr><td>Employee Skill Name</td><td>Textbox</td><td>None</td></tr>
<tr><td>Employee Skill Description</td><td>Label</td><td>This label indicates that the employee type description must be entered</td></tr>
<tr><td>Employee Skill Description</td><td>Textbox</td><td>None</td></tr>
<tr><td>Cancel</td><td>Button</td><td>When clicked, the Employee screen will show.</td></tr>
<tr><td>Create</td><td>Button</td><td>When clicked, the confirmation modal will appear.</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 6: The Human Resource Officer enters the required details and clicks on the “Create” button.
[ALT]</td>
<td></td>
<td>Step 7: The system captures and validates the information entered:
• EmployeeSkillName
• EmployeeSkillDescription
In the Employee Skill table
[ALT]</td>
</tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>Step 8: The system loads the Employee Skill Confirmation Modal with the following controls:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Confirmation Required</td><td>Label</td><td>This is the header for the modal</td></tr>
<tr><td>Are you sure the entered details are correct?</td><td>Label</td><td>This is the information used to prompt confirmation from the Human Resource Officer</td></tr>
<tr><td>Yes</td><td>Button</td><td>Selected when the Human Resource Officer accepts confirmation</td></tr>
<tr><td>No</td><td>Button</td><td>Selected when the Human Resource Officer denies confirmation</td></tr>
</table></td></tr>
<tr><td></td><td>Step 9: The Human Resource Officer accepts confirmation of the employee skill by clicking the “Yes” button. [ALT]</td><td></td><td>Step 10: The system saves the creation of the Employee Skill record using and SQL insert query and saves the following data:
• EmployeeSkillID
• EmployeeSkillName
• EmployeeSkillDescription
In the EmployeeSkill table.</td></tr>
<tr><td></td><td></td><td></td><td>Step 11: The system displays a modal indicating the success of creating an Employee Skill, with the following controls: [ALT]</td></tr>
</table>

			Input Name / **Control Type** / **Notes** **Result** / Label / This is the header for the modal **The employee type has been added successfully** / Label / This is the information that indicates that the employee skill was created successfully . **OK** / Button / Selected when the Human Resource Officer accepts confirmation
ALTERNATE COURSES:	**Step 6:** The Human Resource Officer clicks the cancel button. Terminate use case.		
	Step 8: Input validation has failed. **Input Name** / **Control Type** / **Notes** **Validation Error** / Label / This is the header for the modal **The entered details are not in the correct format. Please try again.** / Label / This is the information used to indicate that the entered details are not valid. **OK** / Button / None The Human Resource Officer clicks the "OK" button. Return to step 6.		
	Step 9: The Human Resource Officer does not confirm the creation of the Employee Skill by clicking "No". The system displays a confirmation Modal: **Input Name** / **Control Type** / **Notes** **Confirmation Required** / Label / This is the header for the modal		

	Are you sure you want to cancel this process?	Label	This is the information used to prompt confirmation from the Human Resource Officer
	Yes	Button	Selected when the Human Resource Officer accepts confirmation
	No	Button	Selected when the Human Resource Officer denies confirmation

The Human Resource Officer confirms the termination of this use case by clicking on the “Yes” button. The system will respond by terminating the use case.

ALT: The Human Resource Officer denies confirmation for the termination of this use case by clicking on the “No” button. Return to Step 6.

Step 11A: The Employee Skill already exists on the system. The system displays an error message in the form of a Modal, with the following controls:

Input Name	Control Type	Notes
Duplicate Error	Label	This is the header for the modal
This employee type already exists on the system	Label	This is the information used to indicate that the employee type already exists on the system
OK	Button	None

The Human Resource Officer clicks “OK” to acknowledge the message.

Terminate use case.

Step 11B: The system failed to successfully create an Employee Skill. An error message is displayed, with the following controls:

Input Name	Control Type	Notes
Result	Label	This is the header for the modal
An unexpected error has been encountered. Please try again later.	Label	This is the information that indicates that the employee skill was not created successfully.

	OK	Button	Selected when the Human Resource Officer accepts confirmation
	The Human Resource Officer clicks the “OK” button Terminate use case.		
CONCLUSION:	This use case concludes when the Human Resource Officer is informed of the successful creation of the Employee Skill		
POST-CONDITION:	The Human Resource Officer has successfully added an Employee Type onto the system and saved it in the EmployeeSkill table		
BUSINESS RULES:	• The employee type description must be concise. • Only the management team will have an account		
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none		
ASSUMPTIONS:	none		
OPEN ISSUES:	none		

1.2.6.13 Search Employee Skill

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Tshepang Mashao</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Search Employee Skill</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">6.13</td><td rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Moderate</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Human Resource Officer</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the event wherein the Human Resource Officer wants to search for an Employee Skill on the system. This use case starts when the Human Resource Officer wants to find a certain Employee Skill. This use case will conclude when the Human Resource Officer has a successful search result.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The Human Resource Officer is logged onto the system
• The Employee Skill must exist on the system</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The Human Resource Officer wants to search for an Employee Skill</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td>Step 1: The Human Resource Officer wants to search for an</td><td></td><td></td></tr>
</table>

<table>
<tr><td></td><td>Employee Skill.</td><td></td><td></td></tr>
<tr><td></td><td>Step 2: The Human Resource Officer clicks the Employee card.</td><td></td><td>Step 3: The system responds by loading the Employee screen, with the following controls:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Search Employee</td><td>Heading</td><td>None</td></tr>
<tr><td>Insert keyword:</td><td>Label</td><td>This label is for the search parameter</td></tr>
<tr><td>Search Parameter</td><td>Textbox</td><td>None</td></tr>
<tr><td>Search</td><td>Button</td><td>None</td></tr>
<tr><td>Search Result Table</td><td>Table</td><td>This table is not visible until the system has searched the Employee table and a match has been found</td></tr>
<tr><td>View</td><td>Button</td><td>This button is found in each row of the table to view the result.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>The button is used when the user wishes to terminate the use case</td></tr>
<tr><td>+</td><td>Button</td><td>This control is used to add an employee, a skill or a type</td></tr>
</table></td></tr>
<tr><td></td><td>Step 4: The Human Resource Officer</td><td></td><td>Step 5: The system validates the keyword phrase entered in the textbox. [ALT]</td></tr>
</table>

<table>
<tr><td></td><td>enters a keyword search phrase in the Search Parameter textbox, clicks the Search Button, then clicks the Employee Skill Search button. [ALT]</td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td>Step 6: The system uses the keyword phrase to retrieve the Employee Skill data using a SQL Read query from the Search Result table and populates the table with the retrieved data.
<table>
<tr><th>Title</th><th>Attribute in Table</th></tr>
<tr><td>Name</td><td>EmployeeSkill Name</td></tr>
<tr><td>Description</td><td>EmployeeSkill Description</td></tr>
</table></td></tr>
<tr><td></td><td></td><td></td><td>Step 7: The system then populates the table using the retrieved information and displays it to the Employee screen. [ALT]</td></tr>
<tr><td></td><td>Step 8: The Human Resource Officer can then select an Employee type record from the table to view the details of the desired Employee Skill by clicking the “View” button.</td><td></td><td>Step 9: The system displays the selected Employee Skill information in the form of a Modal: [ALT]
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>View Employee Skill</td><td>Heading</td><td>This serves as a header for the modal</td></tr>
<tr><td>Employee Skill Name</td><td>Label</td><td>This label indicates that the name of the employee skill will be shown</td></tr>
</table></td></tr>
</table>

			Employee Skill Name	Textbox	This textbox is populated with the data of the attribute and is disabled.
			Description	Label	This label indicates that the description of the employee skill will be shown
			Employee Skill Description	Textbox	This textbox is populated with the data of the attribute and is disabled.
			X	Button	Selected when the Human Resource Officer wants to close the modal

ALTERNATE COURSES:

Step 4: The Human Resource Officer no longer wants to search for an employee skill. Terminate use case. Return to Employee Screen.

Step 5: The system failed to validate the search parameters.

Input Name	Control Type	Notes
Validation Error	Label	This is the header for the modal
The entered details are not in the correct format. Please try again.	Label	This is the information used to indicate that the entered details are not valid.
OK	Button	None

The Human Resource Officer clicks the “OK” button.

Return to step 4.

<table>
<tr><td></td><td>
Step 7: The entered keyword results in no matches found. A modal will display to indicate the error, with the following controls:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Result</td><td>Label</td><td>This is the header for the modal.</td></tr>
<tr><td>There is no employee that matches with the entered details.</td><td>Label</td><td>This is the information used to indicate that the entered details have no matches.</td></tr>
<tr><td>OK</td><td>Button</td><td>None</td></tr>
</table>
The Human Resource Officer clicks the “OK” button
Terminate use case.
</td></tr>
<tr><td></td><td>
Step 9: The system failed to display the details of the selected Employee Skill. A modal will display, with the following controls:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Result</td><td>Label</td><td>This is the header for the modal</td></tr>
<tr><td>An unexpected error has been encountered. Please try again later.</td><td>Label</td><td>This is the information that indicates that the employee type was not created successfully.</td></tr>
<tr><td>OK</td><td>Button</td><td>Selected when the Human Resource Officer accepts confirmation</td></tr>
</table>
The Human Resource Officer clicks the “OK” button
Terminate use case.
</td></tr>
<tr><td>CONCLUSION:</td><td>This use case concludes when the system displays the search results.</td></tr>
<tr><td>POST-CONDITION:</td><td>• The Human Resource Officer has successfully searched for an Employee Skill and retrieved the details from the EmployeeSkill table</td></tr>
<tr><td>BUSINESS RULES:</td><td>• Only the Human Resource Officer is allowed to search for an employee skill</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td>none</td></tr>
<tr><td>ASSUMPTIONS:</td><td>none</td></tr>
<tr><td>OPEN ISSUES:</td><td>none</td></tr>
</table>

1.2.7. Team Subsystem

1.2.7.1 Create Team

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Amore Rossouw</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Create Team</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">7.1</td><td>Abstract: ☐</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td><td>Extension: ☑</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td><td></td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Operations Manager (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">Human Resource Officer (PSA)</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">● None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">● None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case narrative describes the process the human resource officer will follow when they want to create a new construction team. The system will capture and validate the team information if the information was validated successfully the system will store it in the TEAM table. A construction team consists of employees that are stored in the EMPLOYEE table.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">● The Human resource officer must be logged into the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The operations manager requests for a new team to be created by the human resource officer.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td></td><td></td><td>Step 1:
The system invokes use case 6.3 “Search Employee” in order to get the following attributes for the “Select Site manager” dropdown and the “Select member” dropdown:</td></tr>
</table>

- Employee_ID
- Employee_Name

The table attributes are read via an HttpGet method in the Typescript component and an API request.

Step 2:

The system responds by loading the “Add Teams” Screen with Team table information using the SQL Read query with the following controls:

Input Name	Control Type	Notes
Create Team	Header	none
Team name	Label	none
Team name	Input	none
Select site manager	Label	none
Select Site manager	Drop-down list	The select options are populated with the Employee Name attribute of each employee in the Employee table. The value of each item in the drop down is set to the corresponding Employee_ID.
Select member	Label	none
Select member	Drop-down list	The select options are populated with the Employee Name attribute of each employee in the Employee table. The value of each item in the drop down is set to the corresponding Employee_ID.

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td></td><td></td><td></td></tr>
<tr><td>Add (+)</td><td>button</td><td>This button adds the selected employee to the team.</td></tr>
<tr><td>Team Members Table</td><td>Table</td><td>Populated with the “Employee_Name” attribute of each employee read from the employee table that was added with the “Add (+)” button.</td></tr>
<tr><td>Edit Team</td><td>Button</td><td>None</td></tr>
<tr><td>Cancel</td><td>Button</td><td>This button is used when the human resource officer wishes to terminate this process.</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 3: The operations manager gives the human resource officer the details of the new Team.</td><td>Step 4: The human resource officer fills in the necessary information into the appropriate input fields requested by the system.</td><td></td></tr>
<tr><td></td><td></td><td>Step 5: The human resource officer selects the “Create Team” button. [ALT]</td><td>Step 6: The system captures and validates the information supplied by the human resource officer against the Team table using Angular Validation: [ALT]
<table>
<tr><th>Input Name</th><th>Attribute in table</th><th>Validation requirements</th></tr>
<tr><td>Team Name</td><td>TeamName</td><td>Required</td></tr>
</table>
</td></tr>
</table>

Select Site manager	Site_Manager Employee_ID	Required

The system captures and validates the information supplied by the human resource officer against the **Employee Team** table using Angular Validation:

Input Name	Attribute in Table	Notes
Date	Date_Assigned	Required. Must be of type date
Employee Status	Employee_ Team_Status	Required
Team members	Employee_ID	Required

Step 7:
The system displays an Angular Material Confirmation Modal, with the following controls, to prompt the human resource officer for confirmation of the entered details:

Input Name	Control Type	Notes
Confirm information	Header	none
Please confirm that the details entered are correct?	Label	none
Yes	button	This button confirms that the entered details were correct
No	button	This button confirms that the entered

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td></td><td></td><td>details were not correct</td></tr>
</table>
</td></tr>
<tr><td></td><td></td><td>Step 8:
The human resource officer confirms the captured information by clicking the “Yes” button.
[ALT]</td><td>
Step 9:
The system uses a SQL insert query to save the captured information in the TEAM table as follows:
<table>
<tr><th>Details</th><th>Attribute in Table</th><th>Notes</th></tr>
<tr><td>Team ID</td><td>Team_ID</td><td>ID is generated by the system by using the last ID in the table and incrementing it by one.</td></tr>
<tr><td>Team Name</td><td>Team_Name</td><td>none</td></tr>
<tr><td>Site manager</td><td>Site_ Manager_ Employee_ID</td><td>The ID will be the same as the site manager’ s Employee_ID</td></tr>
<tr><td>Team Status</td><td>Team_Status</td><td>The Team status is automatically set to active when a new team is created.</td></tr>
</table>
The system uses a SQL insert query to save the captured information in the EMPLOYEE TEAM table as follows:
<table>
<tr><th>Details</th><th>Attribute in Table</th><th>Notes</th></tr>
<tr><td>Team ID</td><td>Team_ID</td><td>ID is from the Team table</td></tr>
<tr><td>Employee ID</td><td>Employee_ID</td><td>ID is from the</td></tr>
</table>
</td></tr>
</table>

			<table><tr><td></td><td></td><td>Employee table</td></tr><tr><td>Date</td><td>Date_Assigned</td><td>The current date will be stored.</td></tr><tr><td>Employee Status</td><td>Employee_ Team_Status</td><td>The Employee Team status is automatically set to active when a new team is created.</td></tr></table> **[ALT]**
			Step 10: The system notifies human resource officer that the Team was added successfully, using an Angular Material Success Notification Modal with the following controls: <table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Success</td><td>Header</td><td>none</td></tr><tr><td>The team was successfully created</td><td>Label</td><td>none</td></tr><tr><td>OK</td><td>button</td><td>None</td></tr></table>
		Step 11: The human resource officer clicks the “Ok” button.	
ALTERNATE COURSES:	**ALT Step 5:** The human resource officer selects the “Cancel” button. The use case is terminated.		
	ALT Step 6: • The captured details are not valid. The system displays a notification indicating that the information received is incorrect or		

<table>
<tr><td></td><td>incomplete with an Angular Material Invalid Notification Modal as follows:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Information Invalid</td><td>Header</td><td>none</td></tr>
<tr><td>The entered details were incorrect. Please fill in the correct information.</td><td>Label</td><td>none</td></tr>
<tr><td>OK</td><td>button</td><td>None</td></tr>
</table>
The system returns to step 5.</td></tr>
<tr><td></td><td>ALT Step 8:
The Human Resource Officer does not confirm that the information captured is correct by clicking the “No” button.
The system returns to step 1.</td></tr>
<tr><td></td><td>ALT Step 9:
The system does not successfully add the Team information to the TEAM and TEAM EMPLOYEE table. The system notifies the user that the Team has not been added successfully with an Angular Material Error Modal as follows:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Error</td><td>Header</td><td>none</td></tr>
<tr><td>An error has occurred while saving the Team details. Please try again.</td><td>Label</td><td>none</td></tr>
<tr><td>OK</td><td>button</td><td>None</td></tr>
</table>
The system terminates this use case.</td></tr>
<tr><td>CONCLUSION:</td><td>This use case concludes when the human resource officer has created a new Team successfully.</td></tr>
<tr><td>POST-CONDITION:</td><td>● The team has been added to the TEAM table and EMPLOYEE TEAM table successfully.</td></tr>
</table>

1.2.7.2 Search Team

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Amore
Rossouw</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Search Team</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">7.2</td><td rowspan="3">Business Requirements:☐
System Analysis: o
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Medium</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">The User (PBA) [the management director, director, operations manager, human resource officer and the health and safety officer]</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">• None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case narrative describes the process the user will follow when they want to search for a team or when the system invokes this use case.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The User must be logged into the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The user wants to search for a team.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1:
The user wants to search for an existing team. The user clicks on the “Team” option in the side menu bar</td><td></td><td>Step 2:
The system responds by loading the “Teams “screen with the following controls:
<table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Teams</td><td>Header</td><td>none</td></tr></table></td></tr>
</table>

<table>
<tr>
<td></td>
<td>on the home screen.</td>
<td></td>
<td>
<table>
<tr><td>+</td><td>Button</td><td>This control is used to add a Team</td></tr>
<tr><td>Search Team</td><td>Label</td><td>none</td></tr>
<tr><td>Search Parameter</td><td>Input</td><td>none</td></tr>
<tr><td>Search</td><td>Button</td><td>none</td></tr>
<tr><td>Team Table</td><td>Table</td><td>This table is populated with the “team name” and “site manager” attributes from the TEAM table. An “edit” button will be next to each Team record.</td></tr>
<tr><td>Edit Team</td><td>Button</td><td>This control is used to edit a Team</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 3:
The user clicks on the “Search Team” input and enters the requested search parameter. Thereafter the user clicks on the “Search Team” button.</td>
<td></td>
<td>Step 4:
The system captures and validates the information, supplied by the User, against the TEAM table using Angular Validation:

<table>
<tr><th>Input Name</th><th>Attribute in table</th><th>Validation requirements</th></tr>
<tr><td>Team Name</td><td>Team_Name</td><td>Required</td></tr>
</table>

[ALT]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Step 5:
A SQL read query is used to read the following attributes from the following tables based on the parameter provided:

The Table is searched based on the Team Name:

TEAM Table:
• Team_ID
• Team_Name
• Site_Manager_Employee_ID</td>
</tr>
</table>

			• Team_Status The Employee Team table is saved based on the Team ID associated with the Team Name: **EMPLOYEE TEAM** Table: • Team_ID • Employee_ID • Date_Assigned • Status The Employee Table is searched based on the Employee ID's associated with the Team ID: **EMPLOYEE** Table: • Employee_ID • Employee_Name • Employee_Surname The table attributes are read via an HttpGet method in the Typescript component and an API request.
			Step 6: The system populates the Team Table with the data received in the previous step **[ALT]**

Step 7:
The User selects the team from the search result by clicking on the "Team Name" label.
[ALT]

Step 8:
The system responds by loading the "View Team" page with the following controls:

Input Name	Control Type	Notes
Team Name	Header	This header's text is dynamically changed to the selected Team's team name.
Site Manager	Label	none
Site Manager Name	Label	This label's text is dynamically changed to the selected Team's Site Manager name.
Team Member Table	Table	This table is populated with the "Employee_Name" "Date_Assigned" and "Status" attributes from the Employee and Employee Team table.
Back	Button	This button will return the user to step 3.

ALTERNATE COURSES:

ALT Step 4:
The captured search parameters are not valid. The system displays an Angular Material Invalid Notification modal indicating that the information received is incorrect or incomplete:

Input Name	Control Type	Notes
Information Invalid	Header	none
The entered parameters were incorrect. Please try again.	Label	none
OK	button	None

The system returns to step 3.

<table>
<tr><td></td><td>ALT Step 6:
The team that the user searched does not exist within the database. The system notifies the user that no search results were found by displaying an Angular Material Error modal with the following controls:

<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>No Team found</td><td>Header</td><td>none</td></tr>
<tr><td>There was no Team found that matched your search request.</td><td>Label</td><td>none</td></tr>
<tr><td>OK</td><td>button</td><td>None</td></tr>
</table>

The system will return to Step 3.</td></tr>
<tr><td></td><td>ALT Step 7 (A):
The operations manager requests for a new team to be created. The human resource officer clicks on the “Add (+)” button. The system invokes use case “7.1 Create Team”</td></tr>
<tr><td></td><td>ALT Step 7 (B):
The human resource officer clicks on the “Edit” button for the Team they want to edit. The system invokes use case”7.3 Edit Team”.</td></tr>
<tr><td>CONCLUSION:</td><td>This use case concludes when the system retrieves and displays the team search results.</td></tr>
<tr><td>POST-CONDITION:</td><td>● The searched team is retrieved and displayed.</td></tr>
<tr><td>BUSINESS RULES:</td><td>● Only the users mentioned above are allowed to search the teams.</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td>none</td></tr>
<tr><td>ASSUMPTIONS:</td><td>none</td></tr>
<tr><td>OPEN ISSUES:</td><td>none</td></tr>
</table>

1.2.7.3 Edit Team

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Amore Rossouw</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Edit Team</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">7.3</td><td>Abstract: ☐</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Medium</td><td>Extension: ☑</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td><td></td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Operations Manager (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">Human Resource Officer (PSA)</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">• None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case narrative describes the process the human resource officer will follow when they want to update an existing team.
The system will capture and validate the team information if the information was validated successfully the system will update the team in the TEAM table.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The user must be logged into the system.
• The team that the user wishes to update must exist in the database.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">. The human resource officer wants to update an existing Team.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td></td><td colspan="2">Step 1:
The system responds by loading the “Edit Team” screen with the following controls:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Edit Team</td><td>Header</td><td>none</td></tr>
<tr><td>Team name</td><td>Label</td><td>none</td></tr>
<tr><td>Team name</td><td>Input</td><td>This control is</td></tr>
</table></td></tr>
</table>

					Populated with the current Team_Name attribute that was obtained in the use case ”7.2 Search team”.
			Select site manager	Label	none
			Select Site manager	Drop-down list	This control’ default value is set to the current Employee_Name attribute of the Site Manager that was obtained in the use case ”7.2 Search team”. The default value should be set to the current Employee_ID The select options are populated with the Employee Name attribute of each employee in the Employee table. The value of each item in the drop down is set to the corresponding Employee_ID.
			Select member	Label	none
			Select member	Drop-down list	The select options are populated with the Employee Name attribute of each employee in the Employee table. The value of each item in the drop down is set to the corresponding Employee_ID.

			Add (+)	button	This button adds the selected employee to the team.
			Team Members Table	Table	The table populated with the “Employee_Name” attribute of each employee from the employee table and the employees that were added with the “Add (+)” button. The value of each item in the table is set to the corresponding Employee_ID.
			Remove team member	button	This button will remove the team member from the ‘Team Members Table’ and change their status to inactive.
			Team Status	Label	none
			Team Status	Radio button	The radio button has two options namely: “Active” and “Inactive”
			Create Team	Button	None
			Cancel	Button	This button is used when the human resource officer wishes to terminate this process.
	Step 2: The operations manager gives the human resource officer the	**Step 3**: The human resource officer fills in the necessary information	**Step 4:** The system captures and validates the information supplied by the human resource officer against the **Team** table using Angular Validation using Angular Validation:		

<table>
<tr>
<td></td>
<td>details of the new Team</td>
<td>into the appropriate input fields requested by the system.
The human resource officer selects the “Edit Team” button.
[ALT]</td>
<td>
<table>
<tr><th>Input Name</th><th>Attribute in table</th><th>Validation requirements</th></tr>
<tr><td>Team Name</td><td>TeamName</td><td>Required</td></tr>
<tr><td>Select Site manager</td><td>Site_Manager Employee_ID</td><td>Required</td></tr>
</table>

The system captures and validates the information supplied by the human resource officer against the Employee Team table using Angular Validation using Angular Validation:

<table>
<tr><th>Input Name</th><th>Attribute in Table</th><th>Notes</th></tr>
<tr><td>Date</td><td>Date_Assigned</td><td>Required.
Must be of type date</td></tr>
<tr><td>Employee Status</td><td>Employee_ Team_Status</td><td>Required</td></tr>
<tr><td>Team members</td><td>Employee_ID</td><td>Required</td></tr>
</table>
[ALT]
</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Step 5:
The system displays an Angular Material Confirmation Modal, with the following controls, to prompt the human resource officer for confirmation of the entered details:

<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Confirm information</td><td>Header</td><td>none</td></tr>
<tr><td>Please confirm that the details entered are correct?</td><td>Label</td><td>none</td></tr>
<tr><td>Yes</td><td>button</td><td>This button confirms that the entered details were correct</td></tr>
</table>
</td>
</tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>No</td><td>button</td><td>This button confirms that the entered details were not correct</td></tr>
</table>
</td></tr>
<tr><td></td><td></td><td>Step 6:
The human resource officer confirms the captured information by clicking the “Yes” button.
[ALT]</td><td>
Step 7:
The system uses a SQL insert query to save the captured information in the TEAM table as follows:
<table>
<tr><th>Details</th><th>Attribute in Table</th><th>Notes</th></tr>
<tr><td>Team ID</td><td>Team_ID</td><td>The Team ID does not change.</td></tr>
<tr><td>Team Name</td><td>Team_Name</td><td>none</td></tr>
<tr><td>Site manager</td><td>Site_ Manager_ Employee_ID</td><td>The ID will be the same as the site manager’ s Employee_ID</td></tr>
<tr><td>Team Status</td><td>Team_Status</td><td>none</td></tr>
</table>
The system uses a SQL insert query to save the captured information in the EMPLOYEE TEAM table as follows:
<table>
<tr><th>Details</th><th>Attribute in Table</th><th>Notes</th></tr>
<tr><td>Team ID</td><td>Team_ID</td><td>ID is from the Team table</td></tr>
<tr><td>Employee ID</td><td>Employee_ID</td><td>ID is from the Employee table</td></tr>
<tr><td>Date</td><td>Date_Assigned</td><td>If new employees</td></tr>
</table>
</td></tr>
</table>

			are added the current date will be saved, if team members have already been assigned to the team their assigned dates will remain the same.
			Employee Status	Employee_ Team_Status	The statuses of Team members who have been removed from the team will be set to inactive.

[ALT]

Step 8:

The system notifies human resource officer that the Team was
Updated successfully, using an Angular Material Success Notification Modal with
the following controls:

Input Name	Control Type	Notes
Success	Header	none
The team details were updated successfully	Label	none
OK	button	None

Step 9:

		The human resource officer clicks the "Ok" button.	

ALTERNATE COURSES:

ALT Step 3:

The human resource officer selects the "Cancel" button. The use case is terminated.

ALT Step 4:

- The captured details are not valid. The system displays a notification indicating that the information received is incorrect or incomplete with a Angular Material Invalid Notification Modal as follows:

Input Name	Control Type	Notes
Information Invalid	Header	none
The entered details were incorrect. Please fill in the correct information.	Label	none
OK	button	None

The system returns to step 5.

ALT Step 6:

The Human Resource Officer does not confirm that the information captured is correct by clicking the "No" button.

The system returns to step 1.

ALT Step 7:

The system does not successfully update the Team information to the TEAM and TEAM EMPLOYEE table. The system notifies the user that the Team has not been updated successfully with an Angular Material Error Modal as follows:

Input Name	Control Type	Notes
Error	Header	none
An error has occurred while saving the Team details. Please try again.	Label	none
OK	button	None

The system terminates this use case.

CONCLUSION:	This use concludes when the human resource officer has successfully updated the Team.
POST-CONDITION:	• The team has been updated in the **TEAM** table.

1.2.8. Supplier Order Subsystem

1.2.8.1 Search Project Line-Item List

Gohvan Construction			
Author (s): Vincent Yu	**Date: 28/06/2021**	**Version: 3.0**	
USE CASE NAME:	**Search Project Line-Item List**	**USE CASE TYPE**	
USE CASE ID:	**8.1**	**Business Requirements:** □	
PRIORITY:	High	**System Analysis:** □	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	User (PBA) [Operations Manager and Financial Officer]		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	Managing Director		
DESCRIPTION:	This use case describes the event when a user wants to search for a Project Line-Item List on the system. This use case can also be used/called by other use case.		
PRE-CONDITION:	• The User needs to be logged on to the system.		
TRIGGER:	The User would like to search for a project's line-item list.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The User clicks on the "**Menu**" to navigate to the click on the **Supplier Order** tab and		**Step 2**: The system invokes use case 4.9 Search Project

<table>
<tr><td rowspan="2"></td><td>then clicking “Search Project Line-Item List”</td><td></td><td></td></tr>
<tr><td></td><td></td><td>
Step 3:
The system responds by loading the “Search Project Line-Item List” screen already populated with the Project table information using the MySql Server in the .NET Controller to read the Project table using LINQ and LAMBDA Queries.
<table>
<tr><th>Table</th><th>Attribute</th></tr>
<tr><td rowspan="5">Project</td><td>Project ID</td></tr>
<tr><td>Project Name</td></tr>
<tr><td>Site Name</td></tr>
<tr><td>BS Number</td></tr>
<tr><td>Date Started</td></tr>
</table>

The “Search Project Line-Item List” screen has the following controls:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Search Project Line-Item List</td><td>Heading</td><td>None</td></tr>
<tr><td>Search Bar</td><td>Textbox</td><td>This is used to get the search parameter from the Financial Officer.</td></tr>
<tr><td>Sort</td><td>Drop down</td><td>This is used to get the filter option from the User.</td></tr>
<tr><td>Results Table</td><td>Table</td><td>This table will contain the saved Project information per row.</td></tr>
<tr><td>View Project Line-Item List</td><td>Button</td><td>Each record in the Results Table will have a “View Project Line-Item List” button. This allows the User to view the stored</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td></td><td></td><td>Project Line-Item List.</td></tr>
<tr><td>Back</td><td>Button</td><td>This is used to help the User navigate back.</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 4:
The User enters the search parameter [Project ID or Project Name] in the Search Bar control.</td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td>Step 5:
The system captures and validates the search parameter [Project ID or Project Name] supplied by the user is in the appropriate format and is complete. [ALT]
<table>
<tr><th>Input Name</th><th>Attribute in Table</th><th>Notes</th></tr>
<tr><td>Search Parameter [Project ID]</td><td>Project_ID</td><td>Required, maximum of 13 characters.</td></tr>
<tr><td>Search Parameter [Project Name]</td><td>Project_Name</td><td>Required, maximum of 128 characters</td></tr>
</table>
</td></tr>
<tr><td></td><td></td><td></td><td>Step 6:
The system Invokes use case 4.9 Search Project.</td></tr>
<tr><td></td><td></td><td></td><td>Step 7:
The system retrieves the Project record result based on the search parameter supplier by the User from the Project table:
<table>
<tr><th>Table</th><th>Attribute</th></tr>
<tr><td>Project</td><td>Project ID</td></tr>
<tr><td></td><td>Project Name</td></tr>
<tr><td></td><td>Site Name</td></tr>
<tr><td></td><td>BS Number</td></tr>
<tr><td></td><td>Date Started</td></tr>
</table>
The Project record is displayed in the Results Table control. [ALT]</td></tr>
</table>

<table>
<tr><td></td><td>Step 8:
The User clicks the “View Project Line-Item List” button.</td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td>Step 9:
The system responds by navigating to the Project’s “View Project Line-Item List” modal.
The system retrieves the following attribute(s) from the Project Line-Item List, Line-Item and Price table:
<table>
<tr><th>Table</th><th>Attribute</th></tr>
<tr><td rowspan="3">Project Line-Item</td><td>Project ID</td></tr>
<tr><td>Project Line-Item ID</td></tr>
<tr><td>Quantity</td></tr>
<tr><td rowspan="2">Line-Item</td><td>Name</td></tr>
<tr><td>Description</td></tr>
<tr><td>Price</td><td>Price Amount</td></tr>
</table>

The retrieved attributes are displayed in the following controls:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Project Line-Item List</td><td>Heading</td><td>None</td></tr>
<tr><td>Results Table</td><td>Table</td><td>This table will contain the saved Project line-item list, line-item and price information per row.</td></tr>
<tr><td>OK</td><td>Button</td><td>This is used to navigate back.</td></tr>
</table></td></tr>
</table>

<table>
<tr><td>ALTERNATE COURSES:</td><td>
ALT Step 5:
The entered search parameter is not in the correct format or not completed.

The system displays a Dialog of the validation error to the financial officer.
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Validation Error</td><td>Label</td><td>None</td></tr>
<tr><td>Error Message</td><td>Label</td><td>This is used to inform the Financial Officer about the validation error.</td></tr>
<tr><td>OK</td><td>Button</td><td>This is used by the Financial Officer to acknowledge the error.</td></tr>
</table>
The User clicks the “OK” button
Return to Step 2
</td></tr>
<tr><td></td><td>
ALT Step 7:
There is no project in the system that corresponds to the entered search parameter.

The system displays the following control to inform the user that no record was found based on the provided search parameter:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>No Results Found</td><td>Label</td><td>This is used to inform the financial officer no record was found.</td></tr>
</table>
The user clears the Search Bar control.
Return to Step 2
</td></tr>
<tr><td>CONCLUSION:</td><td>This use case concludes when the system retrieves and displays the project line-item list search results.</td></tr>
<tr><td>POST-CONDITION:</td><td>• The Project Line-Item table was searched for an entry that corresponds with the search parameter.
• The searched project line-item list is retrieved and displayed.</td></tr>
<tr><td>BUSINESS RULES:</td><td>• The project line-item list needs to be present on the system in order to be searched for.</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td>None</td></tr>
<tr><td>ASSUMPTIONS:</td><td>None</td></tr>
<tr><td>OPEN ISSUES:</td><td>None</td></tr>
</table>

1.2.8.2 Edit Project Line-Item List

Gohvan Construction			
Author (s): Vincent Yu	**Date: 28/06/2021**	**Version: 3.0**	
USE CASE NAME:	**Edit Project Line-Item List**	**USE CASE TYPE**	
USE CASE ID:	**8.2**	**Business Requirements:** □	
PRIORITY:	High	**System Analysis:** □	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	Operations Manager (PBA)		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	• Managing Director • Financial Officer		
DESCRIPTION:	This use case describes the event when the Operations Manager wants to make necessary changes to the project's line-item list. Such as, adding items, removing items, and inputting the quantities.		
PRE-CONDITION:	• Operations Manager needs to be logged on to the system. • An TSS Report needs to be uploaded. • The Operations Manager is already on a specific project.		
TRIGGER:	The Operations Manager wants to edit the line-item list for a project.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action
	Step 1: The User clicks on the "**Supplier Order**" card to navigate to the "**Supplier Order**" screen and then clicking "**Edit Project Line-Item List**"		**Step 2**: The system invokes use case 8.1 Search Project Line-Item List.
			Step 3:

The system responds by loading the "**Edit Project Line-Item List**" screen already populated with the **Project Line-Item, Line-Item and Price** table information using the MySql Server in the .NET Controller to read the **Project Line-Item, Line-Item and Price** table using LINQ and LAMBDA Queries.

Table	Attribute
Project Line-Item	Project Line-Item ID
	Project ID
	Quantity
Line-Item	Name
	Description
Price	Price Amount

The "**Edit Project Line-Item List**" screen has the following controls:

Control Name	Control Type	Notes
Edit Project Line-Item List	Heading	None
Description	Label	This is used to inform the financial officer of the required info.
Results Table	Table	This table will aggregate and contain the saved **Project Line-item, Line-Item and Price** information per row.
Quantity	(Column) Textbox, Number up and down	Each record in the Results Table will have a "Quantity" input. This allows the Operations manager to edit the quantity for that specific record in the results table which is the stored Project Line-Item List.

<table>
<tr>
<td></td>
<td></td>
<td></td>
<td>
<table>
<tr><td>Remove</td><td>Button</td><td>Each record in the Results Table will have a “Remove” button. This allows the Operations manager to remove the Line-Item from the Project Line-Item List.</td></tr>
<tr><td>Add</td><td>Button</td><td>This is used when the Operations Manager wants to add Line-Item(s) to the Project Line-Item List.</td></tr>
<tr><td>Submit</td><td>Button</td><td>This is used when the Operations Manager wants to submit the edits made.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>This is used to help the User navigate back.</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 4:
The Operations Manager enters the desired quantities using the textbox control on each record of the results table. And then clicks the Submit button. [ALT]</td>
<td></td>
<td>Step 5:
The system captures and validates the Quantities supplied by the Operations Manager is in the appropriate format and is complete. [ALT]
<table>
<tr><th>Input Name</th><th>Attribute in Table</th><th>Notes</th></tr>
<tr><td>Quantity</td><td>Quantity</td><td>Required, maximum of 13 characters. Of type number</td></tr>
</table>
</td>
</tr>
</table>

Step 6:

The system prompts the Operations Manager to confirm the updates made to the Project Line-Item List. The system displays the "**Confirm Project Line-Item List"** modal with the provided updates from the Operations manager using the following controls:

Control Name	Control Type	Notes
Confirm Project Line-Item List	**Heading**	None
Description	**Label**	This is used to inform the financial officer of the required info.
Results Table	**Table**	This table will aggregate and contain the saved **Project Line-item, Line-Item and Price** information per row. With the provided changes from the Operations Manager.
Cancel	**Button**	This is used when the Operations Manager wants confirm the list is incorrect. Or wants to navigate back.
Confirm	**Button**	This is used when the Operations Manager wants to confirm that the list is correct.

Step 7:

The Operations Manager clicks the "Confirm" button. **[ALT]**

Step 8:

The system stores the updated Project Lie-Item List information in the **Project Line-Item** table:

Table	Attribute
Project Line-Item	**Project Line-Item ID**
	Project ID
	Quantity

<table>
<tr>
<td></td>
<td></td>
<td></td>
<td>Step 9:
The system displays a “Project Line-Item List Updated” alert informing the Operations Manager that the updates has been confirmed and successfully saved.
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Alert</td><td>Alert</td><td>None</td></tr>
<tr><td>Updates Confirmed</td><td>Label</td><td>This is used to inform the Operations Manager that the updates were confirmed and saved.</td></tr>
<tr><td>OK</td><td>Button</td><td>This is used to acknowledge the alert.</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 10:
The Operations Manager clicks the “OK” button.</td>
<td></td>
<td></td>
</tr>
</table>

ALTERNATE COURSES:

ALT Step 4a:

The Operations Manager clicks the “Add” button.

ALT Step 5a:

The system invokes use case 10.2 Search Line-Item.

ALT Step 6a:

The system responds by loading the “**Add project line-items**” screen already populated with the **Line-Item** table information using the MySql Server in the .NET Controller to read the **Line-Item** table using LINQ and LAMBDA Queries.

Table	Attribute
Line-Item	Line-item ID
	Name
	Description

The system prompts the Operations Manager to select the line-items to be added to the project’s line-item list. Using the following controls on the “**Add project line-items**” screen

Control Name	Control Type	Notes
Add Project line-items	Heading	None
Search Bar	Textbox	This is used to get the search parameter from the Operations Manager.
Sort	Drop down	This is used to get the filter option from the Operations Manager.
Line-Item Table	Table	This table will contain the saved **Line-Item** information per row.
Add	Button	This button will be used by the Operations manager to select line-items.
Add Table	Table	This table will contain the selected line-items chosen by the Operations Manager.
Remove	Button	This will be used to remove a line-item in the “Add Table”.
Submit	Button	This will be used to submit the newly added line-items
Cancel	Button	This is used to cancel or navigate back.

ALT Step 7a:

The Operations Manager wants to choose the necessary line-items to be added to the project line-item list.

ALT Step 8a:

The Operations Manager selects all the necessary line-items by using the “Add” button and then clicks the “Submit” button.

ALT Step 9a:

The system displays the newly selected line-items in a "**Confirm additional line-items**" modal and prompts the operations manager for confirmation.

Control Name	Control Type	Notes
Confirm Added Project line-items	Modal	None
Confirm Added Project line-items	Heading	None
Results Table	Table	This is used to display the selected line-items
Confirm	Button	This is used when the user wants to confirm the line-items to be added.
Cancel	Button	This is used when the user wants to confirm the items are incorrect, or navigate back.

ALT Step 10a:

The Operations Manager wants to confirm the add line-items.

ALT Step 11a:

The Operations Manager clicks "Confirm" button to confirm the added line-items.

ALT Step 12a:

The system stores the added line-items into the project line-item table.

Table	Attribute
Project Line-Item	Project Line-Item ID
	Project ID

The system prompts the Operations Manager that the line-items have been successfully added and saved.

Control Name	Control Type	Notes
Alert	Alert	None
Updates Confirmed	Label	This is used to inform the Operations Manager that the updates were confirmed and saved.
OK	Button	This is used to acknowledge the alert.

ALT Step 13a:

The Operations Manager clicks "OK" button.

Return to Step 2

ALT Step 4b:

The Operations Manager clicks the "Remove" button on a line-item record in the results table.

ALT Step 5b:

The system prompts the Operations Manager with a "**Remove Project Line-Item**" modal, to confirm that the line-item will be removed:

Control Name	Control Type	Notes
Alert	**Alert**	None
Remove Line-Item	**Label**	This is used to ask the Operations Manager for confirmation to remove line-item.
Yes	**Button**	This is used to confirm the alert.
No	**Button**	This is used to exit the use Alternate Step (Return to Step 3)

ALT Step 6b:

The Operations Manager wants to confirm removing the line-item.

ALT Step 7b:

The Operations Manager clicks the "Yes" button the alert.

ALT Step 8b:

The system removes the line-item from the project's line-item list. And prompts the Operations Manager that the line-item was remove with an alert.

Control Name	Control Type	Notes
Alert	**Alert**	None
Line-Item Removed	**Label**	This is used to inform the Operations Manager that the lie-item was removed.
OK	**Button**	This is used to acknowledge the alert.

ALT Step 9b:

The Operations Manager clicks the OK button the alert.

Return to Step 2

CONCLUSION:	The use case concludes once The Operations Manager confirms the changes and the system saves the required edits.
POST-CONDITION:	• The Project Line-Item List is updated
BUSINESS RULES:	• Only the Operations Manager has access to editing the project line-item list. • Only the project line-item list of the current project [This use case can only be accessed by navigating through the project's homepage] can be edited.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None
ASSUMPTIONS:	None
OPEN ISSUES:	None

1.2.8.3 Create Supplier Purchase Order

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Vincent Yu</td><td colspan="2">Date: 28/06/2021</td><td>Version: 3.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Create Supplier Purchase Order</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">8.3</td><td rowspan="3">Business Requirements: □
System Analysis: □
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Financial Officer (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">Managing Director</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case narrative describes the process the Financial Officer will follow when they want to create and send a new order to a supplier. The system will capture and validate the supplier's order information if the information was validated successfully the system will store it in the Supplier_Purchase_Order table. A supplier order consists out of line items that are stored in the Project_Line_Item table.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• Financial Officer needs to be logged on to the system.
• Suppliers have to be on the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The Financial Officer wants to send an order to a supplier.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td>Step 1:
The financial officer clicks on the "Supplier Order" card and then</td><td></td><td>Step 2:
The system invokes use case 8.1 Search Project Line-Item.
The system invokes use case 9.2 Search Supplier.</td></tr>
</table>

<table>
<tr><td></td><td>clicking “Create Supplier Purchase Orders” to navigate to the “Create Supplier Order” screen.</td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td>
Step 3:
The system responds by loading the “Create Supplier Quote” screen already populated with the Project Line-Item and Supplier table information using the MySql Server in the .NET Controller to read the Project Line-Item and Supplier table using LINQ and LAMBDA Queries.
<table>
<tr><th>Table</th><th>Attribute</th></tr>
<tr><td rowspan="3">Project Line-Item</td><td>Project Line-Item ID</td></tr>
<tr><td>Project ID</td></tr>
<tr><td>Quantity</td></tr>
<tr><td rowspan="2">Supplier</td><td>Supplier ID</td></tr>
<tr><td>Supplier Name</td></tr>
</table>
The “Create Supplier Purchase Order” screen has the following controls:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Create Supplier Purchase Order</td><td>Heading</td><td>None</td></tr>
<tr><td>Description</td><td>Label</td><td>This used to guide the user.</td></tr>
<tr><td>Results Table</td><td>Table</td><td>This table will contain the saved Project Line-Item information per row.</td></tr>
<tr><td>Assign Supplier Column</td><td>Column in Results</td><td>Each record in the Results Table will have</td></tr>
</table>
</td></tr>
</table>

<table>
<tr>
<td></td>
<td></td>
<td></td>
<td>
<table>
<tr><td></td><td>table (Dropdown)</td><td>a “Assign Supplier” dropdown control. This allows the User to assign the Supplier for that specific line-item.</td></tr>
<tr><td>Submit</td><td>Button</td><td>This is used when the user wishes to submit the supplier assignments.</td></tr>
<tr><td>Cancel</td><td>Button</td><td>This is used to help the User navigate back or cancel.</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 4:
The financial officer uses the “Assign Supplier” Dropdown control to assign suppliers to each line-item.
Then clicks the submit button when complete</td>
<td></td>
<td>Step 5:
The system captures and validates the provided details supplied by the financial is in the appropriate format and is complete. [ALT]
<table>
<tr><th>Input Name</th><th>Attribute in Table</th><th>Notes</th></tr>
<tr><td>Supplier Name</td><td>Supplier Name</td><td>Required, maximum of 128 characters.</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Step 6:
The system prompts the financial officer to confirm that the supplier purchase orders are correct. The system responds by loading the “confirm supplier purchase order” modal which has the following controls:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Confirm Supplier Purchase Order</td><td>Heading</td><td>None</td></tr>
</table>
</td>
</tr>
</table>

			<table><tr><td>Description</td><td>Label</td><td>This used to guide the user.</td></tr><tr><td>Supplier PO Group Box</td><td>Group Box</td><td>This is used to group and display the line-items for a specific supplier. A group Box will be generated for each PO.</td></tr><tr><td>Supplier Name</td><td>Label</td><td>This is used to identify the supplier of the specific PO.</td></tr><tr><td>Results Table</td><td>Table</td><td>This is used to display the lien-items to be ordered from the specific supplier.</td></tr><tr><td>Confirm</td><td>Button</td><td>This used when the financial officer wants to confirm the create d POs.</td></tr><tr><td>Cancel</td><td>Button</td><td>This is used to help the User navigate back or cancel.</td></tr></table>
	Step 7: The financial officer clicks the “Confirm” button. **[ALT]**		**Step 8:** The system stores the newly created Supplier Purchases Orders in the **Supplier Purchase Order** table: <table><tr><th>Table</th><th>Attribute</th></tr><tr><td rowspan="5">Supplier Purchase Order</td><td>Supplier Purchase Order ID</td></tr><tr><td>Supplier Purchase Order</td></tr><tr><td>Supplier ID</td></tr><tr><td>Date</td></tr><tr><td>PO Amount</td></tr></table> The system alerts the financial officer that the supplier purchase orders have

been successfully created and saved. The "**Supplier Purchase orders Created"** alert is displayed with the following controls.

Control Name	Control Type	Notes
Alert	**Alert**	None
Supplier POs Created	**Label**	This is used to inform the financial officer that the supplier POs were created and saved.
OK	**Button**	This is used to acknowledge the alert.

Step 9:
The financial officer clicks the "OK" button.

Step 10:
The system alerts the financial officer that the supplier purchase orders will be sent to her. The system informs the financial officer that these Supplier POs need to be sent to the relevant suppliers.

Control Name	Control Type	Notes
Alert	**Alert**	None
Send POs to Relevant Suppliers	**Label**	This is used to inform the financial officer that the supplier POs need to be sent to the correct suppliers.
OK	**Button**	This is used to acknowledge the alert.

Step 11:
The financial officer clicks the "OK" button.

Step 12:
The system retrieves the following attribute(s) from the **Supplier Purchase Order** table:

Table	Attribute	Notes
Supplier Purcha	**Supplier Purcha**	This is the stored PDF supplier purchase order document

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>se Order</th><th>se Order</th><th></th></tr>
</table>
These supplier POs are then sent to the financial officer as requested.
</td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 3:
The provided information is not in the correct format or not completed.
The system displays a Dialog of the validation error to the financial officer.
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Validation Error</td><td>Label</td><td>None</td></tr>
<tr><td>Error Message</td><td>Label</td><td>This is used to inform the Financial Officer about the validation error.</td></tr>
<tr><td>OK</td><td>Button</td><td>This is used by the Financial Officer to acknowledge the error.</td></tr>
</table>
The financial clicks the “OK” button
Return to Step 2</td></tr>
<tr><td></td><td colspan="3">ALT Step 7:
The financial officer clicks the “Cancel” button.
ALT Step 8:
The system displays a “Supplier Purchase Orders Not Created” Alert.
ALT Step 9:
The financial officer clicks the “OK” button
Return to Step 2</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">The use case concludes when the supplier orders are stored in the system and the supplier orders are sent to the relevant suppliers.</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">• The system sent supplier purchase orders to the financial officer.</td></tr>
<tr><td>BUSINESS RULES:</td><td colspan="3">• Only the financial officer has access to create supplier purchase orders.</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td colspan="3">None</td></tr>
<tr><td>ASSUMPTIONS:</td><td colspan="3">None</td></tr>
<tr><td>OPEN ISSUES:</td><td colspan="3">None</td></tr>
</table>

1.2.8.4 Capture Supplier Quote

Gohvan Construction			
Author (s): Vincent Yu	**Date: 28/06/2021**	**Version: 3.0**	
USE CASE NAME:	**Capture Supplier Quote**	**USE CASE TYPE**	
USE CASE ID:	**8.4**	**Business Requirements:** □	
PRIORITY:	High	**System Analysis:** □	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	Financial Officer (PBA)		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	Managing Director		
DESCRIPTION:	This use case describes the event when a supplier sends a quotation to the Financial Officer. The Financial Officer will either accept or reject the quote. The system will capture and store the accepted quote.		
PRE-CONDITION:	• Financial Officer has to be logged into the system. • Financial Officer needs to be in the **specific project** to capture Supplier Quotes.		
TRIGGER:	The Supplier sends a quotation for the financial officer to accept or reject.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:** Manual Action	Automated Action
	Step 1: The Financial Officer clicks on the "**Supplier Order**" card on the project's home screen. And		**Step 2:** The system invokes use case 5.1 "**Search Supplier**"

<table>
<tr><td></td><td>then clicks the “Capture Quote” button</td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td>
Step 3:
The system responds by loading the “Capture Supplier Quote” screen already populated with the Supplier table information using the MySql Server in the .NET Controller to read the Supplier table using LINQ and LAMBDA Queries.
<table>
<tr><th>Table</th><th>Attribute</th></tr>
<tr><td rowspan="2">Supplier</td><td>Supplier ID</td></tr>
<tr><td>Supplier Name</td></tr>
</table>
The system prompts the financial officer to upload and provide details for the supplier quote.
The “Capture Supplier Quote” screen has the following controls:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Capture Supplier Quote</td><td>Heading</td><td>None</td></tr>
<tr><td>Description</td><td>Label</td><td>This is used to inform the financial officer about what is required.</td></tr>
<tr><td>Upload</td><td>File Upload</td><td>This is used to upload the supplier quote.</td></tr>
<tr><td>Supplier Quote Details</td><td>Label</td><td>None</td></tr>
<tr><td>Quote Number</td><td>Label & Textbox</td><td>These controls are used to get the quote number of the supplier quote.</td></tr>
</table>
</td></tr>
</table>

			Supplier — **Label & Dropdown** — These controls are used to select the supplier the quote was received from. **Quote Amount** — **Label & Textbox** — This is used to get the amount of the supplier quote. **Submit** — **Button** — This is used when the financial officer wants to submit the uploaded supplier quote. **Cancel** — **Button** — This is used to help the Financial Officer navigate back.
	Step 4: The Financial Officer uses the "Upload" control to upload the pdf supplier quote. Uses the Supplier dropdown to select the supplier. Uses the Quote Number and Amount controls to enter the Quote Number and Amount. Then clicks the submit button.		**Step 5:** The system captures and validates the provided details supplied by the Financial Officer. **[ALT]** **Input Name** / **Attribute in Table** / **Notes** **Supplier** / **Supplier Name** / Required, of type string. **Supplier Quote** / **Supplier Quote** / Required, of type file. **Quote Number** / **Supplier Quote ID** / Required, of type number **Quote Amount** / **Supplier Quote Amount** / Required, maximum of 128 characters
			Step 6:

The system prompts the financial to confirm the provided information, using the "**Confirm Supplier Quote**" modal which has the following controls:

Control Name	Control Type	Notes
Confirm Supplier Quote	**Heading**	None
Supplier Quote	**PDF viewer**	This is the pdf document uploaded by the financial officer.
Quote number	**Label**	This is used to display the supplier quote number..
Quote amount	**Label**	This is used to display the Amount.
Supplier	**Label**	This is used to display the Supplier.
Confirm	**Button**	This is used to confirm the supplier quote.
Cancel	**Button**	This is used to help the Financial Officer navigate back.

Step 7:
The Financial Officer clicks the "**Confirm**" button.
[ALT]

Step 8:
The system stores the following attribute(s) in the **Supplier Quote** table:

Table	Attribute	Notes
Supplier Quote	**Supplier Quote ID**	This is the primary key used to track supplier quotes.
	Supplier Quote	This is the pdf document of the supplier quote.
	Supplier ID	This the supplier who sent the quote.
	Date received	This is the date the supplier quote was received.
	Total Amount	This is the amount on the supplier quote.

The system prompts the financial officer that the POP was captured successfully with a "**Supplier Quote Captured**" alert:

Control Name	Control Type	Notes
Alert	**Alert**	None
Supplier Quote Captured	**Label**	This is used to inform the financial officer that the supplier quote was captured.
OK	**Button**	This is used to acknowledge the alert.

Step 9:
The financial officer clicks the "OK" button.

<table>
<tr><td>ALTERNATE COURSES:</td><td>ALT Step 5:
The provided information is not in the correct format or not completed.
The system displays a Dialog of the validation error to the financial officer.
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Validation Error</td><td>Label</td><td>None</td></tr>
<tr><td>Error Message</td><td>Label</td><td>This is used to inform the Financial Officer about the validation error.</td></tr>
<tr><td>OK</td><td>Button</td><td>This is used by the Financial Officer to acknowledge the error.</td></tr>
</table>
The financial clicks the “OK” button
Return to Step 2</td></tr>
<tr><td></td><td>ALT Step 7:
The financial officer clicks the cancel button.
ALT Step 8:
The system displays a “Supplier Quote Not Captured” Alert.
ALT Step 9:
The financial officer clicks the “ok” button.
Return to Step 2</td></tr>
<tr><td>CONCLUSION:</td><td>The use case concludes when the supplier quote is captured and stored on the system</td></tr>
<tr><td>POST-CONDITION:</td><td>• The supplier quote is uploaded</td></tr>
<tr><td>BUSINESS RULES:</td><td>• Only the Financial Officer can capture and upload quotes.</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td>None</td></tr>
<tr><td>ASSUMPTIONS:</td><td>None</td></tr>
<tr><td>OPEN ISSUES:</td><td>None</td></tr>
</table>

1.2.8.5 Search Supplier Quote

Gohvan Construction			
Author (s): Vincent Yu	**Date: 28/06/2021**	**Version: 3.0**	
USE CASE NAME:	**Search Supplier Quote**	**USE CASE TYPE**	
USE CASE ID:	**8.5**	**Business Requirements:** □	
PRIORITY:	Medium	**System Analysis:** □	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	User (PBA) [Operations Manager and Financial Officer]		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	Managing Director		
DESCRIPTION:	This use case describes the event when a user wants to search for a supplier quote on the system. This use case can also be used/called by other use case.		
PRE-CONDITION:	• The User needs to be logged on to the system. • Only stored supplier quotes can be retrieved.		
TRIGGER:	The User would like to search for a project's line-item list.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:** Manual Action	Automated Action
	Step 1: The User clicks on the "**Menu**" to navigate to the click on the **Supplier Order** tab and then clicking		**Step 2:** The system responds by loading the "**Search Supplier Quote**" screen already populated with the **Supplier Quote** table information using the MySql Server in the .NET Controller to read the **Supplier Quote** table using LINQ and LAMBDA Queries.

	"Search Supplier Quote"		**Table** / **Attribute** **Supplier Quote** / **Supplier Quote ID** **Supplier Name** **Supplier Quote** **Date Received** **Total Amount** The "**Search Supplier Quote**" screen has the following controls: **Control Name** / **Control Type** / **Notes** **Search Supplier Quote** / **Heading** / None **Search Bar** / **Textbox** / This is used to get the search parameter from the User. **Sort** / **Drop down** / This is used to get the filter option from the User. **Results Table** / **Table** / This table will contain the saved **Supplier Quote** information per row. **Download Supplier Quote** / **Button** / Each record in the Results Table will have a "Download Supplier Quote" button. This allows the User to download the stored Supplier Quote. **Back** / **Button** / This is used to help the User navigate back.
	Step 3:		**Step 4:**

<table>
<tr>
<td></td>
<td>The User enters the search parameter [Supplier ID, Supplier Name, Project ID, or Project Name] in the Search Bar control.</td>
<td></td>
<td>The system captures and validates the search parameter [Supplier ID, Supplier Name, Project ID, or Project Name] supplied by the user is in the appropriate format and is complete. [ALT]
<table>
<tr><th>Input Name</th><th>Attribute in Table</th><th>Notes</th></tr>
<tr><td>Search Parameter [Supplier ID]</td><td>Supplier_ID</td><td>Required, maximum of 13 characters.</td></tr>
<tr><td>Search Parameter [Supplier Name]</td><td>Supplier_Name</td><td>Required, maximum of 256 characters</td></tr>
<tr><td>Search Parameter [Project ID]</td><td>Project_ID</td><td>Required, maximum of 13 characters.</td></tr>
<tr><td>Search Parameter [Project Name]</td><td>Project_Name</td><td>Required, maximum of 256 characters</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Step 5:
The system Invokes use case 9.2 Search Supplier.
The system invokes use case 4.9 Search Project.</td>
</tr>
</table>

<table>
<tr><td rowspan="3"></td><td></td><td></td><td>Step 6:
The system retrieves the Supplier Quote record result based on the search parameter supplier by the User from the Supplier Quote table:
<table><tr><th>Table</th><th>Attribute</th></tr><tr><td rowspan="5">Supplier Quote</td><td>Supplier Quote ID</td></tr><tr><td>Supplier Name</td></tr><tr><td>Supplier Quote</td></tr><tr><td>Date Received</td></tr><tr><td>Total Amount</td></tr></table>The Supplier Quote record is displayed in the Results Table control. [ALT]</td></tr>
<tr><td>Step 7:
The User clicks the “Download Supplier Quote” button.</td><td></td><td>Step 8:
The system retrieves the following attribute(s) from the Supplier Quote table:
<table><tr><th>Table</th><th>Attribute</th><th>Notes</th></tr><tr><td>Supplier Quote</td><td>Supplier Quote</td><td>This is the stored PDF supplier quote document</td></tr></table>This document is then sent to the user as requested.</td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 4:
The entered search parameter is not in the correct format or not completed.

The system displays a Dialog of the validation error to the financial officer.
<table><tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Validation Error</td><td>Label</td><td>None</td></tr><tr><td>Error Message</td><td>Label</td><td>This is used to inform the Financial Officer about the validation error.</td></tr><tr><td>OK</td><td>Button</td><td>This is used by the Financial Officer to acknowledge the error.</td></tr></table>The User clicks the “OK” button
Return to Step 2</td></tr>
</table>

<table>
<tr><td></td><td>ALT Step 6:
There is no project in the system that corresponds to the entered search parameter.

The system displays the following control to inform the user that no record was found based on the provided search parameter:
<table><tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>No Results Found</td><td>Label</td><td>This is used to inform the financial officer no record was found.</td></tr></table>
The user clears the Search Bar control.
Return to Step 2</td></tr>
<tr><td>CONCLUSION:</td><td>This use case concludes when the system retrieves and displays the project line-item list search results.</td></tr>
<tr><td>POST-CONDITION:</td><td>• The Supplier Quote table was searched for an entry that corresponds with the search parameter.
• The searched supplier quote is retrieved and downloadable.</td></tr>
<tr><td>BUSINESS RULES:</td><td>• The supplier quote needs to be present on the system in order to be searched for.</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td>None</td></tr>
<tr><td>ASSUMPTIONS:</td><td>None</td></tr>
<tr><td>OPEN ISSUES:</td><td>None</td></tr>
</table>

1.2.8.6 Search Supplier Purchase Order

Gohvan Construction			
Author (s): Vincent Yu	**Date: 28/06/2021**	**Version: 3.0**	
USE CASE NAME:	**Search Supplier Purchase Order**	**USE CASE TYPE**	
USE CASE ID:	**8.6**	**Business Requirements:** □	
PRIORITY:	Medium	**System Analysis:** □	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	User (PBA) [Operations Manager and Financial Officer]		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	Managing Director		
DESCRIPTION:	This use case describes the event when a user wants to search for a supplier purchase order on the system. This use case can also be used/called by other use case.		
PRE-CONDITION:	• The User needs to be logged on to the system. • Only stored supplier purchase orders can be retrieved.		
TRIGGER:	The User would like to search for a supplier purchase order.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:** Manual Action	Automated Action
	Step 1: The User clicks on the "**Menu**" to navigate to the click on the **Supplier Order** tab and		**Step 2:** The system responds by loading the "**Search Supplier Purchase Order**" screen already populated with the **Supplier Purchase Order** table information using the MySql Server in the .NET Controller to read the **Supplier**

<table>
<tr>
<td></td>
<td>then clicking “Search Supplier Purchase Order”</td>
<td></td>
<td>
Purchase Order table using LINQ and LAMBDA Queries.
<table>
<tr><th>Table</th><th>Attribute</th></tr>
<tr><td rowspan="5">Supplier Purchase Order</td><td>Supplier Purchase Order ID</td></tr>
<tr><td>Supplier Name</td></tr>
<tr><td>Supplier Purchase Order</td></tr>
<tr><td>Date Received</td></tr>
<tr><td>Total Amount</td></tr>
</table>
The “Search Supplier Purchase Order” screen has the following controls:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Search Supplier Quote</td><td>Heading</td><td>None</td></tr>
<tr><td>Search Bar</td><td>Textbox</td><td>This is used to get the search parameter from the User.</td></tr>
<tr><td>Sort</td><td>Drop down</td><td>This is used to get the filter option from the User.</td></tr>
<tr><td>Results Table</td><td>Table</td><td>This table will contain the saved Supplier Purchase Order information per row.</td></tr>
<tr><td>Download Supplier Purchase Order</td><td>Button</td><td>Each record in the Results Table will have a “Download Supplier Purchase Order” button. This allows the User to download the</td></tr>
</table>
</td>
</tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td></td><td></td><td>stored Supplier Purchase Order.</td></tr>
<tr><td>Back</td><td>Button</td><td>This is used to help the User navigate back.</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 3:
The User enters the search parameter [Supplier ID, Supplier Name, Project ID, or Project Name] in the Search Bar control.</td><td></td><td>
Step 4:
The system captures and validates the search parameter [Supplier ID, Supplier Name, Project ID, or Project Name] supplied by the user is in the appropriate format and is complete. [ALT]
<table>
<tr><th>Input Name</th><th>Attribute in Table</th><th>Notes</th></tr>
<tr><td>Search Parameter [Supplier ID]</td><td>Supplier_ID</td><td>Required, maximum of 13 characters.</td></tr>
<tr><td>Search Parameter [Supplier Name]</td><td>Supplier_Name</td><td>Required, maximum of 256 characters</td></tr>
<tr><td>Search Parameter [Project ID]</td><td>Project_ID</td><td>Required, maximum of 13 characters.</td></tr>
<tr><td>Search Parameter [Project Name]</td><td>Project_Name</td><td>Required, maximum of 256 characters</td></tr>
</table>
</td></tr>
<tr><td></td><td></td><td></td><td>Step 5:
The system Invokes use case 9.2 Search Supplier.
The system invokes use case 4.9 Search Project.</td></tr>
</table>

<table>
<tr><td rowspan="2"></td><td></td><td></td><td>Step 6:
The system retrieves the Supplier Purchase Order record result based on the search parameter supplier by the User from the Supplier Purchase Order table:
<table><tr><th>Table</th><th>Attribute</th></tr><tr><td rowspan="5">Supplier Purchase Order</td><td>Supplier Purchase Order ID</td></tr><tr><td>Supplier Name</td></tr><tr><td>Supplier Purchase Order</td></tr><tr><td>Date Received</td></tr><tr><td>Total Amount</td></tr></table>The Supplier Purchase Order record is displayed in the Results Table control. [ALT]</td></tr>
<tr><td>Step 7:
The User clicks the “Download Supplier Purchase Order” button.</td><td></td><td>Step 8:
The system retrieves the following attribute(s) from the Supplier Purchase Order table:
<table><tr><th>Table</th><th>Attribute</th><th>Notes</th></tr><tr><td>Supplier Purchase Order</td><td>Supplier Purchase Order</td><td>This is the stored PDF supplier Purchase Order document</td></tr></table>This document is then sent to the user as requested.</td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 4:
The entered search parameter is not in the correct format or not completed.
The system displays a Dialog of the validation error to the financial officer.
<table><tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Validation Error</td><td>Label</td><td>None</td></tr><tr><td>Error Message</td><td>Label</td><td>This is used to inform the Financial Officer about the validation error.</td></tr></table></td></tr>
</table>

<table>
<tr><td></td><td>
<table>
<tr><td>OK</td><td>Button</td><td>This is used by the Financial Officer to acknowledge the error.</td></tr>
</table>
The User clicks the “OK” button
Return to Step 2
</td></tr>
<tr><td></td><td>
ALT Step 4:
There is no project in the system that corresponds to the entered search parameter.
The system displays the following control to inform the user that no record was found based on the provided search parameter:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>No Results Found</td><td>Label</td><td>This is used to inform the financial officer no record was found.</td></tr>
</table>
The user clears the Search Bar control.
Return to Step 2
</td></tr>
<tr><td>CONCLUSION:</td><td>This use case concludes when the system retrieves and displays the project line-item list search results.</td></tr>
<tr><td>POST-CONDITION:</td><td>

The Supplier Purchase Order table was searched for an entry that corresponds with the search parameter.
The searched supplier Purchase Order is retrieved and downloadable.

</td></tr>
<tr><td>BUSINESS RULES:</td><td>

The supplier Purchase Order needs to be present on the system in order to be searched for.

</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td>None</td></tr>
<tr><td>ASSUMPTIONS:</td><td>None</td></tr>
<tr><td>OPEN ISSUES:</td><td>None</td></tr>
</table>

1.2.9. Supplier Subsystem

1.2.9.1 Add Supplier

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Amore Rossouw</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Add Supplier</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">9.1</td><td>Abstract: ☐</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Medium</td><td>Extension: ☑</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td><td></td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">The Procurement and Logistics Officer (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">• None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case narrative describes the process the Procurement and Logistics Officer will follow when they want to create a new supplier. The system will capture and validate the supplier information if the information was validated successfully the system will store it in the SUPPLIER table. Only the Procurement and Logistics Officer must be able to add new suppliers.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">Procurement and Logistics Officer must be logged into the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The Procurement and Logistics Officer wants to add a new supplier.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td></td><td>Step 1:
The system responds by loading the “Add Supplier” Screen with Supplier table information using the SQL Read query with the following controls:
<table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Add Supplier</td><td>Header</td><td>none</td></tr></table></td></tr>
</table>

			Supplier Name	Label	none
			Supplier Name	Input	none
			Email Address	Label	none
			Email Address	Input	none
			Telephone Number	Label	none
			Telephone Number	Input	none
			Fax Number	Label	none
			Fax Number	Input	none
			Vat Number	Label	none
			Vat Number	Input	none
			PO box	Label	none
			PO box	Input	none
			Business address	Label	none
			Street address	Label	none
			Street address	Input	none
			City	Label	none
			City	Input	none
			Province	Label	none
			Province	Input	none
			Zip code	Label	none
			Zip code	Input	none
			Upload Supplier documents	Button	This control allows the user to upload a file from their computer

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>Supplier documents Table</td><td>Table</td><td>Populated with the “Supplier_ Document_ Name” attribute of each Supplier document that was added with the “Upload supplier document” Button. A “Remove” button will be next to each Supplier document</td></tr>
<tr><td>Remove</td><td>button</td><td>This control will allow the user to remove a document that they have uploaded.</td></tr>
<tr><td>Add Supplier</td><td>button</td><td>none</td></tr>
<tr><td>Cancel</td><td>button</td><td>none</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 2:
The Procurement and Logistics Officer fills in the necessary information into the appropriate input fields requested by the system.</td><td></td><td></td></tr>
<tr><td></td><td>Step 3:
The Procurement and Logistics</td><td></td><td>Step 4:
The system captures and validates the information supplied by the Procurement and Logistics Officer against the Supplier table using Angular Validation:</td></tr>
</table>

<table>
<tr>
<td></td>
<td>Officer selects the “Add Supplier” button.
[ALT]</td>
<td></td>
<td>[ALT]

<table>
<tr><th>Input Name</th><th>Attribute in table</th><th>Validation requirements</th></tr>
<tr><td>Supplier Name</td><td>Supplier_ Name</td><td>required</td></tr>
<tr><td>Email Address</td><td>Email</td><td>Must be of type email</td></tr>
<tr><td>Telephone Number</td><td>Tel_No</td><td>required</td></tr>
<tr><td>Fax Number</td><td>Fax_No</td><td>required</td></tr>
<tr><td>Vat Number</td><td>Vat_No</td><td>required</td></tr>
<tr><td>PO box</td><td>Po_Box</td><td>required</td></tr>
<tr><td>Street address</td><td>Street_ Address</td><td>required</td></tr>
<tr><td>City</td><td>City</td><td>required</td></tr>
<tr><td>Province</td><td>Province</td><td>required</td></tr>
<tr><td>Zip code</td><td>Zip_code</td><td>required</td></tr>
</table>

The system captures and validates the information supplied by the Procurement and Logistics Officer against the Supplier Documents table using Angular Validation:

<table>
<tr><th>Input Name</th><th>Attribute in table</th><th>Validation requirements</th></tr>
<tr><td>Supplier Document Name</td><td>Supplier_ Document_ Name</td><td>required</td></tr>
<tr><td>Supplier Document</td><td>Supplier_ document</td><td>required</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Step 5:
The system displays an Angular Material Confirmation Modal, with the following controls, to prompt the Procurement and Logistics Officer for confirmation of the entered details:</td>
</tr>
</table>

			<table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Confirm information</td><td>Header</td><td>none</td></tr><tr><td>Please confirm that the details entered are correct?</td><td>Label</td><td>none</td></tr><tr><td>Yes</td><td>button</td><td>This button confirms that the entered details were correct</td></tr><tr><td>No</td><td>button</td><td>This button confirms that the entered details were not correct</td></tr></table>
	Step 6: The Procurement and Logistics Officer confirms the captured information by clicking the “Yes” button. **[ALT]**		**Step 7:** The system uses a SQL insert query to save the captured information in the **SUPPLIER** table as follows: <table><tr><th>Details</th><th>Attribute in table</th><th>Notes</th></tr><tr><td>Supplier ID</td><td>Supplier_ ID</td><td>ID is generated by the system by using the last ID in the table and incrementing it by one.</td></tr><tr><td>Supplier Name</td><td>Supplier_ Name</td><td>none</td></tr></table>

Email Address	Email	Must be of type email
Telephone Number	Tel_No	none
Fax Number	Fax_No	none
Vat Number	Vat_No	none
PO box	Po_Box	none
Street address	Street_ Address	none
City	City	none
Province	Province	none
Zip code	Zip_code	none

The system uses a SQL insert query to save the captured information in the **SUPPLIER DOCUMENT** table as follows:

Details	Attribute in table	Notes
Supplier Document ID	Supplier_ document_ ID	ID is generated by the system by using the last ID in the table and incrementing it by one.
Supplier Document Name	Supplier_ Document_ Name	none
Supplier Document	Supplier_ document	None

[ALT]

Step 8:
The system notifies Procurement and Logistics Officer that the Supplier was
added successfully, using an Angular Material Success Notification Modal with
the following controls:

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Success</td><td>Header</td><td>none</td></tr>
<tr><td>The Supplier was successfully created</td><td>Label</td><td>none</td></tr>
<tr><td>OK</td><td>button</td><td>None</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 9:
The Procurement and Logistics Officer clicks the “Ok” button.</td><td></td><td></td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 3:
The Procurement and Logistics Officer selects the “Cancel” button. The use case is terminated.</td></tr>
<tr><td></td><td colspan="3">ALT Step 4:
• The captured details are not valid. The system displays a notification indicating that the information received is incorrect or incomplete with an Angular Material Invalid Notification Modal as follows:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Information Invalid</td><td>Header</td><td>none</td></tr>
<tr><td>The entered details were incorrect. Please fill in the correct information.</td><td>Label</td><td>none</td></tr>
<tr><td>OK</td><td>button</td><td>None</td></tr>
</table>
The system returns to step 1.</td></tr>
<tr><td></td><td colspan="3">ALT Step 6:
The Procurement and Logistics Officer does not confirm that the information captured is correct by clicking the “No” button.
The system returns to step 1.</td></tr>
<tr><td></td><td colspan="3">ALT Step 7:
The system does not successfully add the Supplier information to the SUPPLIER and SUPPLIER DOCUMENT table. The system notifies the</td></tr>
</table>

<table>
<tr><td></td><td>user that the Supplier has not been added successfully with an Angular Material Error Modal as follows:

<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Error</td><td>Header</td><td>none</td></tr>
<tr><td>An error has occurred while saving the Supplier details. Please try again.</td><td>Label</td><td>none</td></tr>
<tr><td>OK</td><td>button</td><td>None</td></tr>
</table>

The system terminates this use case.</td></tr>
<tr><td>CONCLUSION:</td><td>This use case concludes when the Procurement and Logistics Officer has created a new supplier successfully.</td></tr>
<tr><td>POST-CONDITION:</td><td>● The supplier has been added to the SUPPLIER table successfully.</td></tr>
</table>

1.2.9.2 Search Supplier

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Amore Rossouw</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Search Supplier</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">9.2</td><td rowspan="3">Business Requirements: ☐
System Analysis: o
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Low</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">The User (PBA) [management director, director, operations manager, Financial officer and the Procurement and Logistics Officer.]</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">• None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case narrative describes the process the user will follow when they want to search for a Supplier or when the system invokes this use case.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The User must be logged into the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The user wants to search for a supplier.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1:
The User wants to search for an existing supplier.</td><td>Step 2:
The Procurement and Logistics Officer clicks on the “Supplier” option in the side menu bar on the home screen.</td><td>Step 3:
The system responds by loading the “Suppliers “screen with the following controls:
<table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Supplier</td><td>Header</td><td>none</td></tr><tr><td>+</td><td>Button</td><td>This control is used to add a Supplier</td></tr><tr><td>Search Supplier</td><td>Label</td><td>none</td></tr></table></td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>Search Parameter</td><td>Input</td><td>none</td></tr>
<tr><td>Search</td><td>Button</td><td>none</td></tr>
<tr><td>Supplier Table</td><td>Table</td><td>This table is populated with the “supplier name” attributes from the SUPPLIER table. An “edit” button will be next to each Supplier record.</td></tr>
<tr><td>Edit Supplier</td><td>Button</td><td>This control is used to edit a Supplier</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 4:
The Procurement and Logistics Officer clicks on the “Search Supplier” input and enters the requested search parameter. Thereafter the user clicks on the “Search Supplier” button.</td><td></td><td>Step 5:
The system captures and validates the information, supplied by the User, against the SUPPLIER table using Angular Validation:

<table>
<tr><td>Input Name</td><td>Attribute in table</td><td>Validation requirements</td></tr>
<tr><td>Supplier Name</td><td>Supplier_Name</td><td>Required</td></tr>
</table>

[ALT]</td></tr>
<tr><td></td><td></td><td></td><td>Step 6:
A SQL read query is used to read the following attributes from the following tables based on the parameter provided:

The Table is searched based on the Supplier Name:

SUPPLIER Table:
• Supplier_ID
• Supplier_Name
• Email</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>

Tel_No
Fax_No
Vat_No
Po_Box
Street_Address
City
Province
Zip_code

The Table is searched based on the Supplier ID:

SUPPLIER DOCUMENTS Table:

Supplier_Document_ID
Supplier_Document_Name
Supplier_ID

The table attributes are read via an HttpGet method in the Typescript component and an API request.
</td></tr>
<tr><td></td><td></td><td></td><td>Step 7:
The system populates the ‘Supplier Table’ with the data received in the previous step
[ALT]</td></tr>
<tr><td></td><td></td><td>Step 8:
The User selects the Supplier from the search result by clicking on the “Supplier Name” label.
[ALT]</td><td>Step 9:
The system responds by loading the “View Supplier” page with the following controls:

<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Supplier Name</td><td>Header</td><td>This header’s text is dynamically changed to the selected Supplier’s name.</td></tr>
<tr><td>Email Address:</td><td>label</td><td>none</td></tr>
<tr><td>Email Address</td><td>label</td><td>This label’s text is dynamically changed to the selected Supplier’s email.</td></tr>
<tr><td>Telephone Number:</td><td>label</td><td>none</td></tr>
<tr><td>Telephone Number</td><td>label</td><td>This label’s text is dynamically</td></tr>
</table>
</td></tr>
</table>

					changed to the selected Supplier's telephone number.
			Fax Number:	label	none
			Fax Number	label	This label's text is dynamically changed to the selected Supplier's fax number.
			Vat Number:	label	none
			Vat Number	label	This label's text is dynamically changed to the selected Supplier's vat number.
			PO box:	label	none
			PO box	label	This label's text is dynamically changed to the selected Supplier's PO box.
			Street address	label	none
			Street address	label	This label's text is dynamically changed to the selected Supplier's Street address.
			City	label	none
			City	label	This label's text is dynamically changed to the selected Supplier's City.
			Province	label	none

			Province	label	This label's text is dynamically changed to the selected Supplier's Province.
			Zip code	label	none
			Zip code	label	This label's text is dynamically changed to the selected Supplier's Zip code.
			Supplier documents table	Table	This table is populated with the "Supplier document name" and "Supplier document date" attributes from the Supplier documents table. A "download" button will be next to each Supplier record.
			Download	button	This control is used to download the corresponding supplier document from the 'Supplier documents table'
			Cancel	button	none

ALTERNATE COURSES:

ALT Step 5:

The captured search parameters are not valid. The system displays an Angular Material Invalid Notification modal indicating that the information received is incorrect or incomplete:

Input Name	Control Type	Notes
Information Invalid	Header	none

<table>
<tr><td></td><td>
<table>
<tr><td>The entered parameters were incorrect. Please try again.</td><td>Label</td><td>none</td></tr>
<tr><td>OK</td><td>button</td><td>None</td></tr>
</table>
The system returns to step 3.
</td></tr>
<tr><td></td><td>ALT Step 7:
The supplier that the user searched does not exist within the database. The system notifies the user that no search results were found by displaying an Angular Material Error modal with the following controls:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>No Supplier found</td><td>Header</td><td>none</td></tr>
<tr><td>There was no Supplier found that matched your search request.</td><td>Label</td><td>none</td></tr>
<tr><td>OK</td><td>button</td><td>None</td></tr>
</table>
The system will return to Step 3.</td></tr>
<tr><td></td><td>ALT Step 8 (A):
The Procurement and Logistics Officer clicks on the “Add (+)” button. The system invokes use case “9.1 Add Supplier”</td></tr>
<tr><td></td><td>ALT Step 8 (B):
The Procurement and Logistics clicks on the “Edit” button for the Supplier they want to edit. The system invokes use case “9.3 Edit Supplier”</td></tr>
<tr><td>CONCLUSION:</td><td>This use case concludes when the system retrieves and displays the supplier search results.</td></tr>
<tr><td>POST-CONDITION:</td><td>• The searched supplier is retrieved and displayed.</td></tr>
<tr><td>BUSINESS RULES:</td><td>• Only the users mentioned above are allowed to search the supplier.</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td>none</td></tr>
<tr><td>ASSUMPTIONS:</td><td>none</td></tr>
<tr><td>OPEN ISSUES:</td><td>none</td></tr>
</table>

1.2.9.3 Edit Supplier

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Amore Rossouw</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Edit Supplier</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">9.3</td><td rowspan="3">Abstract : ☐
Extension: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Medium</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">The Procurement and Logistics Officer (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">• None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case narrative describes the process the Procurement and Logistics Officer will follow when they want to update an existing supplier. The system will capture and validate the supplier information if the information was validated successfully the system will update the supplier in the SUPPLIER table. Only the Procurement and Logistics Officer is allowed to update an existing supplier.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The user must be logged into the system.
• The supplier that the user wishes to update must exist in the database.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The Procurement and Logistics Officer wants to update an existing supplier.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td></td><td colspan="2">Step 1:
The system responds by loading the “Edit Supplier” screen with the following controls:

Input Name | Control Type | Notes</td></tr>
</table>

			Edit Supplier	Header	none
			Supplier Name	Label	none
			Supplier Name	Input	This control is Populated with the current Supplier_Name attribute that was obtained in the use case ”9.2 Search supplier”.
			Email Address	Label	none
			Email Address	Input	This control is Populated with the current Email attribute that was obtained in the use case ”9.2 Search supplier”.
			Telephone Number	Label	none
			Telephone Number	Input	This control is Populated with the current Tel_No attribute that was obtained in the use case ”9.2 Search supplier”.
			Fax Number	Label	none
			Fax Number	Input	This control is Populated with the current Fax_No attribute that was obtained in the use case ”9.2 Search supplier”.
			Vat Number	Label	none

			Vat Number	Input	This control is Populated with the current Vat_No attribute that was obtained in the use case ”9.2 Search supplier”.
			PO box	Label	none
			PO box	Input	This control is Populated with the current PO_box attribute that was obtained in the use case ”9.2 Search supplier”.
			Business address	Label	none
			Street address	Label	none
			Street address	Input	This control is Populated with the current Street_ Address attribute that was obtained in the use case ”9.2 Search supplier”.
			City	Label	none
			City	Input	This control is Populated with the current city attribute that was obtained in the use case ”9.2 Search supplier”.
			Province	Label	none
			Province	Input	This control is Populated with the current province attribute that

					was obtained in the use case ”9.2 Search supplier”.
			Zip code	Label	none
			Zip code	Input	This control is Populated with the current Zip_code attribute that was obtained in the use case ”9.2 Search supplier”.
			Supplier documents upload	button	This control will allow the user to upload a new document.
			Remove	button	This control will allow the user to remove a document that they have uploaded.
			Supplier documents Table	Table	Populated with the “Supplier_ Document_ Name” attribute of each Supplier document from the SUPPLIER DOCUMENT table and the documents that were added with the “Upload supplier document” Button. A “Remove” button will be next to each Supplier document
			Edit Supplier	Button	none

			Cancel | Button | none
	Step 2: The Procurement and Logistics Officer fills in the necessary information into the appropriate input fields requested by the system. The Procurement and Logistics Officer selects the "Edit Supplier" button. **[ALT]**		**Step 3**: The system captures and validates the information supplied by the Procurement and Logistics Officer against the **Supplier** table using Angular Validation: **[ALT]**

Input Name	Attribute in table	Validation requirements
Supplier Name	Supplier_ Name	required
Email Address	Email	Must be of type email
Telephone Number	Tel_No	required
Fax Number	Fax_No	required
Vat Number	Vat_No	required
PO box	Po_Box	required
Street address	Street_ Address	required
City	City	required
Province	Province	required
Zip code	Zip_code	required

The system captures and validates the information supplied by the Procurement and Logistics Officer against the **Supplier Documents** table using Angular Validation:

Input Name	Attribute in table	Validation requirements
Supplier Document Name	Supplier_ Document_ Name	required
Supplier Document	Supplier_ document	required

Step 4:

The system displays an Angular Material Confirmation Modal, with the following controls, to prompt the Procurement and Logistics Officer for confirmation of the entered details:

Input Name	Control Type	Notes
Confirm information	Header	none
Please confirm that the details entered are correct?	Label	none
Yes	button	This button confirms that the entered details were correct
No	button	This button confirms that the entered details were not correct

Step 5:

The Procurement and Logistics Officer confirms the captured information by clicking the “Yes” button.
[ALT]

Step 6:

The system uses a SQL update query to save the captured information in the **SUPPLIER** table as follows:

Details	Attribute in table	Notes
Supplier ID	Supplier_ ID	ID is generated by the system by using the last ID in the table and incrementing it by one.

			<table><tr><td>Supplier Name</td><td>Supplier_ Name</td><td>none</td></tr><tr><td>Email Address</td><td>Email</td><td>Must be of type email</td></tr><tr><td>Telephone Number</td><td>Tel_No</td><td>none</td></tr><tr><td>Fax Number</td><td>Fax_No</td><td>none</td></tr><tr><td>Vat Number</td><td>Vat_No</td><td>none</td></tr><tr><td>PO box</td><td>Po_Box</td><td>none</td></tr><tr><td>Street address</td><td>Street_ Address</td><td>none</td></tr><tr><td>City</td><td>City</td><td>none</td></tr><tr><td>Province</td><td>Province</td><td>none</td></tr><tr><td>Zip code</td><td>Zip_code</td><td>none</td></tr></table> The system uses a SQL update query to save the captured information in the **SUPPLIER DOCUMENT** table as follows: <table><tr><th>Details</th><th>Attribute in table</th><th>Notes</th></tr><tr><td>Supplier Document ID</td><td>Supplier_ document_ ID</td><td>ID is generated by the system by using the last ID in the table and incrementing it by one.</td></tr><tr><td>Supplier Document Name</td><td>Supplier_ Document_ Name</td><td>none</td></tr><tr><td>Supplier Document</td><td>Supplier_ document</td><td>None</td></tr></table> **[ALT]**
			Step 7: The system notifies Procurement and Logistics Officer that the Supplier was

<table>
<tr><td rowspan="2"></td><td></td><td></td><td>updated successfully, using an Angular Material Success Notification Modal with the following controls:
<table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Success</td><td>Header</td><td>none</td></tr><tr><td>The Supplier was successfully updated</td><td>Label</td><td>none</td></tr><tr><td>OK</td><td>button</td><td>None</td></tr></table></td></tr>
<tr><td>Step 8:
The Procurement and Logistics Officer clicks the “Ok” button.</td><td></td><td></td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 2:
The Procurement and Logistics Officer selects the “Cancel” button. The use case is terminated.</td></tr>
<tr><td></td><td colspan="3">ALT Step 3:
• The captured details are not valid. The system displays a notification indicating that the information received is incorrect or incomplete with an Angular Material Invalid Notification Modal as follows:
<table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Information Invalid</td><td>Header</td><td>none</td></tr><tr><td>The entered details were incorrect. Please fill in the correct information.</td><td>Label</td><td>none</td></tr><tr><td>OK</td><td>button</td><td>None</td></tr></table>The system returns to step 1.</td></tr>
<tr><td></td><td colspan="3">ALT Step 5:
The Procurement and Logistics Officer does not confirm that the information captured is correct by clicking the “No” button.
The system returns to step 1.</td></tr>
</table>

<table>
<tr><td></td><td>ALT Step 6:
The system does not successfully update the Supplier information to the SUPPLIER table. The system notifies the user that the Supplier has not been updated successfully with an Angular Material Error Modal as follows:

<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Error</td><td>Header</td><td>none</td></tr>
<tr><td>An error has occurred while saving the Supplier details. Please try again.</td><td>Label</td><td>none</td></tr>
<tr><td>OK</td><td>button</td><td>None</td></tr>
</table>
The system terminates this use case</td></tr>
<tr><td>CONCLUSION:</td><td>This use concludes when the Procurement and Logistics Officer has successfully updated the Supplier.</td></tr>
<tr><td>POST-CONDITION:</td><td>• The supplier has been updated in the SUPPLIER table</td></tr>
</table>

1.2.10. Inventory Management Subsystem

1.2.10.1 Add-Line Item

Gohvan Construction		
Author (s): **Amore Rossouw**	**Date: 28/06/2021**	**Version: 1.0**
USE CASE NAME:	**Add Line-item**	**USE CASE TYPE**
USE CASE ID:	**10.1**	**Abstract:** ☐
PRIORITY:	Medium	**Extension:** ☑
SOURCE:	Gohvan Construction System	
PRIMARY BUSINESS ACTOR:	The Procurement and Logistics Officer (PBA)	
PRIMARY THE SYSTEM ACTOR:	None	
OTHER PARTICIPATING ACTORS:	• None	
OTHER INTERESTED STAKEHOLDERS:	• None	
DESCRIPTION:	This use case narrative describes the process the Procurement and Logistics Officer will follow when they want to add a new line-item. The system will capture and validate the line-item information if the information was validated successfully the system will store it in the **LINE-ITEM** table. Only the Procurement and Logistics Officer must be able to add new line-items.	
PRE-CONDITION:	• Procurement and Logistics Officer must be logged into the system.	
TRIGGER:	The Procurement and Logistics Officer wants to add a new line-item.	
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**
		Manual Action / Automated Action
		Step 1: The system responds by loading the “Add Line-Item” Screen with Line-Item table information using the SQL Read query with the following controls:

Input Name	Control Type	Notes

			Add Line-Item	Header	none
			Line-Item Name	Label	none
			Line-Item Name	Input	none
			Line-Item Type	Label	none
			Select Line-Item Type	Select drop down list	The select options are populated with the line-item Type Description attribute of each in the Line-Item Type table. The value of each item in the drop down is set to the corresponding Line-item Type ID.
			Price	Label	none
			Price	Input	none
			Per	Label	none
			Per unit	Input	none
			Description	Label	none
			Description	Textbox	none
			Add Line-item	Button	none
			Cancel	Button	none
	Step 2: The Procurement and Logistics Officer fills in the necessary information into the appropriate input fields requested				

<table>
<tr><td></td><td>by the system.</td><td></td><td></td></tr>
<tr><td></td><td>Step 3:
The Procurement and Logistics Officer selects the “Add Line-Item” button. [ALT]</td><td></td><td>Step 4:
The system captures and validates the information supplied by the Procurement and Logistics Officer against the Line-Item table using Angular Validation:
[ALT]

<table>
<tr><th>Input Name</th><th>Attribute in table</th><th>Validation requirements</th></tr>
<tr><td>Line-Item Name</td><td>Line-Item_Name</td><td>required</td></tr>
<tr><td>Select Line-Item Type</td><td>Line-Item_type</td><td>required</td></tr>
<tr><td>Per unit</td><td>perUnit</td><td>required</td></tr>
<tr><td>Description</td><td>Description</td><td>required</td></tr>
</table>

The system captures and validates the information supplied by the Procurement and Logistics Officer against the Price table using Angular Validation:

<table>
<tr><th>Input Name</th><th>Attribute in table</th><th>Validation requirements</th></tr>
<tr><td>Price</td><td>Price</td><td>required</td></tr>
</table>
</td></tr>
<tr><td></td><td></td><td></td><td>Step 5:
The system displays an Angular Material Confirmation Modal, with the following controls, to prompt the Procurement and Logistics Officer for confirmation of the entered details:

<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Confirm information</td><td>Header</td><td>none</td></tr>
<tr><td>Please confirm that the details entered are correct?</td><td>Label</td><td>none</td></tr>
</table>
</td></tr>
</table>

<table>
<tr>
<td></td>
<td></td>
<td></td>
<td>
<table>
<tr><td>Yes</td><td>button</td><td>This button confirms that the entered details were correct</td></tr>
<tr><td>No</td><td>button</td><td>This button confirms that the entered details were not correct</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 6:
The Procurement and Logistics Officer confirms the captured information by clicking the “Yes” button.
[ALT]</td>
<td></td>
<td>
Step 7:
The system uses a SQL insert query to save the captured information in the LINE-ITEM table as follows:
<table>
<tr><th>Details</th><th>Attribute in table</th><th>Notes</th></tr>
<tr><td>Line-Item ID</td><td>Line-Item_ID</td><td>ID is generated by the system by using the last ID in the table and incrementing it by one.</td></tr>
<tr><td>Line-Item Name</td><td>Line-Item_Name</td><td>none</td></tr>
<tr><td>Select Line-Item Type</td><td>Line-Item_type</td><td>none</td></tr>
<tr><td>Price</td><td>PriceID</td><td>none</td></tr>
<tr><td>Per unit</td><td>perUnit</td><td>none</td></tr>
<tr><td>Description</td><td>Description</td><td>none</td></tr>
</table>
The system uses a SQL insert query to save the captured information in the PRICE table as follows:
</td>
</tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>Details</th><th>Attribute in table</th><th>Notes</th></tr>
<tr><td>Price ID</td><td>Price_ID</td><td>ID is generated by the system by using the last ID in the table and incrementing it by one.</td></tr>
<tr><td>Price</td><td>Price</td><td>none</td></tr>
<tr><td>Date</td><td>Date</td><td>The Date will correspond to the date the price was added.</td></tr>
</table>
[ALT]
</td></tr>
<tr><td></td><td></td><td></td><td>
Step 8:
The system notifies Procurement and Logistics Officer that the Line-Item was added successfully, using an Angular Material Success Notification Modal with the following controls:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Success</td><td>Header</td><td>none</td></tr>
<tr><td>The Line-Item was successfully created</td><td>Label</td><td>none</td></tr>
<tr><td>OK</td><td>button</td><td>None</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 9:
The Procurement and Logistics Officer clicks the “Ok” button.</td><td></td><td></td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 3:
The Procurement and Logistics Officer selects the “Cancel” button. The use case is terminated.</td></tr>
<tr><td></td><td colspan="3">ALT Step 4:</td></tr>
</table>

<table>
<tr><td></td><td>
The captured details are not valid. The system displays a notification indicating that the information received is incorrect or incomplete with an Angular Material Invalid Notification Modal as follows:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Information Invalid</td><td>Header</td><td>none</td></tr>
<tr><td>The entered details were incorrect. Please fill in the correct information.</td><td>Label</td><td>none</td></tr>
<tr><td>OK</td><td>button</td><td>None</td></tr>
</table>
The system returns to step 1.
</td></tr>
<tr><td></td><td>
ALT Step 6:
The Procurement and Logistics Officer does not confirm that the information captured is correct by clicking the “No” button.
The system returns to step 1.
</td></tr>
<tr><td></td><td>
ALT Step 7:
The system does not successfully add the Line-item information to the LINE-ITEM and LINE-ITEM EMPLOYEE table. The system notifies the user that the Line-item has not been added successfully with an Angular Material Error Modal as follows:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Error</td><td>Header</td><td>none</td></tr>
<tr><td>An error has occurred while saving the Line-item details. Please try again.</td><td>Label</td><td>none</td></tr>
<tr><td>OK</td><td>button</td><td>None</td></tr>
</table>
The system terminates this use case.
</td></tr>
<tr><td>CONCLUSION:</td><td>This use case concludes when the Procurement and Logistics Officer has added a new line-item successfully.</td></tr>
<tr><td>POST-CONDITION:</td><td>The line-item has been added to the LINE-ITEM table successfully.</td></tr>
</table>

1.2.10.2 Search-Line Item

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Amore Rossouw</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Search Line-item</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">10.2</td><td rowspan="3">Business Requirements: ☐
System Analysis: o
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Medium</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">The Procurement and Logistics Officer (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">• None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the process the Procurement and Logistics Officer will follow when they want to search for a line-item.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The User must be logged into the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The user wants to search for a line-item.</td></tr>
<tr><td>TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td>Step 1:
The User wants to search for an existing line-item. The user clicks on the “Inventory Management” option in the side menu bar on the home screen</td><td></td><td>Step 2:
The system responds by loading the “Inventory Management “screen with the following controls:

<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Line-items</td><td>Header</td><td>none</td></tr>
<tr><td>+</td><td>Button</td><td>This control is used to add a Line-item</td></tr>
<tr><td>Search Line-item</td><td>Label</td><td>none</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>Search Parameter</td><td>Input</td><td>none</td></tr>
<tr><td>Search</td><td>Button</td><td>none</td></tr>
<tr><td>Line-item Table</td><td>Table</td><td>This table is populated with the “line-item description” and “Type” attributes from the LINE-ITEM table and the “price” and “Date” attribute from the PRICE table. An “edit” button will be next to each Line-item record.</td></tr>
<tr><td>Edit Line-item</td><td>Button</td><td>This control is used to edit a Line-item</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 3:
The user clicks on the “Search Line-Item” input and enters the requested search parameter. Thereafter the user clicks on the “Search Line-Item” button.</td><td></td><td>Step 4:
The system captures and validates the information, supplied by the User, against the LINE-ITEM table using Angular Validation:

<table>
<tr><th>Input Name</th><th>Attribute in table</th><th>Validation requirements</th></tr>
<tr><td>Line-Item Name</td><td>Line-Item_Name</td><td>Required</td></tr>
</table>

[ALT]</td></tr>
<tr><td></td><td></td><td></td><td>Step 5:
A SQL read query is used to read the following attributes from the following tables based on the parameter provided:

The Table is searched based on the Line-item Name:

LINE-ITEM Table:
• Line-item_ID</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>

Line-item_Name
PriceID
perUnit
Description

The Table is searched based on the Price ID:

PRICE Table:

PriceID
Price
Date

The table attributes are read via an HttpGet method in the Typescript component and an API request.
</td></tr>
<tr><td></td><td></td><td></td><td>Step 6:
The system populates the ‘Line-Item Table’ with the data received in the previous step
[ALT]</td></tr>
<tr><td></td><td>Step 7:
The User selects the Line-item from the search result by clicking on the “Line-item Name” label.
[ALT]</td><td></td><td>Step 8:
The system responds by loading the “View Line-item” page with the following controls:

<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Line-Item Name</td><td>Header</td><td>This header’s text is dynamically changed to the selected Line-item’s name.</td></tr>
<tr><td>Line-Item Type:</td><td>Label</td><td>none</td></tr>
<tr><td>Line-Item Type</td><td>Label</td><td>This label’s text is dynamically changed to the selected Line-item’s type.</td></tr>
<tr><td>Price</td><td>Label</td><td>none</td></tr>
<tr><td>Price</td><td>Label</td><td>This label’s text is dynamically changed to the selected</td></tr>
</table>
</td></tr>
</table>

					Line-item's price.
			Per	Label	none
			Per unit	Label	This label's text is dynamically changed to the selected Line-item's per unit.
			Description	Label	none
			Description	Label	This label's text is dynamically changed to the selected Line-item's desription.

ALTERNATE COURSES:

ALT Step 4:

The captured search parameters are not valid. The system displays an Angular Material Invalid Notification modal indicating that the information received is incorrect or incomplete:

Input Name	Control Type	Notes
Information Invalid	Header	none
The entered parameters were incorrect. Please try again.	Label	none
OK	button	None

The system returns to step 2.

ALT Step 6:

The line-item that the user searched does not exist within the database. The system notifies the user that no search results were found by displaying an Angular Material Error modal with the following controls:

Input Name	Control Type	Notes
No Line-item found	Header	none
There was no Line-item found that matched your search request.	Label	none
OK	button	None

The system will return to Step 2.

	ALT Step 7 (A): The Procurement and Logistics Officer clicks on the “Add (+)” button
	ALT Step 7 (B): The Procurement and Logistics clicks on the “Edit” button for the Line-item they want to edit.
CONCLUSION:	This use case concludes when the system retrieves and displays the line-item search results.
POST-CONDITION:	• The searched line-item is retrieved and displayed.
BUSINESS RULES:	• Only the Procurement and Logistics Officer is allowed to search a line-item.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.10.3 Edit line-item

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Amore Rossouw</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Edit Line-Item</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">10.3</td><td rowspan="3">Abstract: ☐
Extension: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Medium</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">The Procurement and Logistics Officer (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">• None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case narrative describes the process the Procurement and Logistics Officer will follow when they want to update an existing line-item. The system will capture and validate the line-item information, if the information was validated successfully the system will update the line-item in the LINE-ITEM table. Only the Procurement and Logistics Officer is allowed to update an existing line-item.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The user must be logged into the system.
• The line-item that the user wishes to update must exist in the database.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The Procurement and Logistics Officer wants to update an existing line-item.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td></td><td colspan="2">Step 1:
The system responds by loading the “Edit Line-Item” screen with the following controls:

<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Edit Line-Item</td><td>Header</td><td>none</td></tr>
</table>
</td></tr>
</table>

				Line-Item Name	Label	none
				Line-Item Name	Input	This control is Populated with the current line-item_Name attribute that was obtained in the use case ”10.2 search Line-item ”.
				Line-Item Type	Label	none
				Select Line-Item Type	Select drop down list	This control’ default value is set to the current line-item_type attribute that was obtained in the use case ”10.2 search Line-item ”.The default value should be set to the current Line-item-Type ID. The select options are populated with the line-item Type Description attribute of each in the Line-Item Type table. The value of each item in the drop down is set to the corresponding Line-item Type ID.
				Price	Label	none

			<table><tr><td>Price</td><td>Input</td><td>This control is Populated with the current line-item_Price attribute that was obtained in the use case ”10.2 search Line-item ”.</td></tr><tr><td>Per</td><td>Label</td><td>none</td></tr><tr><td>Per unit</td><td>Input</td><td>This control is Populated with the current line-item_perUnit attribute that was obtained in the use case ”10.2 search Line-item ”.</td></tr><tr><td>Description</td><td>Label</td><td>none</td></tr><tr><td>Description</td><td>Textbox</td><td>This control is Populated with the current line-item_ Description attribute that was obtained in the use case ”10.2 search Line-item ”.</td></tr><tr><td>Edit Line-Item</td><td>Button</td><td>none</td></tr></table>
	Step 2: The Procurement and Logistics Officer fills in the necessary information		**Step 3**: The system captures and validates the information supplied by the Procurement and Logistics Officer against the Line-Item table using Angular Validation: **[ALT]**

into the appropriate input fields requested by the system.

The Procurement and Logistics Officer selects the "Edit Line-Item" button. **[ALT]**

Input Name	Attribute in table	Validation requirements
Line-Item Name	Line-Item_Name	required
Select Line-Item Type	Line-Item_type	required
Per unit	perUnit	required
Description	Description	required

The system captures and validates the information supplied by the Procurement and Logistics Officer against the Price table using Angular Validation:

(The price table is only updated if the user changed the price of the line-item)

Input Name	Attribute in table	Validation requirements
Price	Price	required

Step 4:

The system displays an Angular Material Confirmation Modal, with the following controls, to prompt the Procurement and Logistics Officer for confirmation of the entered details:

Input Name	Control Type	Notes
Confirm information	Header	none
Please confirm that the details entered are correct?	Label	none
Yes	button	This button confirms that the entered details were correct
No	button	This button confirms that the

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td></td><td></td><td>entered details were not correct</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 5:
The Procurement and Logistics Officer confirms the captured information by clicking the “Yes” button.
[ALT]</td><td></td><td>
Step 6:
The system uses a SQL insert query to save the captured information in the LINE-ITEM table as follows:
<table>
<tr><th>Details</th><th>Attribute in table</th><th>Notes</th></tr>
<tr><td>Line-Item ID</td><td>Line-Item_ID</td><td>ID is generated by the system by using the last ID in the table and incrementing it by one.</td></tr>
<tr><td>Line-Item Name</td><td>Line-Item_Name</td><td>none</td></tr>
<tr><td>Select Line-Item Type</td><td>Line-Item_type</td><td>none</td></tr>
<tr><td>Price</td><td>PriceID</td><td>none</td></tr>
<tr><td>Per unit</td><td>perUnit</td><td>none</td></tr>
<tr><td>Description</td><td>Description</td><td>none</td></tr>
</table>
The system uses a SQL insert query to save the captured information in the PRICE table as follows:

(The price table is only updated if the user changed the price of the line-item)
<table>
<tr><th>Details</th><th>Attribute in table</th><th>Notes</th></tr>
<tr><td>Price ID</td><td>Price_ID</td><td>ID is generated by the system by using the last ID in the table and</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td></td><td></td><td>incrementing it by one.</td></tr>
<tr><td>Price</td><td>Price</td><td>none</td></tr>
<tr><td>Date</td><td>Date</td><td>The Date will correspond to the date the price was added.</td></tr>
</table>
[ALT]
</td></tr>
<tr><td></td><td></td><td></td><td>
Step 7:
The system notifies Procurement and Logistics Officer that the Line-Item was added successfully, using an Angular Material Success Notification Modal with the following controls:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Success</td><td>Header</td><td>none</td></tr>
<tr><td>The Line-Item was successfully created</td><td>Label</td><td>none</td></tr>
<tr><td>OK</td><td>button</td><td>None</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 8:
The Procurement and Logistics Officer clicks the “Ok” button.</td><td></td><td></td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 2:
The Procurement and Logistics Officer selects the “Cancel” button. The use case is terminated.</td></tr>
<tr><td></td><td colspan="3">ALT Step 3:
• The captured details are not valid. The system displays a notification indicating that the information received is incorrect or incomplete with an Angular Material Invalid Notification Modal as follows:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Information Invalid</td><td>Header</td><td>none</td></tr>
</table>
</td></tr>
</table>

	<table><tr><td>The entered details were incorrect. Please fill in the correct information.</td><td>Label</td><td>none</td></tr><tr><td>OK</td><td>button</td><td>None</td></tr></table> The system returns to step 3.
	ALT Step 5: The Procurement and Logistics Officer does not confirm that the information captured is correct by clicking the “No” button. The system returns to step 3.
	ALT Step 6: The system does not successfully update the Line-item information to the LINE-ITEM table. The system notifies the user that the Line-item has not been updated successfully with an Angular Material Error Modal as follows: <table><tr><td>Input Name</td><td>Control Type</td><td>Notes</td></tr><tr><td>Error</td><td>Header</td><td>none</td></tr><tr><td>An error has occurred while saving the Line-item details. Please try again.</td><td>Label</td><td>none</td></tr><tr><td>OK</td><td>button</td><td>None</td></tr></table> The system terminates this use case
CONCLUSION:	This use case concludes when the Procurement and Logistics Officer has successfully updated the line-item.
POST-CONDITION:	• The line-item has been updated in the **LINE-ITEM** table

1.2.11. Reporting Subsystem

1.2.11.1 Generate Site Progress Report

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Vincent Yu</td><td colspan="2">Date: 28/06/2021</td><td>Version: 3.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Generate Site Progress Report</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">11.1</td><td rowspan="3">Business Requirements: □
System Analysis: □
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Medium</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Operations Manager (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">Managing Director</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the event when the Operations Manager wants to generate a project progress report. This report will include project details of the site, the sites current status, and the construction in progress log. These details will be consolidated and sent to the user on request.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The Operations Manager needs to be logged in the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The Operations Manager wishes to generate a site progress report.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td>Step 1:
The Operations Manager clicks on the “Menu” to navigate to the click on the</td><td></td><td>Step 2:
The system responds by loading the “Generate Site Progress Report” screen. The “Generate Site Progress Report” screen has the following controls:</td></tr>
</table>

<table>
<tr>
<td></td>
<td>Reporting tab and then clicking “Generate Site Progress Report”</td>
<td></td>
<td>
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Generate Progress Type Report</td><td>Heading</td><td>None</td></tr>
<tr><td>Site Type</td><td>Dropdown</td><td>Control populated with the site types</td></tr>
<tr><td>Start Date</td><td>Label and Date Picker</td><td>Used to capture start date of report</td></tr>
<tr><td>End Date</td><td>Label and Date Picker</td><td>Used to capture end date of report</td></tr>
<tr><td>Generate</td><td>Button</td><td>Used when the Operations Manager wishes to generate the report.</td></tr>
<tr><td>Back</td><td>Button</td><td>Used to navigate back.</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 3:
The Operations Manager selects the desired site type using the dropdown control and selecting the start and end dates. And then clicks the Generate button.</td>
<td></td>
<td>Step 4:
The system uses the SQL Read query to read and aggregate information from the
<table>
<tr><th>Table</th><th>Attribute</th></tr>
<tr><td>Site Type</td><td>Site Type name</td></tr>
<tr><td></td><td>Site Type Description</td></tr>
</table>
<table>
<tr><th>Table</th><th>Attribute</th></tr>
<tr><td>Project</td><td>Project name</td></tr>
<tr><td></td><td>Site location</td></tr>
<tr><td></td><td>Date started</td></tr>
<tr><td></td><td>Project Status</td></tr>
</table>
<table>
<tr><th>Table</th><th>Attribute</th></tr>
<tr><td>Project Status</td><td>Status Name</td></tr>
<tr><td></td><td>Status Description</td></tr>
</table>
<table>
<tr><th>Table</th><th>Attribute</th></tr>
<tr><td></td><td>Company Name</td></tr>
</table>
</td>
</tr>
</table>

			Company Information: Phone Number, Email Address, Address **Table**: Site_Progress_description — **Attribute**: Title, Description, Date Uploaded **Table**: Construction_Photo — **Attribute**: Construction_Photo **Table**: Completion_Photo — **Attribute**: Completion_Photo, Date Uploaded All these attributes are read from their respective table and the report is generate from the information read from these tables. The report generated will be for the information within the specific date range that was entered. **[ALT]**
			Step 5: The system generates the Site Progress Report with the following controls: **Control Name** / **Control Type** / **Notes** **Date Generated:** / **Label** / None **Date Generated** / **Label** / Control filled with the date that the report was generated Generated By: / Label / None Generated By / Label / Control filled with the name of the Operations Manager that

					generated the report.
			Company Name:	**Label**	None
			Company Name	**Label**	Control filled with the name of the company.
			Company Phone Number:	**Label**	none
			Company Phone Number	**Label**	Control filled with the contact of the company.
			Company Email:	**Label**	none
			Company Email	**Label**	Control filled with the email of the company.
			Company Address:	**Label**	none
			Company Address	**Label**	Control filled with the address of the company
			Gohvan Construction Logo	**Picture**	Picture of the Gohvan Construction logo
			Site Progress Report	**Heading**	None
			Duration:	**Label**	None
			Start Date	**Label**	Control filled with the start date of the filter period. dd/mm/yyyy
			End Date	**Label**	Control filled with the end date of the filter period. dd/mm/yyyy
			Site Type:	**Label**	None
			Site type	**Label**	Control filled with the site type name.
			Report Description:	**Heading**	none
			Description	**Label**	Description of the report.

			Total Number of Sites (In this duration):	**Label**	Label Pre-filled with the total number of sites in the provided duration.
			Number of completed Sites:	**Label**	none
			Number of completed Sites	**Label**	Control populated with the number of completed sites.
			Number of on-going Sites:	**Label**	none
			Number of on-going Sites	**Label**	Control populated with the number of on-going sites.
			Site Progress Bar graph	**Chart / Bar Graph**	Bar graph showing the consolidated information
			The below controls are dynamically generated based on the number of records/sites retrieved.		
			Site Name	**Label**	none
			Date started:	**Label**	none
			Date started	**Label**	
			Address:	**Label**	none
			Address	**Label**	
			Status:	**Label**	none
			Status	**Label**	
			Area:	**Label**	none
			Area	**Label**	
			Site Size:	**Label**	none
			Site Size	**Label**	
			Site Progress	**Group Box**	none

			Description		
			Title:	**Label**	none
			Title	**Label**	
			Description:	**Label**	none
			Description	**Label**	
			Date Uploaded:	**Label**	none
			Date Uploaded	**Label**	
			The below group box is only generated if the site status = “Completed”		
			Completion Photos	**Group Box**	none
			Completion Photos:	**Label**	none
			Completion Photo	**Image**	This control is generated for each completion photo.
			Date Uploaded:	**Label**	none
			Date Uploaded	**Label**	Control populated with

Controls on screen:

Control Name	Control Type	Notes
Download	**Button**	Used to download the generated report

	Step 6: The Operations Manager clicks the “Download” button.		**Step 7:** The system will download the report and save it in the Downloads folder on the operations manager’s Computer/Laptop
ALTERNATE COURSES:	**ALT Step 4:** The selected a date range that is out of bounds and no information was retrieved from the respective tables. The system displays a Dialog with the following controls:		

	<table><tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Invalid Date Range</td><td>Heading</td><td>None</td></tr><tr><td>The entered dates are invalid</td><td>Label</td><td>Used to inform the operations manager about the error</td></tr><tr><td>OK</td><td>Button</td><td>None</td></tr></table> **ALT Step 5:** The operations manager clicks the “OK” button **Return to Step 2**
CONCLUSION:	This use case concludes when the report is sent to the Operations Manager.
POST-CONDITION:	• The Project progress report is sent to the user.
BUSINESS RULES:	• The system can only generate reports if there are project and site details on the system. • Site progress report generated upon request. Does not generate automatically.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None
ASSUMPTIONS:	None
OPEN ISSUES:	None

1.2.11.2 Generate Site Type Report

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Vincent Yu</td><td colspan="2">Date: 28/06/2021</td><td>Version: 3.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Generate Site Type Report</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">11.2</td><td rowspan="3">Business Requirements: □
System Analysis: □
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Low</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Operations Manager (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• Managing Director</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the event when the operations manager wants to generate a site type report. This report will include the different types of sites with the number of completed sites and the sites that are still in progress / construction. These details will be consolidated and sent to the user on request.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">The Operations Manager needs to be logged in the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The Operations Manager wants to generate a site type report.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td>Step 1:
The User clicks on the “Menu” to navigate to the click on the Reporting tab and then clicking “Generate</td><td></td><td>Step 2:
The system responds by loading the “Generate Site Type Report” screen. The “Generate Site Type Report” screen has the following controls:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Generate Site</td><td>Heading</td><td>None</td></tr>
</table></td></tr>
</table>

<table>
<tr>
<td></td>
<td>Site Type Report”</td>
<td></td>
<td>
<table>
<tr><td>Type Report</td><td></td><td></td></tr>
<tr><td>Start Date</td><td>Label and Date Picker</td><td>Used to capture start date of report</td></tr>
<tr><td>End Date</td><td>Label and Date Picker</td><td>Used to capture end date of report</td></tr>
<tr><td>Generate</td><td>Button</td><td>Used when the Operations Manager wishes to generate the report.</td></tr>
<tr><td>Back</td><td>Button</td><td>Used to navigate back.</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 3:
The Operations Manager selects the desired site type using the dropdown control and selecting the start and end dates. And then clicks the Generate button.</td>
<td></td>
<td>Step 4:
The system uses the SQL Read query to read and aggregate information from the
<table>
<tr><td>Table</td><td>Attribute</td></tr>
<tr><td rowspan="2">TSS</td><td>Site Type ID</td></tr>
<tr><td>Project ID</td></tr>
</table>
<table>
<tr><td>Table</td><td>Attribute</td></tr>
<tr><td rowspan="3">Project</td><td>Project name</td></tr>
<tr><td>Date started</td></tr>
<tr><td>Project Status</td></tr>
</table>
<table>
<tr><td>Table</td><td>Attribute</td></tr>
<tr><td rowspan="2">Project Status</td><td>Status Name</td></tr>
<tr><td>Status Description</td></tr>
</table>
<table>
<tr><td>Table</td><td>Attribute</td></tr>
<tr><td rowspan="2">Site Type</td><td>Site Type name</td></tr>
<tr><td>Site Type Description</td></tr>
</table>
<table>
<tr><td>Table</td><td>Attribute</td></tr>
<tr><td rowspan="4">Company Information</td><td>Company Name</td></tr>
<tr><td>Phone Number</td></tr>
<tr><td>Email Address</td></tr>
<tr><td>Address</td></tr>
</table>
</td>
</tr>
</table>

All these attributes are read from their respective table and the report is generate from the information read from these tables. The report generated will be for the information within the specific date range that was entered. **[ALT]**

Step 5:

The system generates the Site Type Report with the following controls:

Control Name	Control Type	Notes
Date Generated:	**Label**	None
Date Generated	**Label**	Control filled with the date that the report was generated
Generated By:	Label	None
Generated By	Label	Control filled with the name of the Operations Manager that generated the report.
Company Name:	**Label**	None
Company Name	**Label**	Control filled with the name of the company.
Company Phone Number:	**Label**	none
Company Phone Number	**Label**	Control filled with the contact of the company.
Company Email:	**Label**	none
Company Email	**Label**	Control filled with the email of the company.
Company Address:	**Label**	
Company Address	**Label**	Control filled with the address of the company

			Gohvan Construction Logo	**Picture**	Picture of the Gohvan Construction logo
			Generate Site Type Report	**Heading**	None
			Duration:	**Label**	None
			Start Date	**Label**	Control filled with the start date of the filter period. dd/mm/yyyy
			End Date	**Label**	Control filled with the end date of the filter period. dd/mm/yyyy
			Report Description:	**Heading**	
			Description	**Label**	Description of the report.
			Total Number of Sites (In this duration):	**Label**	Label Pre-filled with the total number of sites in the provided duration.
			Number of completed Sites:	**Label**	none
			Number of completed Sites:	**Label**	Control populated with the number of completed sites
			Number of on-going Sites:	**Label**	none
			Number of on-going Sites:	**Label**	Control populated with the number of on-going sites
			Site Type Bar graph	**Chart / Bar Graph**	Bar graph showing the consolidated information
			Controls on screen:		

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Download</td><td>Button</td><td>Used to download the generated report</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 6:
The Operations Manager clicks the “Download” button.</td><td></td><td>Step 7:
The system will download the report and save it in the Downloads folder on the operations manager’s Computer/Laptop</td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 4:
The selected a date range that is out of bounds and no information was retrieved from the respective tables. The system displays a Dialog with the following controls:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Invalid Date Range</td><td>Heading</td><td>None</td></tr>
<tr><td>The entered dates are invalid</td><td>Label</td><td>Used to inform the operations manager about the error</td></tr>
<tr><td>OK</td><td>Button</td><td>None</td></tr>
</table>
ALT Step 5:
The operations manager clicks the “OK” button
Return to Step 2</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">The use case concludes once The Operations Manager receives the report from the system.</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">• The Site Type Report was generated and sent to the operations manager.</td></tr>
<tr><td>BUSINESS RULES:</td><td colspan="3">• Only the Operations Manager has access to generating site type reports.
• The report can only be generated upon request. Not automatically.</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td colspan="3">None</td></tr>
<tr><td>ASSUMPTIONS:</td><td colspan="3">None</td></tr>
<tr><td>OPEN ISSUES:</td><td colspan="3">None</td></tr>
</table>

1.2.11.3 Generate Team Site Completed Report

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Vincent Yu</td><td colspan="2">Date: 28/06/2021</td><td>Version: 3.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Generate Team Site Completed Report</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">11.3</td><td rowspan="3">Business Requirements: □
System Analysis: □
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Medium</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Operations manager (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the event when a user wants to generate a team site completed report. This report will include the details of the construction teams and the projects they completed within a date range provided by the user. These details will be consolidated and sent to the user on request.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The Operations manager needs to be logged in the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The operations manager wishes to generate a team site completed report.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">The user wishes to generate a team site completed report.</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td>Step 1:
The Operations manager clicks on the “Menu” to navigate to the click on the Reporting tab</td><td></td><td>Step 2:
The system responds by loading the “Generate Team Site Completed Report” screen. The “Generate Team Site Completed Report” screen has the following controls:</td></tr>
</table>

<table>
<tr>
<td></td>
<td>and then clicking “Generate Team Site Completed Report”</td>
<td></td>
<td>
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Generate Team Site Completed Report</td><td>Heading</td><td>None</td></tr>
<tr><td>Team Name</td><td>Dropdown</td><td>Control populated with the team names</td></tr>
<tr><td>Start Date</td><td>Label and Date Picker</td><td>Used to capture start date of report</td></tr>
<tr><td>End Date</td><td>Label and Date Picker</td><td>Used to capture end date of report</td></tr>
<tr><td>Generate</td><td>Button</td><td>Used when the Operations Manager wishes to generate the report.</td></tr>
<tr><td>Back</td><td>Button</td><td>Used to navigate back.</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 3:
The Operations Manager selects the desired site type using the dropdown control and selecting the start and end dates. And then clicks the Generate button.</td>
<td></td>
<td>Step 4:
The system uses the SQL Read query to read and aggregate information from the
<table>
<tr><th>Table</th><th>Attribute</th></tr>
<tr><td>TSS</td><td>Site Type ID</td></tr>
<tr><td></td><td>Project ID</td></tr>
</table>
<table>
<tr><th>Table</th><th>Attribute</th></tr>
<tr><td>Site Type</td><td>Site Type ID</td></tr>
<tr><td></td><td>Site Type name</td></tr>
<tr><td></td><td>Site Type Description</td></tr>
</table>
<table>
<tr><th>Table</th><th>Attribute</th></tr>
<tr><td>Project</td><td>Project name</td></tr>
<tr><td></td><td>Site location</td></tr>
<tr><td></td><td>Date started</td></tr>
<tr><td></td><td>Project Status</td></tr>
</table>
</td>
</tr>
</table>

Table	Attribute
Project Status	Status Name
	Status Description

Table	Attribute
Company Information	Company Name
	Phone Number
	Email Address
	Address

Table	Attribute
Team	Team_ID
	Team_Name
	Site Manager Employee ID

Table	Attribute
Employee_Team	Employee ID
	Team_ID

Table	Attribute
Employee	Employee ID
	Employee Type ID
	Emp_Name
	Emp_Surname

Table	Attribute
Employee_Type	EmpType_ID
	EmpType_Description

All these attributes are read from their respective table and the report is generate from the information read from these tables. The report generated will be for the information within the specific date range that was entered. **[ALT]**

Step 5:

The system generates the Site Type Report with the following controls:

Control Name	Control Type	Notes
Date Generated:	**Label**	None
Date Generated	**Label**	Control filled with the date that the report was generated
Generated By:	Label	None
Generated By	Label	Control filled with the name of the Operations Manager that generated the report.
Company Name:	**Label**	None
Company Name	**Label**	Control filled with the name of the company.
Company Phone Number:	**Label**	none
Company Phone Number	**Label**	Control filled with the contact of the company.
Company Email:	**Label**	none
Company Email	**Label**	Control filled with the email of the company.
Company Address:	**Label**	none
Company Address	**Label**	Control filled with the address of the company
Gohvan Construction Logo	**Picture**	Picture of the Gohvan Construction logo
Team Site Completed Report	**Heading**	None

			Duration:	**Label**	None
			Start Date	**Label**	Control filled with the start date of the filter period. dd/mm/yyyy
			End Date	**Label**	Control filled with the end date of the filter period. dd/mm/yyyy
			Site Type:	**Label**	None
			Site type	**Label**	Control filled with the site type name.
			Team Name:	**Label**	None
			Team Name	**Label**	Control filled with the Team name.
			Report Description:	**Heading**	None
			Description	**Label**	Description of the report.
			Site Manager:	**Label**	None
			Site Manager	**Label**	Populated with the Team's Site manager's Name.
			Total Number of Sites (In this duration):	**Label**	Label Pre-filled with the total number of sites in the provided duration.
			Number of completed Sites:	**Label**	none
			Number of completed Sites:	**Label**	Control populated with the number of completed sites
			Number of on-going Sites:	**Label**	none
			Number of on-going Sites:	**Label**	Control populated with the number of on-going sites

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>Sites Bar graph</td><td>Chart / Bar Graph</td><td>Bar graph showing the consolidated information</td></tr>
</table>
Controls on screen:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Download</td><td>Button</td><td>Used to download the generated report</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 6:
The Operations Manager clicks the “Download” button.</td><td></td><td>Step 7:
The system will download the report and save it in the Downloads folder on the operations manager’s Computer/Laptop</td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 4:
The selected a date range that is out of bounds and no information was retrieved from the respective tables. The system displays a Dialog with the following controls:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Invalid Date Range</td><td>Heading</td><td>None</td></tr>
<tr><td>The entered dates are invalid</td><td>Label</td><td>Used to inform the operations manager about the error</td></tr>
<tr><td>OK</td><td>Button</td><td>None</td></tr>
</table>
ALT Step 5:
The operations manager clicks the “OK” button
Return to Step 2</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">This use case concludes when the report is sent to the User.</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">• The Team Site Completed report is sent to the user.
•</td></tr>
<tr><td>BUSINESS RULES:</td><td colspan="3">• Only the operations manager has access to generate team site completed reports.
• The system can only generate reports if there are project and site details on the system.</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND</td><td colspan="3">None</td></tr>
</table>

SPECIFICATIONS:	
ASSUMPTIONS:	None
OPEN ISSUES:	None

1.2.11.4 Generate Employee Document Expiry Report

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s): Vincent Yu</td><td colspan="2">Date: 28/06/2021</td><td>Version: 3.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Generate Employee Document Expiry Report</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">11.4</td><td rowspan="3">Business Requirements: □
System Analysis: □
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Low</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">Time (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">• Human Resource Officer (ERA)</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">Managing Director</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the event when the system automatically generates an employee document expiry report. This report will include employee details, the document details, and the expiry date of the documents. These details will be consolidated and sent to the Human Resource (HR) Officer monthly.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The system must have been in operation for over one month.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">One month has passed and the human resource Officer is expecting an Employee Document Expiry Report.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1:
A month passes, and the Human Resource Officer is expecting an Employee Document Expiry report.</td><td></td><td>Step 2:
The system is prompted to generate the Employee Document Expiry report.</td></tr>
</table>

Step 3:

The system uses the SQL Read query to read and aggregate information from the

Table	Attribute
Employee	Employee ID
	Emp_Name
	Emp_Surname
	Emp_IDNumber
	Emp_EmailAddress

Table	Attribute
Employee_compliance_Document	EmpCompDoc_ID
	EmpCompDoc_Title
	EmpCompDoc_ExpiryDate

All these attributes are read from their respective table and the report is generate from the information read from these tables. The report generated will be for the information within the specific date range that was entered.

Step 4:

The system generates the Site Type Report with the following controls:

Control Name	Control Type	Notes
Date Generated:	Label	None
Date Generated	Label	Control filled with the date that the report was generated
Generated By:	Label	None
Generated By	Label	Control filled with the name of the Operations Manager that generated the report.

			Company Name:	**Label**	None
			Company Name	**Label**	Control filled with the name of the company.
			Company Phone Number:	**Label**	none
			Company Phone Number	**Label**	Control filled with the contact of the company.
			Company Email:	**Label**	none
			Company Email	**Label**	Control filled with the email of the company.
			Company Address:	**Label**	
			Company Address	**Label**	Control filled with the address of the company
			Gohvan Construction Logo	**Picture**	Picture of the Gohvan Construction logo
			Generate Site Type Report	**Heading**	None
			Duration:	**Label**	None
			Start Date	**Label**	Control filled with the start date of the filter period. dd/mm/yyyy
			End Date	**Label**	Control filled with the end date of the filter period. dd/mm/yyyy
			Report Description:	**Heading**	

			Description | **Label** | Description of the report. **Total Number of Sites (In this duration):** | **Label** | Label Pre-filled with the total number of sites in the provided duration. **Number of completed Sites:** | **Label** | none **Number of completed Sites:** | **Label** | Control populated with the number of completed sites **Number of on-going Sites:** | **Label** | none **Number of on-going Sites:** | **Label** | Control populated with the number of on-going sites **Site Type Bar graph** | **Chart / Bar Graph** | Bar graph showing the consolidated information Controls on screen: **Control Name** | **Control Type** | **Notes** **Download** | **Button** | Used to download the generated report
			Step 5: The system reads the Human Resource Officer Information from the User table. The User_Email is read and can be used to send the report to the Human Resource Officer.
			Step 6: The system proceeds to send the generated report to the Human Resource Officer.
	Step 7:		

	The Human Resource Officer receives the generated report from the system.		
ALTERNATE COURSES:	None		
CONCLUSION:	This use case concludes when the report is sent to the Human Resource Officer.		
POST-CONDITION:	• The Employee Document Expiry report is sent to the Human Resource Officer.		
BUSINESS RULES:	• The system can only generate reports if there are employee and employee compliance document details on the system.		
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None		
ASSUMPTIONS:	None		
OPEN ISSUES:	None		

1.2.11.5 Generate Site Acceptance Document

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td colspan="2">Author (s): Vincent Yu</td><td colspan="2">Date: 28/06/2021</td><td>Version: 3.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Generate Site Acceptance Document</td><td colspan="2">USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">11.5</td><td colspan="2" rowspan="3">Business Requirements: □
System Analysis: □
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="4">Operations Manager (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="4">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="4">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="4">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="4">This use case describes the process the system will go through once the site visit for the quality control officer has been requested.
The system will the generate the Site acceptance document. The site acceptance document is a combination of site details and quotation details. The system will generate the basic layout of the acceptance document that can ben later edit in case there are any errors that are picked up.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="4">• The Site visit has been requested.</td></tr>
<tr><td>TRIGGER:</td><td colspan="4">The Operations Manager requests to generate a site acceptance document.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1:
The Operations manager clicks on the “Yes” button on the Project</td><td></td><td>Step 2:
The system uses the SQL Read query to
read and aggregate information from the
Table | Attribute</td></tr>
</table>

	Screen >> Request Site modal. To generate a site acceptance document		
			Step 5: The system generates and displays the Site Acceptance Document Screen with the following controls:

TSS	Site Type
	BS Number
	Cable Length
	Nav Light
	Container Size
	Mast Type
	Container Type

Table	Attribute
Construction Site	
	Breaker Size
	Signs Installed
	Aircon size
	Manhole Installed

Table	Attribute
User	User ID
	Emp_Name
	Emp_Surname
	Emp_EmailAddress

All these attributes are read from their respective table and the report is generate from the information read from these tables. The report generated will be for the information within the specific date range that was entered.

Control Name	Control Type	Notes
Date Generated:	Label	None
Date Generated	Label	Control filled with the date that the report was generated

			Generated By:	Label	None
			Generated By	Label	Control filled with the name of the Operations Manager that generated the report.
			Gohvan Construction Logo	**Picture**	Picture of the Gohvan Construction logo
			Site Acceptance Document	**Heading**	None
			Site Name:	**Label**	None
			Site Name	**Label**	Populated with the Site Name
			Site Details	**Label**	None
			Site Type:	**Label**	None
			Site type	**Label**	Control filled with the site type name.
			BS Number	**Label**	None
			BS Number	**Label**	Populated with the BS Number
			Cable Length	**Label**	None
			Cable Length	**Label**	Populated with the Length of Cable
			Nav Light	**Label**	None
			Nav Light	**Label**	Populated with the Nav Light
			Container Size	**Label**	None
			Container Size	**Label**	Populated with the Size of Container
			Mast Type	**Label**	None

			Mast Type	**Label**	Populated with the Type of Mast
			Container Type	**Label**	None
			Container Type	**Label**	Populated with the Type of Container
			Breaker Size	**Label**	None
			Breaker Size	**Label**	Populated with the Breaker Size
			Signs Installed	**Label**	None
			Signs Installed	**Label**	Populated with yes/no based on if the signs were installed or not.
			Aircon size	**Label**	None
			Aircon size	**Label**	Populated with the Aircon size
			Manhole Installed	**Label**	None
			Manhole Installed	**Label**	Populated with yes/no based on whether the manholes were installed or not.
			Emp_Name	**Label**	Populated with the Operations Managers Name
			Emp_Surname	**Label**	Populated with the Operations Managers Surname
			Emp_EmailAddress	**Label**	Populated with the Operations Managers Email
			Snag Table: Table used to contain snag list, with the following columns		
			Snag Item	**Column**	None
			Description	**Column**	None
			Note	**column**	None
			Controls on screen:		

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Confirm</td><td>Button</td><td>Used to download the generated report</td></tr>
<tr><td>Cancel</td><td>Button</td><td>Used to navigate back</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 4:
The Operations manager clicks on the “Confirm” button on the Generate Site Acceptance Document Screen
[ALT]</td><td></td><td>Step 5:
The system will create a new entry in the Acceptance_document table.
This will generate a new Acceptance_Document_ID for the uploaded acceptance document.
<table>
<tr><th>Table</th><th>Attribute</th></tr>
<tr><td>Acceptance_Document</td><td>Acceptance_Document_ID</td></tr>
</table>
</td></tr>
<tr><td></td><td></td><td></td><td>Step 6:
The system displays a Confirmation Alert with the following controls:
<table>
<tr><th>Control Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Confirmation Alert</td><td>Heading</td><td>None</td></tr>
<tr><td>Acceptance Document Created</td><td>Label</td><td>Used is inform the operations manager the acceptance document was created</td></tr>
<tr><td>OK</td><td>Button</td><td>None</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 7:
The Operations manager clicks on the “OK” button on the Confirmation Alert.</td><td></td><td>Step 8:
The system returns to use case 4.16 Request Site Visit >> Step 8.</td></tr>
</table>

ALTERNATE COURSES:	**ALT Step 4:** The operations manager clicks the “Cancel” Button **ALT Step 5:** The system displays an “Site Acceptance Document not saved” alert. **Terminate Use Case**
CONCLUSION:	This use case is concluded when the Site Acceptance document is displayed to the Operations Manager.
POST-CONDITION:	• The Acceptance Document is uploaded to the system.
BUSINESS RULES:	• Only the Operations Manager or Quality Control Officer can Generate the site acceptance document. • The Site Acceptance document must be created for a site visit to take place.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None
ASSUMPTIONS:	None
OPEN ISSUES:	None

1.2.12. User Management Subsystem

1.2.12.1 Login

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author(s):
Matthew Veltman</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Login</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">12.1</td><td rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">User (PBA) {Internal employees of Gohvan Construction}</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the process whereby a user of the system would like to log into the system. The use case begins when the person enters their login details. If the users' login details are correct and match the details in the database, the use case will conclude by allowing the user to login to the system.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">The person must have been registered onto the system by the Human Resources manager.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">User wants to log in to the system</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The user clicks on the "Login" button found on the "Login" page.</td><td></td><td>Step 2: The system displays the "Login" card with the following elements:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Login</td><td>Heading</td><td>None</td></tr>
<tr><td>Email</td><td>Label</td><td>None</td></tr>
<tr><td>Email</td><td>Text Box</td><td>None</td></tr>
</table></td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>Password</td><td>Label</td><td>None</td></tr>
<tr><td>Password</td><td>Text Box</td><td>Characters inputted by user are masked with a visible star (*).</td></tr>
<tr><td>Sign in</td><td>Button</td><td>None</td></tr>
<tr><td>Forgot password</td><td>Link</td><td>Invokes Use Case 12.3 “Reset Password”</td></tr>
</table>
</td></tr>
<tr><td rowspan="6"></td><td>Step 3: The user enters their email and password to log into the system. (ALT)</td><td></td><td>Step 4: The system captures and validates the information, entered by the user, with the information in the User Table. (ALT)</td></tr>
<tr><td></td><td></td><td>Step 5: A SQL query is used to read the email attribute from the User table. (ALT)</td></tr>
<tr><td></td><td></td><td>Step 6: The system then uses a hashing algorithm on the entered password to the password that corresponds with the entered email from the User table. (ALT)</td></tr>
<tr><td></td><td></td><td>Step 7: A SQL query is used to read the user's role of the business from the employee table.</td></tr>
<tr><td></td><td></td><td>Step 8: The system logs the user into the system.</td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 3: The user cancels the login. Return to step 1.</td></tr>
<tr><td></td><td colspan="3">ALT Step 4: The user inputted information is incorrect Return to step 3.</td></tr>
<tr><td></td><td colspan="3">ALT Step 5: The user inputted information has no username that correlates within the database. Return to step 3.</td></tr>
<tr><td></td><td colspan="3">ALT Step 6: The user inputted information has no username that correlates within the database. Return to step 3.</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">System logs the person into the system.</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">The person may access the areas of the system available to them.</td></tr>
<tr><td>BUSINESS RULES:</td><td colspan="3">The user must be registered through the Human Resources manager</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td colspan="3">none</td></tr>
<tr><td>ASSUMPTIONS:</td><td colspan="3">none</td></tr>
<tr><td>OPEN ISSUES:</td><td colspan="3">none</td></tr>
</table>

1.2.12.2 Logout

Gohvan Construction			
Author(s): Matthew Veltman	**Date: 28/06/2021**	**Version: 1.0**	
USE CASE NAME:	**Logout**	**USE CASE TYPE**	
USE CASE ID:	**12.2**	**Business Requirements:** ☐	
PRIORITY:	High	**System Analysis:** ☐	
SOURCE:	Gohvan Construction System	**System Design:** ☑	
PRIMARY BUSINESS ACTOR:	User (PBA) {Internal employees of Gohvan Construction}		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	None		
DESCRIPTION:	This use case describes the process whereby the user that is currently logged into the system wants to log out of the system. The use case begins when the user clicks the logout button. The system will then ask the user to confirm that they want to logout of the system. The use case concludes when the user is logged out of the system.		
PRE-CONDITION:	The user needs to be logged in.		
TRIGGER:	The person wants to log out of the system.		
TYPICAL COURSE OF EVENTS:	**ACTOR ACTION:**	**SYSTEM RESPONSE:**	
		Manual Action	Automated Action

<table>
<tr><td></td><td>Step 1: The user clicks on the logout button.</td><td></td><td>Step 2: The system displays a modal with the following elements:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Logout Message</td><td>Heading</td><td>None</td></tr>
<tr><td>Confirm</td><td>Button</td><td>None</td></tr>
<tr><td>Cancel</td><td>Button</td><td>None</td></tr>
</table></td></tr>
<tr><td rowspan="2"></td><td>Step 3: The user clicks on the confirm button. (ALT)</td><td></td><td>Step 4: The system logs the user out of the system.</td></tr>
<tr><td></td><td></td><td>Step 5: The system displays the login screen.</td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 3: The user clicks on the cancel button. Terminate this use case.</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">The system logs the user out.</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">The user cannot access the system until they login again.</td></tr>
<tr><td>BUSINESS RULES:</td><td colspan="3">Users need to logout of the system as soon as they are done working on the system.</td></tr>
<tr><td>IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:</td><td colspan="3">none</td></tr>
<tr><td>ASSUMPTIONS:</td><td colspan="3">none</td></tr>
<tr><td>OPEN ISSUES:</td><td colspan="3">none</td></tr>
</table>

1.2.12.3 Reset Password

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author(s):
Matthew Veltman</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Reset Password</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">12.3</td><td>Business Requirements: ☐</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td><td>System Analysis: ☐</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td><td>System Design: ☑</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">User (PBA) {Internal employees of Gohvan Construction}</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case describes the process a user goes through to start changing their password. The use case begins with the user forgetting their password. The use case concludes when the system begins the process of resetting the password.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">The user needs to be registered on the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The user has forgotten their password.</td></tr>
<tr><td rowspan="3">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td>Step 1: The user clicks on the forgot password link on the login screen.</td><td></td><td>Step 2: The system displays a modal with the following attributes:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Forgot password message</td><td>Heading</td><td>None</td></tr>
<tr><td>Confirm</td><td>Button</td><td>None</td></tr>
<tr><td>Cancel</td><td>Button</td><td>None</td></tr>
</table></td></tr>
<tr><td>Step 3: The user clicks on the confirm button. (ALT)</td><td></td><td>Step 4: The system displays a modal with the following attributes:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Reset password message</td><td>Heading</td><td>None</td></tr>
<tr><td>Confirm</td><td>Button</td><td>None</td></tr>
</table></td></tr>
</table>

			Cancel | Button | None
	Step 5: The user clicks on the confirm button. (ALT)		Step 6: The system displays a "Request email" card with the following attributes: Input Name | Control Type | Notes Email request message | Heading | None **Email** | label | None Email | Textbox | None Confirm | Button | None Cancel | Button | None
	Step 7: The user enters their email in the email textbox. (ALT)		Step 8: The system captures and validates the email. (ALT)
			Step 9: The system uses a SQL query to retrieve the user's personal Security question from the User table based on finding the record with the entered email. (ALT)
			Step 10: The system displays a "Security question" card with the following attributes: Input Name | Control Type | Notes Security question message | Heading | None **Security question** | label | None **Security question answer** | label | None Security question answer | Textbox | None Confirm | Button | None Cancel | Button | None
	Step 11: The user enters their answer in the security question answer textbox. (ALT)		Step 12: The system captures and validates the security question answer. (ALT)
			Step 13: The system uses a SQL query to compare and validate the user given answer

<table>
<tr><td></td><td></td><td></td><td>with the Security answer from the User table. (ALT)</td></tr>
<tr><td></td><td></td><td></td><td>Step 14: The system displays a “Reset password” card with the following attributes:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Reset password message</td><td>Heading</td><td>None</td></tr>
<tr><td>Password</td><td>label</td><td>None</td></tr>
<tr><td>Password</td><td>Textbox</td><td>None</td></tr>
<tr><td>Confirm Password</td><td>Label</td><td>None</td></tr>
<tr><td>Confirm Password</td><td>Textbox</td><td>None</td></tr>
<tr><td>Confirm</td><td>Button</td><td>None</td></tr>
<tr><td>Cancel</td><td>Button</td><td>None</td></tr>
</table></td></tr>
<tr><td></td><td>Step 15: The user enters their new password in both the password textbox and the confirm password textbox. (ALT)</td><td></td><td>Step 16: The system captures and validates the entered password. (ALT)</td></tr>
<tr><td></td><td></td><td></td><td>Step 17: The system displays a modal with the following attributes:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Reset password message</td><td>Heading</td><td>None</td></tr>
<tr><td>Confirm</td><td>Button</td><td>None</td></tr>
<tr><td>Cancel</td><td>Button</td><td>None</td></tr>
</table></td></tr>
<tr><td></td><td>Step 18: The user clicks on the confirm button. (ALT)</td><td></td><td>Step 19: A SQL query is used to update the user’s password in the User table to the new entered password.</td></tr>
<tr><td></td><td></td><td></td><td>Step 20: The system displays a modal with the following attributes:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Reset password message</td><td>Heading</td><td>None</td></tr>
<tr><td>Confirm</td><td>Button</td><td>None</td></tr>
</table></td></tr>
<tr><td>ALTERNATE</td><td colspan="3">ALT Step 3: The user cancels the forgot password. Terminate use case.</td></tr>
</table>

COURSES:	ALT Step 5: The user cancels the reset password. Terminate use case.
	ALT Step 7: The user cancels the reset password. Terminate use case.
	ALT Step 8: The entered email has a Validation error. Return to step 7.
	ALT Step 9: The entered email was not found in the User table. Return to step 7.
	ALT Step 11: The user cancels the answering of the security question. Terminate use case.
	ALT Step 12: The entered security question answer has a Validation error. Return to step 11.
	ALT Step 13: The entered security question answer does not equal the one found in the User table. Return to step 11.
	ALT Step 15: The user cancels the reset password. Terminate use case.
	ALT Step 15: The entered passwords have a Validation error. Return to step 21.
	ALT Step 18: The user cancels the reset password. Terminate use case.
CONCLUSION:	The user has reset their password
POST-CONDITION:	The user logs into the system with the new password.
BUSINESS RULES:	The user needs to register on the system.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	The user is registered on the system
OPEN ISSUES:	none

1.2.13. Client Subsystem

1.2.13.1 Add Client

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Amore Rossouw</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Add Client</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">13.1</td><td>Abstract: ☐</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">High</td><td>Extension: ☑</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td><td></td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">The Operations manager (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">• None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case narrative describes the process the Operations manager will follow when they want to create a new client. The system will capture and validate the client information if the information was validated successfully the system will store it in the CLIENT table.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">The Operations manager must be logged into the system.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The Operations manager wants to add a new client.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td></td><td></td><td>Step 1:
The system responds by loading the “Add Client” Screen with Client table information using the SQL Read query with the following controls:

<table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr></table></td></tr>
</table>

			Add Client	Header	none
			Client Name	Label	none
			Client Name	Input	none
			Email Address	Label	none
			Email Address	Input	none
			Telephone Number	Label	none
			Telephone Number	Input	none
			Vat Number	Label	none
			Vat Number	Input	none
			PO box	Label	none
			PO box	Input	none
			Business address	Label	none
			Street address	Label	none
			Street address	Input	none
			City	Label	none
			City	Input	none
			Province	Label	none
			Province	Input	none
			Zip code	Label	none
			Zip code	Input	none
			Company registration number	Label	none
			Company registration number	Input	none
			Add Client	button	none
	Step 2: The operations				

<table>
<tr><td></td><td>manager fills in the necessary information into the appropriate input fields requested by the system.</td><td></td><td></td></tr>
<tr><td></td><td>Step 3:
The Operations manager selects the “Add Client” button. [ALT]</td><td></td><td>Step 4:
The system captures and validates the information supplied by the Operations manager against the Client table using Angular Validation:
[ALT]
<table>
<tr><th>Input Name</th><th>Attribute in table</th><th>Validation requirements</th></tr>
<tr><td>Client Name</td><td>Client_ Name</td><td>required</td></tr>
<tr><td>Email Address</td><td>Email</td><td>Must be of type email</td></tr>
<tr><td>Telephone Number</td><td>Tel_No</td><td>required</td></tr>
<tr><td>Fax Number</td><td>Fax_No</td><td>required</td></tr>
<tr><td>Vat Number</td><td>Vat_No</td><td>required</td></tr>
<tr><td>PO box</td><td>Po_Box</td><td>required</td></tr>
<tr><td>Street address</td><td>Street_ Address</td><td>required</td></tr>
<tr><td>City</td><td>City</td><td>required</td></tr>
<tr><td>Province</td><td>Province</td><td>required</td></tr>
<tr><td>Zip code</td><td>Zip_code</td><td>required</td></tr>
<tr><td>Company registration number</td><td>Company _Reg_No</td><td>required</td></tr>
</table></td></tr>
<tr><td></td><td></td><td></td><td>Step 5:
The system displays an Angular Material Confirmation Modal, with the following controls, to prompt the Operations manager for confirmation of the entered details:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
</table></td></tr>
</table>

			Confirm information | Header | none Please confirm that the details entered are correct? | Label | none Yes | button | This button confirms that the entered details were correct No | button | This button confirms that the entered details were not correct
	Step 6: The Operations manager confirms the captured information by clicking the "Yes" button. **[ALT]**		**Step 7:** The system uses a SQL insert query to save the captured information in the **CLIENT** table as follows: Details | Attribute in table | Notes Client ID | Client_ ID | ID is generated by the system by using the last ID in the table and incrementing it by one. Client Name | Client_ Name | none Email Address | Email | Must be of type email Telephone Number | Tel_No | none

			<table><tr><td>Vat Number</td><td>Vat_No</td><td>none</td></tr><tr><td>PO box</td><td>Po_Box</td><td>none</td></tr><tr><td>Street address</td><td>Street_ Address</td><td>none</td></tr><tr><td>City</td><td>City</td><td>none</td></tr><tr><td>Province</td><td>Province</td><td>none</td></tr><tr><td>Zip code</td><td>Zip_code</td><td>none</td></tr><tr><td>Company registration number</td><td>Company _Reg_No</td><td>required</td></tr></table> **[ALT]**
			Step 8: The system notifies Operations manager that the Client was added successfully, using an Angular Material Success Notification Modal with the following controls: <table><tr><td>Input Name</td><td>Control Type</td><td>Notes</td></tr><tr><td>Success</td><td>Header</td><td>none</td></tr><tr><td>The Client was successfully created</td><td>Label</td><td>none</td></tr><tr><td>OK</td><td>button</td><td>None</td></tr></table>
	Step 9: The Operations manager clicks the “Ok” button.		
ALTERNATE COURSES:	**ALT Step 3:** The Operations manager selects the “Cancel” button. The use case is terminated.		
	ALT Step 4: • The captured details are not valid. The system displays a notification indicating that the information received is incorrect or incomplete with an Angular Material Invalid Notification Modal as follows:		

	<table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Information Invalid</td><td>Header</td><td>none</td></tr><tr><td>The entered details were incorrect. Please fill in the correct information.</td><td>Label</td><td>none</td></tr><tr><td>OK</td><td>button</td><td>None</td></tr></table> The system returns to step 1.
	ALT Step 6: The Operations manager does not confirm that the information captured is correct by clicking the “No” button. The system returns to step 1.
	ALT Step 7: The system does not successfully add the Client information to the CLIENT. The system notifies the user that the Client has not been added successfully with an Angular Material Error Modal as follows: <table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Error</td><td>Header</td><td>none</td></tr><tr><td>An error has occurred while saving the Client details. Please try again.</td><td>Label</td><td>none</td></tr><tr><td>OK</td><td>button</td><td>None</td></tr></table> The system terminates this use case.
CONCLUSION:	This use case concludes when the Operations manager has created a new client successfully.
POST-CONDITION:	• The client has been added to the CLIENT table successfully.

1.2.13.2 Search Client

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Amore Rossouw</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Search Client</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">13.2</td><td rowspan="3">Business Requirements: ☐
System Analysis: ☐
System Design: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Medium</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">The User (PBA): management director, director, operations manager and the financial officer. The User must be logged into the system.</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">• None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">• None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case narrative describes the process the user will follow when they want to update an existing Client. The system will capture and validate the Client information if the information was validated successfully the system will update the Client in the CLIENT table.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The user must be logged into the system.
The Client that the user wishes to update must exist in the database.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The user wants to update an existing Client.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td>Step 1:
The User wants to search for an existing client.
The user clicks on the “Client” option in the side menu bar on the home screen.</td><td></td><td>Step 2:
The system responds by loading the “Client “screen with the following controls

<table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Clients</td><td>Header</td><td>none</td></tr></table></td></tr>
</table>

<table>
<tr>
<td></td>
<td></td>
<td></td>
<td>
<table>
<tr><td>+</td><td>Button</td><td>This control is used to add a client</td></tr>
<tr><td>Search Client</td><td>Label</td><td>none</td></tr>
<tr><td>Search Parameter</td><td>Input</td><td>none</td></tr>
<tr><td>Search</td><td>Button</td><td>none</td></tr>
<tr><td>Client Table</td><td>Table</td><td>This table is populated with the “client name” attribute from the CLIENT table. An “edit” button will be next to each Client record.</td></tr>
<tr><td>Edit Client</td><td>Button</td><td>This control is used to edit a client</td></tr>
</table>
</td>
</tr>
<tr>
<td></td>
<td>Step 3:
The User clicks on the “Search Client” input and enters the requested search parameter. Thereafter the user clicks on the “Search Client” button.</td>
<td></td>
<td>Step 4:
The system captures and validates the information, supplied by the User, against the CLIENT table using Angular Validation:

<table>
<tr><th>Input Name</th><th>Attribute in table</th><th>Validation requirements</th></tr>
<tr><td>Client Name</td><td>Client_Name</td><td>Required</td></tr>
</table>

[ALT]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Step 5:
A SQL read query is used to read the following attributes from the following tables based on the parameter provided:

The Table is searched based on the Client Name:

CLIENT Table:
• Client_ID
• Client_Name
• Email
• Telephone Number
• PO box
• Street_Address</td>
</tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>

City
Province
Zip_code
Vat_No
Company_Reg_No

The table attributes are read via an HttpGet method in the Typescript component and an API request.
</td></tr>
<tr><td></td><td></td><td></td><td>Step 6:
The system populates the ‘Client Table’ with the data received in the previous step
[ALT]</td></tr>
<tr><td></td><td>Step 7:
The User selects the Client from the search result by clicking on the “Client Name” label.
[ALT]</td><td></td><td>Step 8:
The system responds by loading the “View Client” page with the following controls:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Client Name</td><td>Header</td><td>This header’s text is dynamically changed to the selected Client’s name.</td></tr>
<tr><td>Telephone Number</td><td>Label</td><td>none</td></tr>
<tr><td>Telephone Number</td><td>label</td><td>none</td></tr>
<tr><td>Email Address:</td><td>label</td><td>none</td></tr>
<tr><td>Email Address</td><td>label</td><td>This label’s text is dynamically changed to the selected Client’s email.</td></tr>
<tr><td>Vat Number:</td><td>label</td><td>none</td></tr>
<tr><td>Vat Number</td><td>label</td><td>This label’s text is dynamically changed to the selected</td></tr>
</table>
</td></tr>
</table>

					Client's vat number.
			PO box:	label	none
			PO box	label	This label's text is dynamically changed to the selected Client's PO box.
			Street address	label	none
			Street address	label	This label's text is dynamically changed to the selected Client's Street address.
			City	label	none
			City	label	This label's text is dynamically changed to the selected Client's City.
			Province	label	none
			Province	label	This label's text is dynamically changed to the selected Client's Province.
			Zip code	label	none
			Zip code	label	This label's text is dynamically changed to the selected Client's Zip code.
			Company registration number:	label	none
			Company registration number	label	This label's text is dynamically

<table>
<tr><td></td><td colspan="3"><table><tr><td></td><td></td><td>changed to the selected Client's registration number</td></tr></table></td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 4:
The captured search parameters are not valid. The system displays an Angular Material Invalid Notification modal indicating that the information received is incorrect or incomplete:<table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>Information Invalid</td><td>Header</td><td>none</td></tr><tr><td>The entered parameters were incorrect. Please try again.</td><td>Label</td><td>none</td></tr><tr><td>OK</td><td>button</td><td>None</td></tr></table>The system returns to step 2.</td></tr>
<tr><td></td><td colspan="3">ALT Step 6:
The client that the user searched does not exist within the database. The system notifies the user that no search results were found by displaying an Angular Material Error modal with the following controls:<table><tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr><tr><td>No Client found</td><td>Header</td><td>none</td></tr><tr><td>There was no Client found that matched your search request.</td><td>Label</td><td>none</td></tr><tr><td>OK</td><td>button</td><td>None</td></tr></table>The system will return to Step 2.</td></tr>
<tr><td></td><td colspan="3">ALT Step 7 (A):
The Operations manager clicks on the “Add (+)” button. The system invokes use case “13.1 Add Client”</td></tr>
<tr><td></td><td colspan="3">ALT Step 7 (B):
The Operations manager clicks on the “Edit” button for the Client they want to edit. The system invokes use case “13.3 Edit Client”</td></tr>
<tr><td>CONCLUSION:</td><td colspan="3">This use case concludes when the system retrieves and displays the client search results.</td></tr>
<tr><td>POST-CONDITION:</td><td colspan="3">• The searched client is retrieved and displayed.</td></tr>
<tr><td>BUSINESS RULES:</td><td colspan="3">Only the users mentioned above are allowed to search the client.</td></tr>
</table>

IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	none
ASSUMPTIONS:	none
OPEN ISSUES:	none

1.2.13.3 Edit Client

<table>
<tr><th colspan="4">Gohvan Construction</th></tr>
<tr><td>Author (s):
Amore Rossouw</td><td colspan="2">Date: 28/06/2021</td><td>Version: 1.0</td></tr>
<tr><td>USE CASE NAME:</td><td colspan="2">Edit Client</td><td>USE CASE TYPE</td></tr>
<tr><td>USE CASE ID:</td><td colspan="2">13.3</td><td rowspan="3">Abstract: ☐
Extension: ☑</td></tr>
<tr><td>PRIORITY:</td><td colspan="2">Medium</td></tr>
<tr><td>SOURCE:</td><td colspan="2">Gohvan Construction System</td></tr>
<tr><td>PRIMARY BUSINESS ACTOR:</td><td colspan="3">The Operations manager (PBA)</td></tr>
<tr><td>PRIMARY THE SYSTEM ACTOR:</td><td colspan="3">None</td></tr>
<tr><td>OTHER PARTICIPATING ACTORS:</td><td colspan="3">None</td></tr>
<tr><td>OTHER INTERESTED STAKEHOLDERS:</td><td colspan="3">None</td></tr>
<tr><td>DESCRIPTION:</td><td colspan="3">This use case narrative describes the process the Operations manager will follow when they want to update an existing client. The system will capture and validate the client information if the information was validated successfully the system will update the client in the CLIENT table.</td></tr>
<tr><td>PRE-CONDITION:</td><td colspan="3">• The user must be logged into the system.
The client that the user wishes to update must exist in the database.</td></tr>
<tr><td>TRIGGER:</td><td colspan="3">The Operations manager wants to update an existing client.</td></tr>
<tr><td rowspan="2">TYPICAL COURSE OF EVENTS:</td><td rowspan="2">ACTOR ACTION:</td><td colspan="2">SYSTEM RESPONSE:</td></tr>
<tr><td>Manual Action</td><td>Automated Action</td></tr>
<tr><td></td><td></td><td colspan="2">Step 1:
The system responds by loading the “Edit Client” screen with the following controls:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Edit Client</td><td>Header</td><td>none</td></tr>
<tr><td>Client Name</td><td>Label</td><td>none</td></tr>
<tr><td>Client Name</td><td>Input</td><td>This control is</td></tr>
</table></td></tr>
</table>

					Populated with the current Client_Name attribute that was obtained in the use case ”13.2 Search Client”.
			Email Address	Label	none
			Email Address	Input	This control is Populated with the current Email attribute that was obtained in the use case ”13.2 Search Client”.
			Telephone Number	Label	none
			Telephone Number	Input	This control is Populated with the current Tel_No attribute that was obtained in the use case ”13.2 Search Client”.
			Vat Number	Label	none
			Vat Number	Input	This control is Populated with the current Vat_No attribute that was obtained in the use case ”13.2 Search Client”.
			PO box	Label	none

			PO box	Input	This control is Populated with the current PO_box attribute that was obtained in the use case ”13.2 Search Client”.
			Business address	Label	none
			Street address	Label	none
			Street address	Input	This control is Populated with the current Street_ Address attribute that was obtained in the use case ”13.2 Search Client”.
			City	Label	none
			City	Input	This control is Populated with the current city attribute that was obtained in the use case ”13.2 Search Client”.
			Province	Label	none
			Province	Input	This control is Populated with the current province attribute that was obtained in the use case ”13.2

			Search Client".
			Zip code	Label	none
			Zip code	Input	This control is Populated with the current Zip_code attribute that was obtained in the use case "13.2 Search Client".
			Company registration number:	label	none
			Company registration number	label	This control is Populated with the current company reg number attribute that was obtained in the use case "13.2 Search Client".
			Edit Client	Button	none
	Step 2: The operations manager fills in the necessary information into the appropriate input fields requested by the system. The operations manager selects the "Edit Client" button. **[ALT]**		**Step 3**: The system captures and validates the information supplied by the Operations manager against the **Client** table using Angular Validation: **[ALT]**

Input Name	Attribute in table	Validation requirements
Client Name	Client_ Name	required
Email Address	Email	Must be of type email
Telephone Number	Tel_No	required

Fax Number	Fax_No	required
Vat Number	Vat_No	required
PO box	Po_Box	required
Street address	Street_ Address	required
City	City	required
Province	Province	required
Zip code	Zip_code	required
Company registration number	Company _Reg_No	required

Step 5:

The system displays an Angular Material Confirmation Modal, with the following controls, to prompt the Operations manager for confirmation of the entered details:

Input Name	Control Type	Notes
Confirm information	Header	none
Please confirm that the details entered are correct?	Label	none
Yes	button	This button confirms that the entered details were correct
No	button	This button confirms that the entered details were not correct

Step 6:		**Step 7:**

The Operations manager confirms the captured information by clicking the “Yes” button. **[ALT]**

The system uses a SQL update query to save the captured information in the **CLIENT** table as follows:

Details	Attribute in table	Notes
Client ID	Client_ ID	ID is generated by the system by using the last ID in the table and incrementing it by one.
Client Name	Client_ Name	none
Email Address	Email	Must be of type email
Telephone Number	Tel_No	none
Vat Number	Vat_No	none
PO box	Po_Box	none
Street address	Street_ Address	none
City	City	none
Province	Province	none
Zip code	Zip_code	none
Company registration number	Company _Reg_No	required

[ALT]

Step 8:
The system notifies Operations manager that the Client was
Updated successfully, using an Angular Material Success Notification Modal with the following controls:

Input Name	Control Type	Notes
Success	Header	none
The Client was	Label	none

<table>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>successfully updated</td><td></td><td></td></tr>
<tr><td>OK</td><td>button</td><td>None</td></tr>
</table>
</td></tr>
<tr><td></td><td>Step 9:
The Operations manager clicks the “Ok” button.</td><td></td><td></td></tr>
<tr><td>ALTERNATE COURSES:</td><td colspan="3">ALT Step 2:
The Operations manager selects the “Cancel” button. The use case is terminated.</td></tr>
<tr><td></td><td colspan="3">ALT Step 3:
• The captured details are not valid. The system displays a notification indicating that the information received is incorrect or incomplete with an Angular Material Invalid Notification Modal as follows:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Information Invalid</td><td>Header</td><td>none</td></tr>
<tr><td>The entered details were incorrect. Please fill in the correct information.</td><td>Label</td><td>none</td></tr>
<tr><td>OK</td><td>button</td><td>None</td></tr>
</table>
The system returns to step 1.</td></tr>
<tr><td></td><td colspan="3">ALT Step 5:
The Operations manager does not confirm that the information captured is correct by clicking the “No” button.
The system returns to step 1.</td></tr>
<tr><td></td><td colspan="3">ALT Step 6:
The system does not successfully update the Client information to the CLIENT table. The system notifies the user that the Client has not been updated successfully with an Angular Material Error Modal as follows:
<table>
<tr><th>Input Name</th><th>Control Type</th><th>Notes</th></tr>
<tr><td>Error</td><td>Header</td><td>none</td></tr>
<tr><td>An error has occurred while saving the Client</td><td>Label</td><td>none</td></tr>
</table></td></tr>
</table>

	details. Please try again. | | | OK | button | None The system terminates this use case
CONCLUSION:	This use concludes when the Operations manager has successfully updated the Client.
POST-CONDITION:	• The client has been updated in the CLIENT table

Conclusion

This section of the deliverable contained the use case diagrams and the use case narratives. The use case diagrams showed the overall view of the system. The use case narratives showed in detail how the system will complete the required business processes.

2. Process - Oriented Design

Introduction

This section indicates context diagram, High, Middle, Primitive data flow diagrams as well as a Decomposition diagram which displays the data flows in and out of the system as a whole for Gohvan Construction.

2.1 Context Diagram

2.1.1 Establishment Subsystem

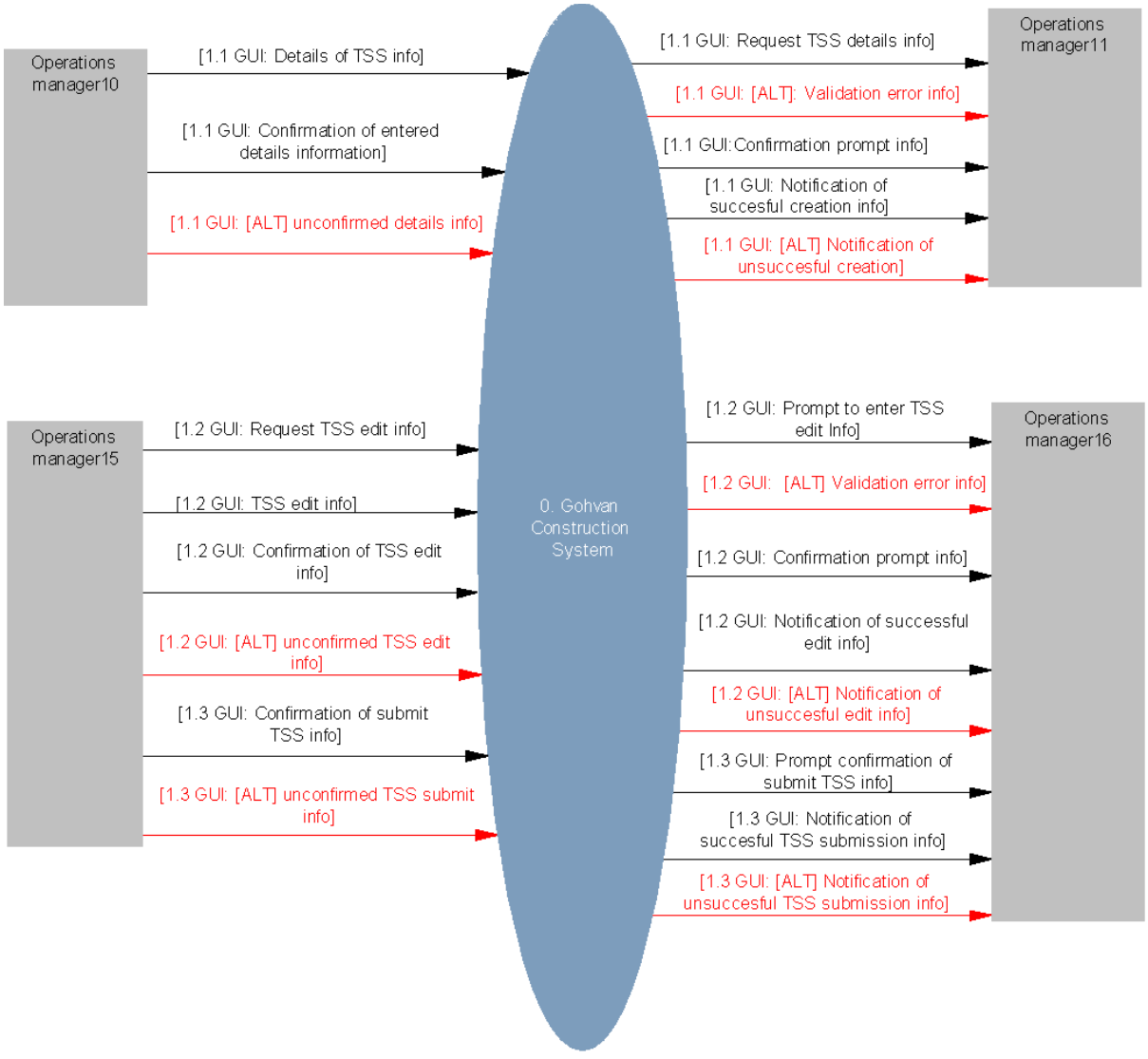


Figure 16: Context diagram Establishment Subsystem

2.1.2 Quotation Subsystem

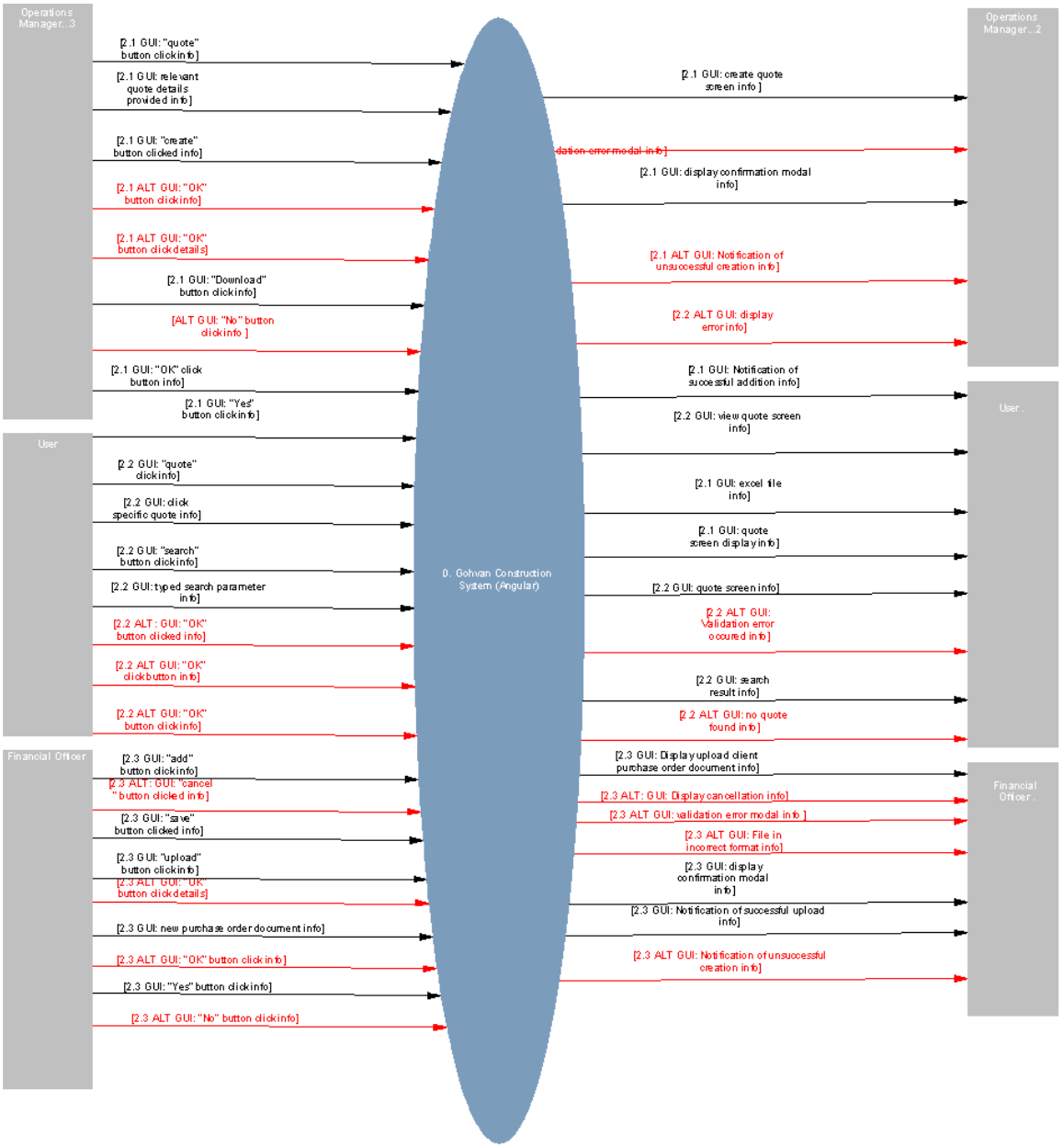


Figure 17: Context diagram Quotation Subsystem

2.1.3 Administration Subsystem

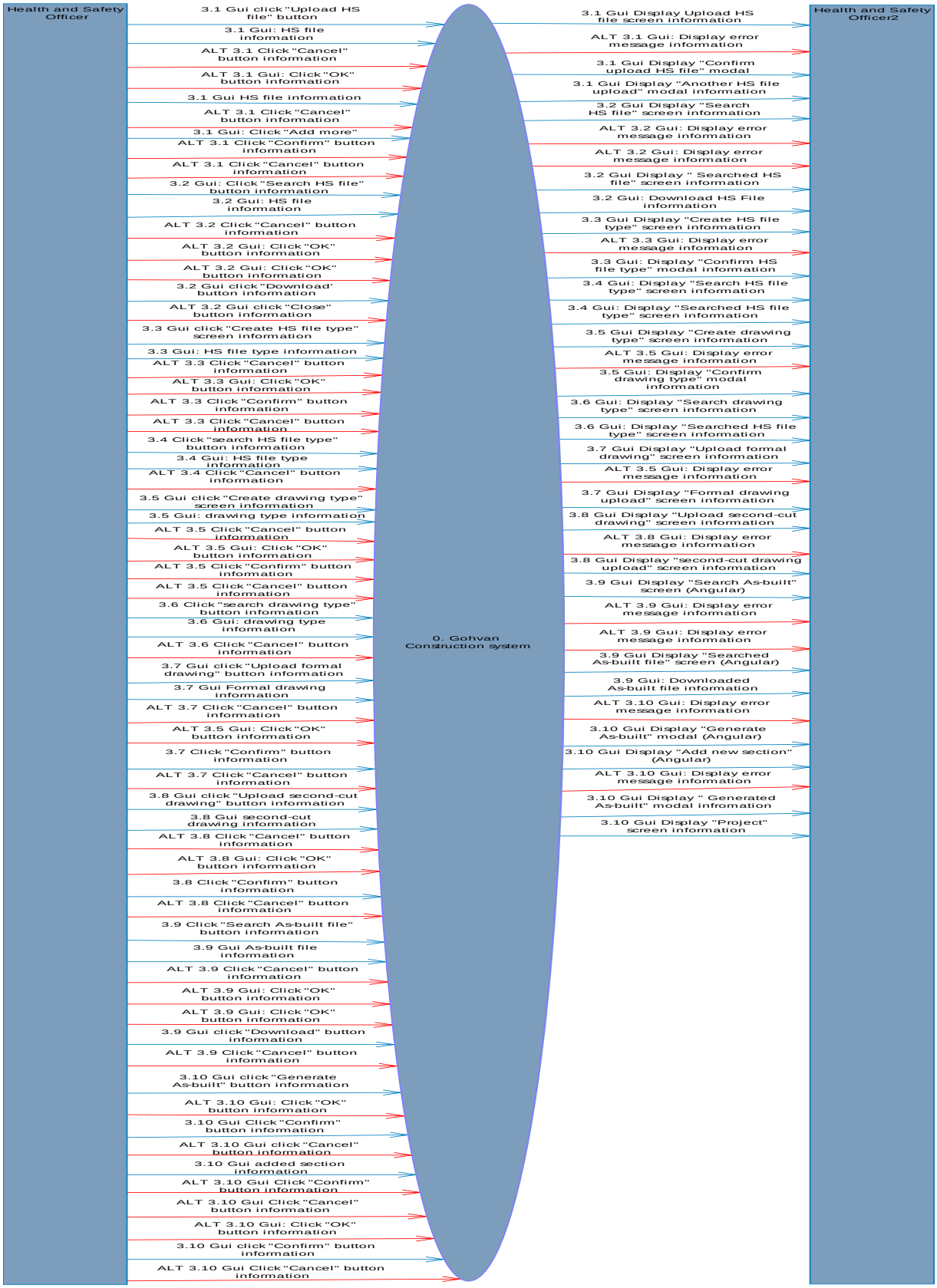


Figure 18: Context diagram Administration Subsystem

2.1.4 Construction Subsystem

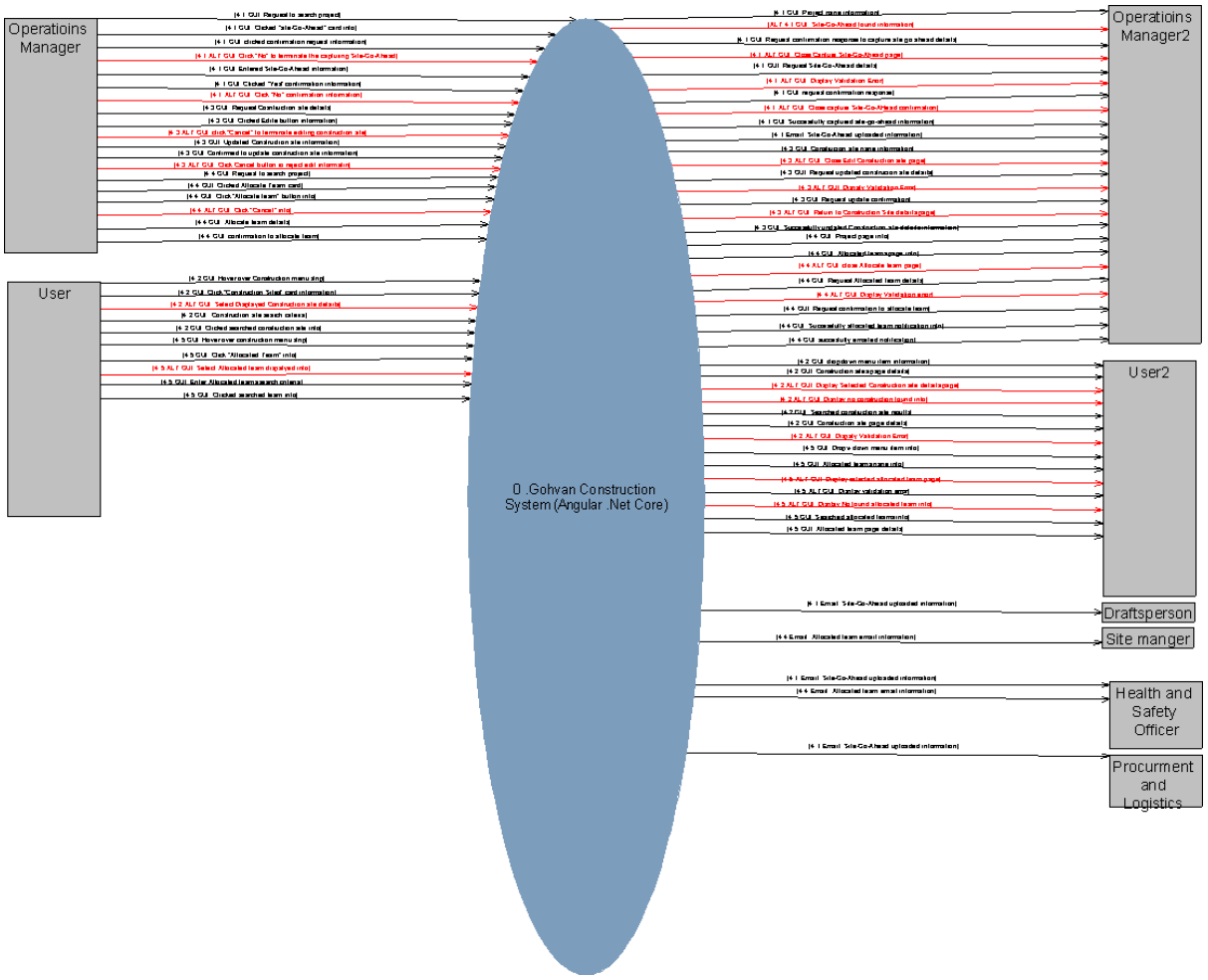


Figure 19: Context diagram Construction Subsystem 4.1-4.5

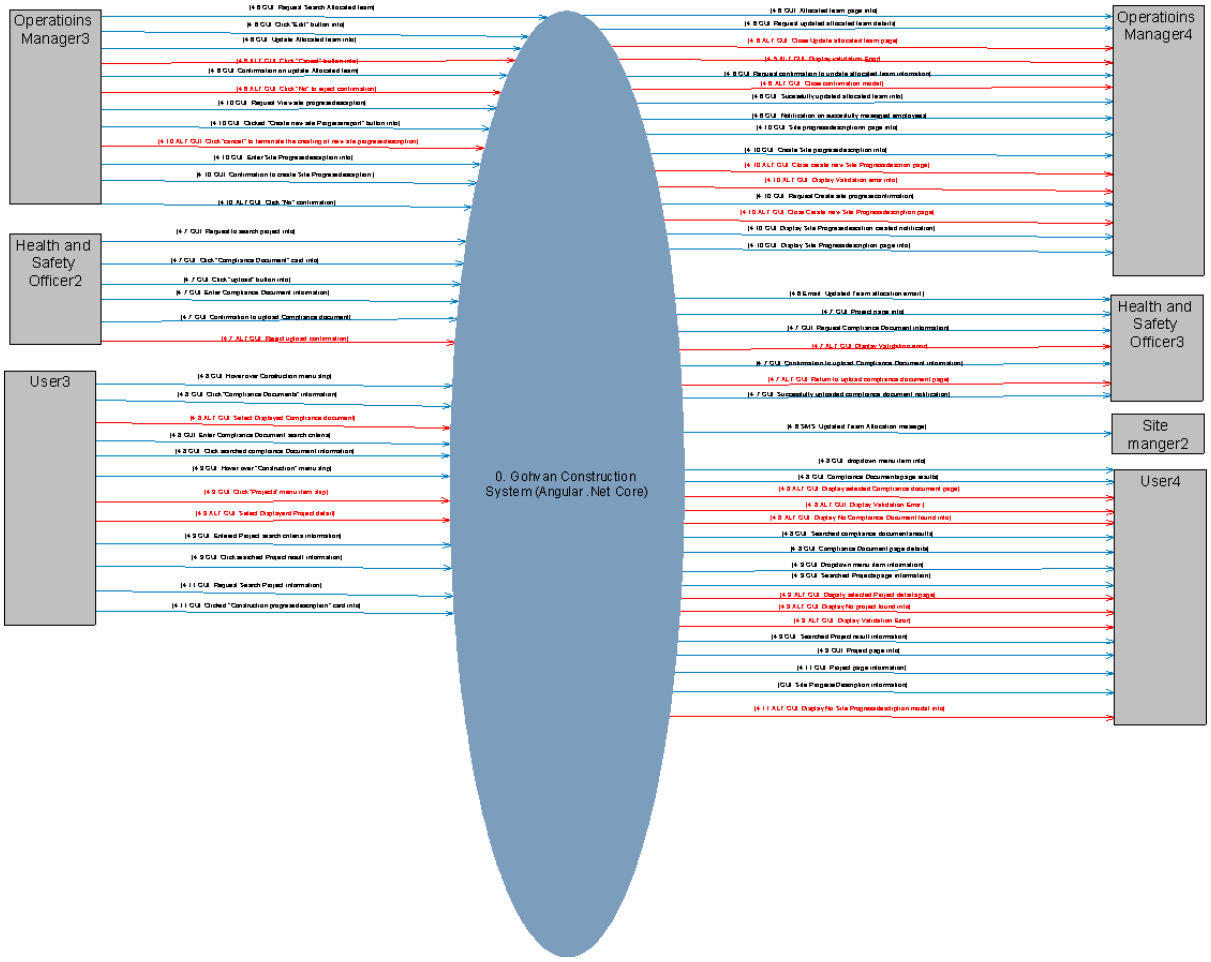


Figure 20: Context diagram Construction Subsystem 4.6-4.11

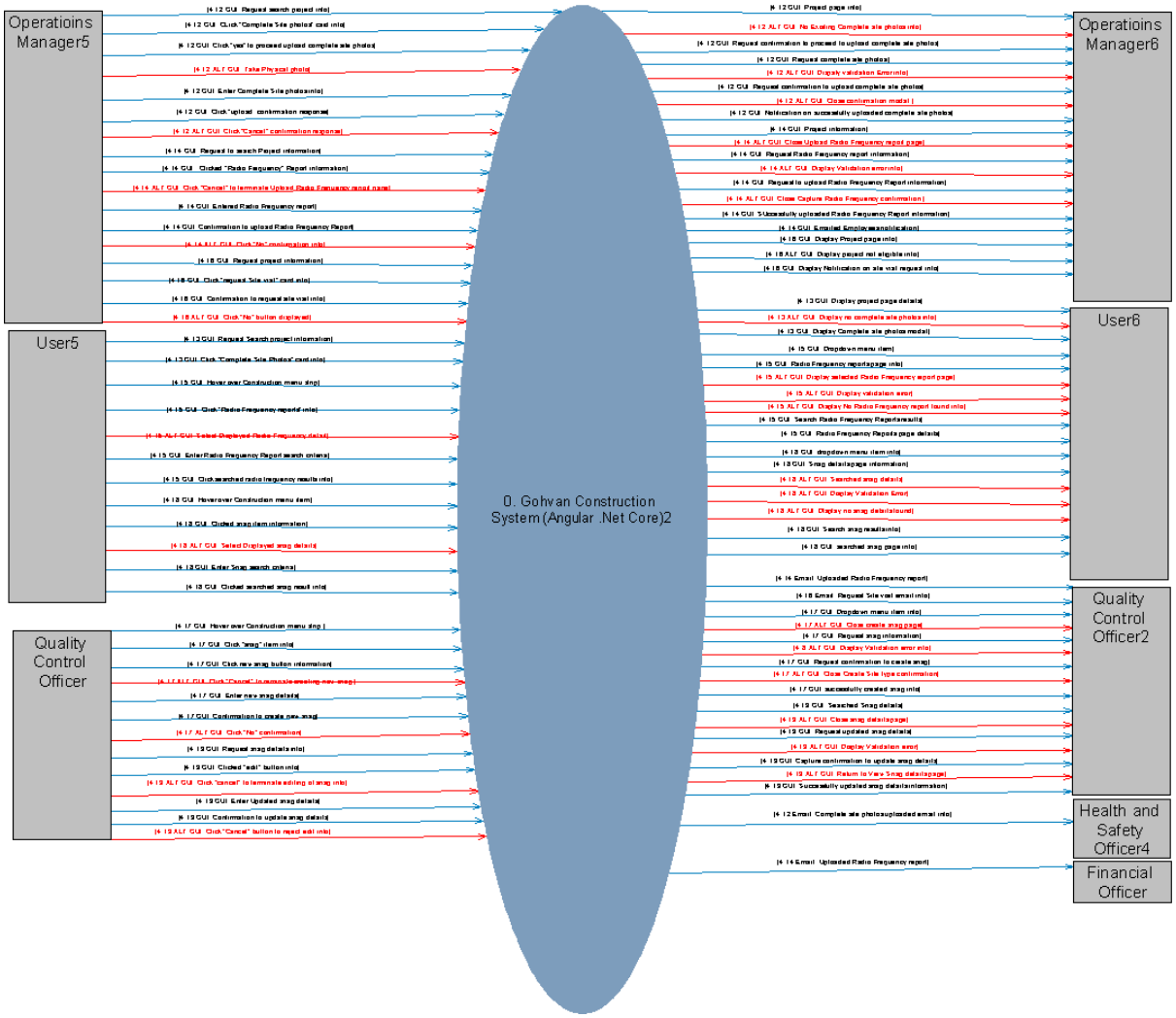


Figure 21: Context diagram Construction Subsystem 4.12-4.19



Figure 22: Context diagram Construction Subsystem 4.20-4.24 & 4.26 & 4.28

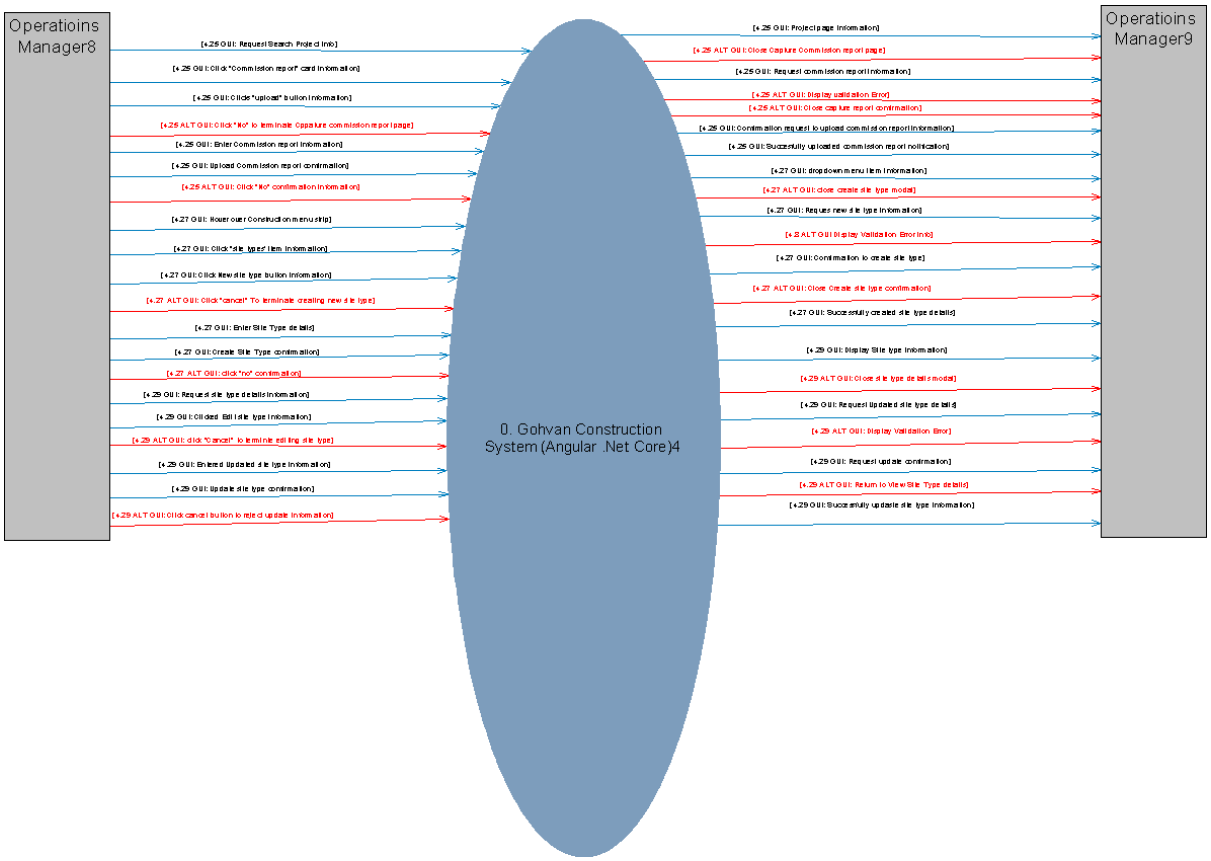


Figure 23: Context diagram Construction Subsystem 4.25 & 4.27 & 4.29

2.1.5 Invoicing Subsystem

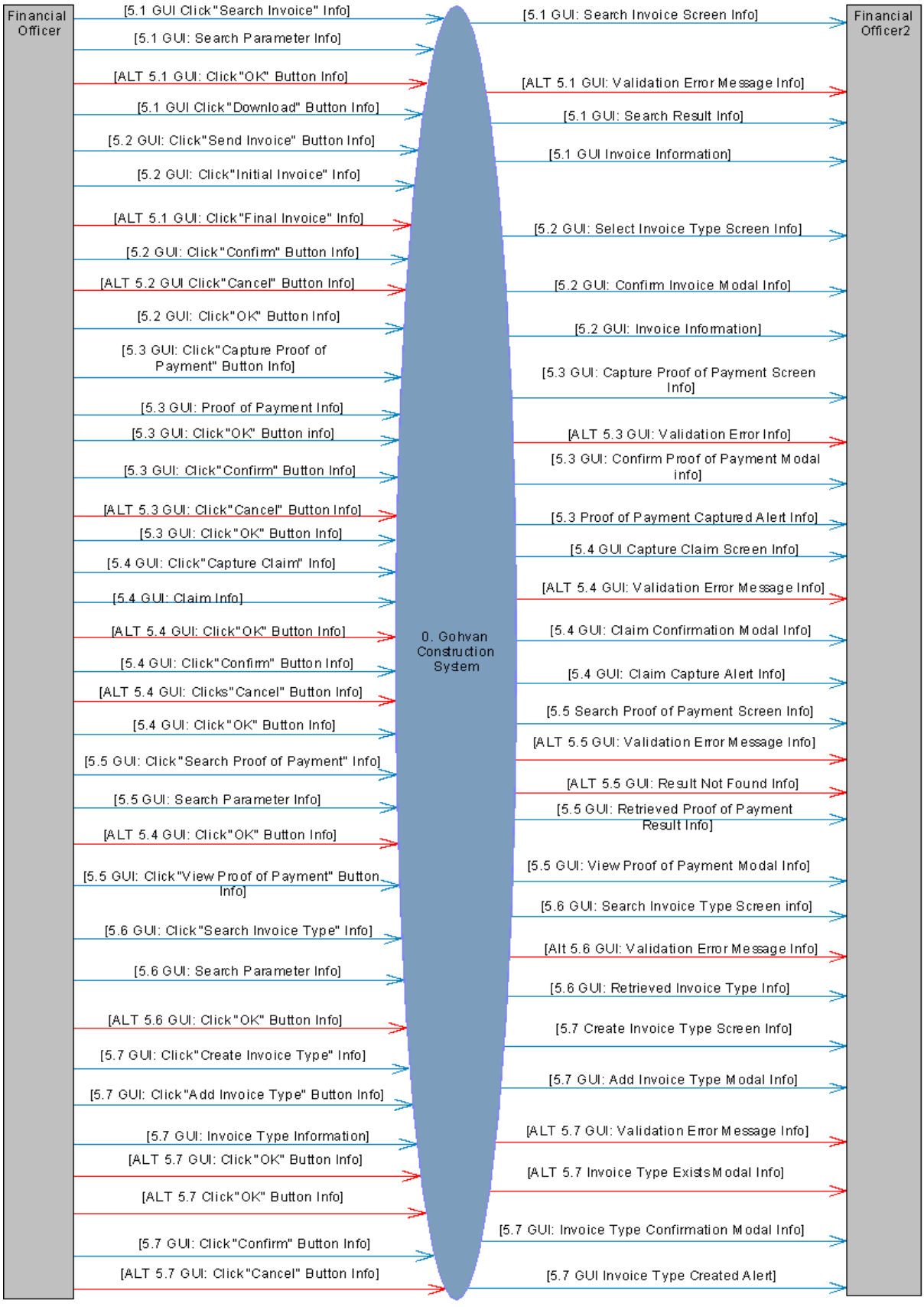


Figure 24: Context diagram Invoicing Subsystem

2.1.6 Human Resource Subsystem

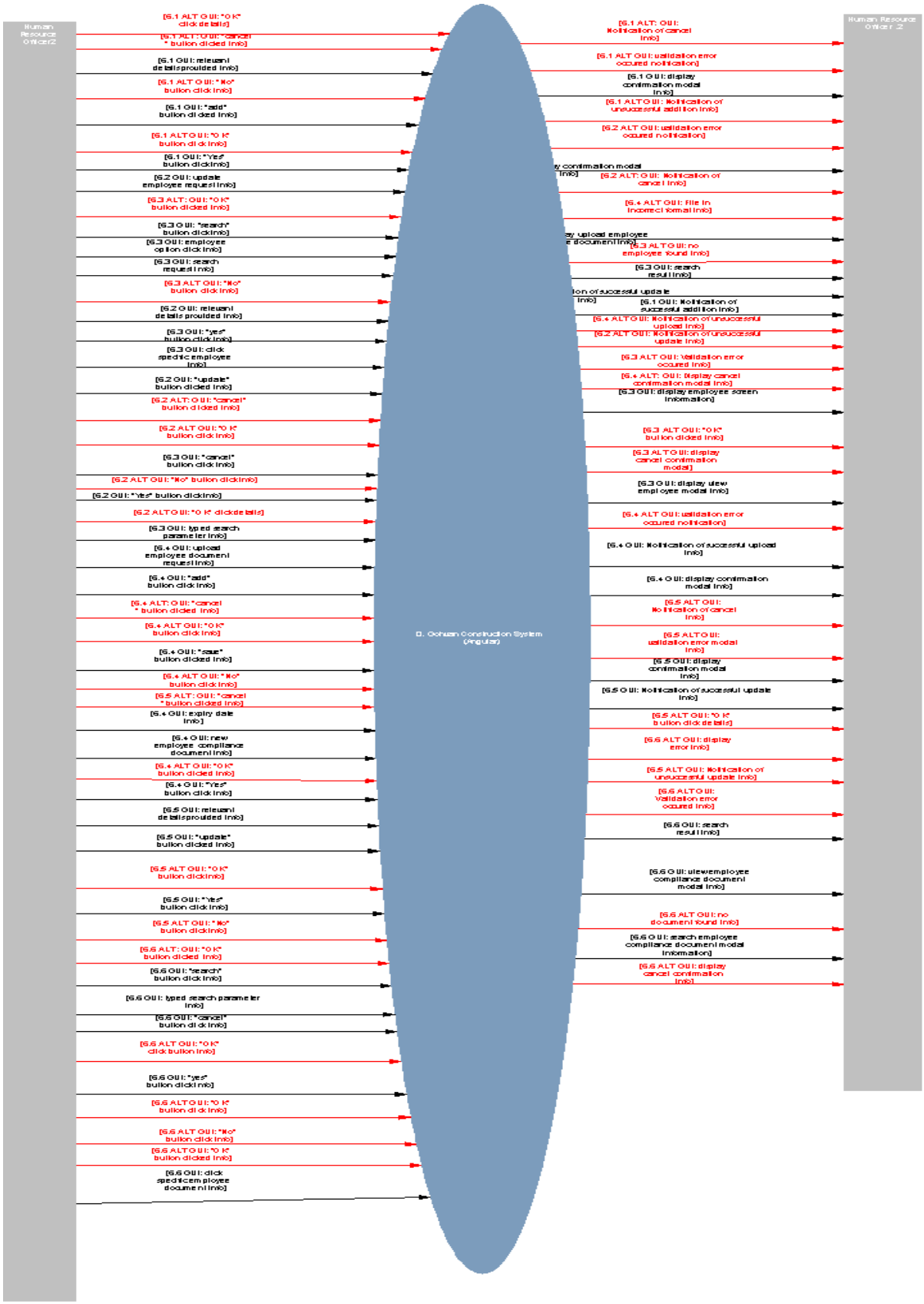


Figure 25: Context diagram Human Resources Subsystem 6.1-6.6

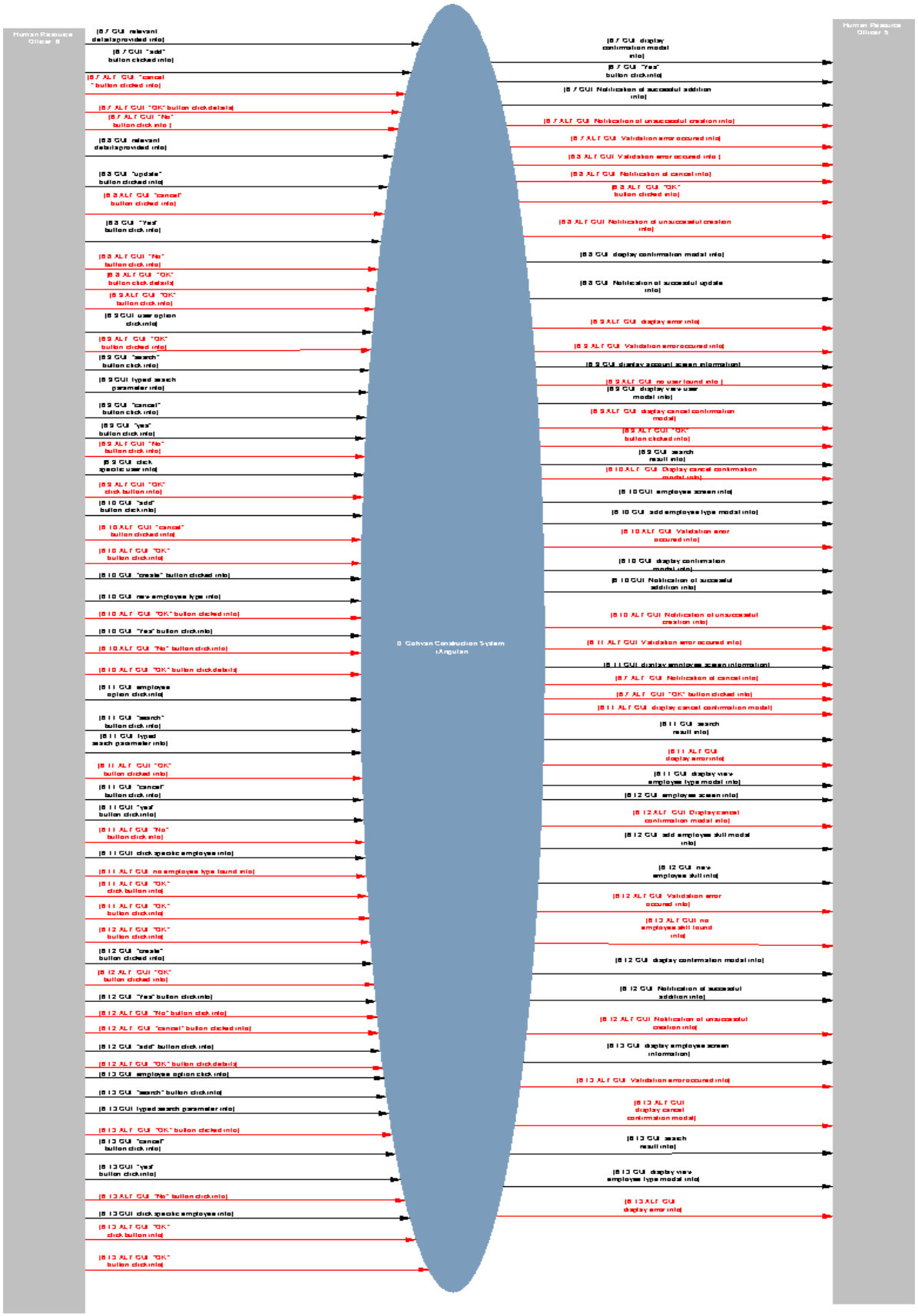


Figure 26: Context diagram Human Resources Subsystem 6.7-6.13

2.1.7 Teams Subsystem

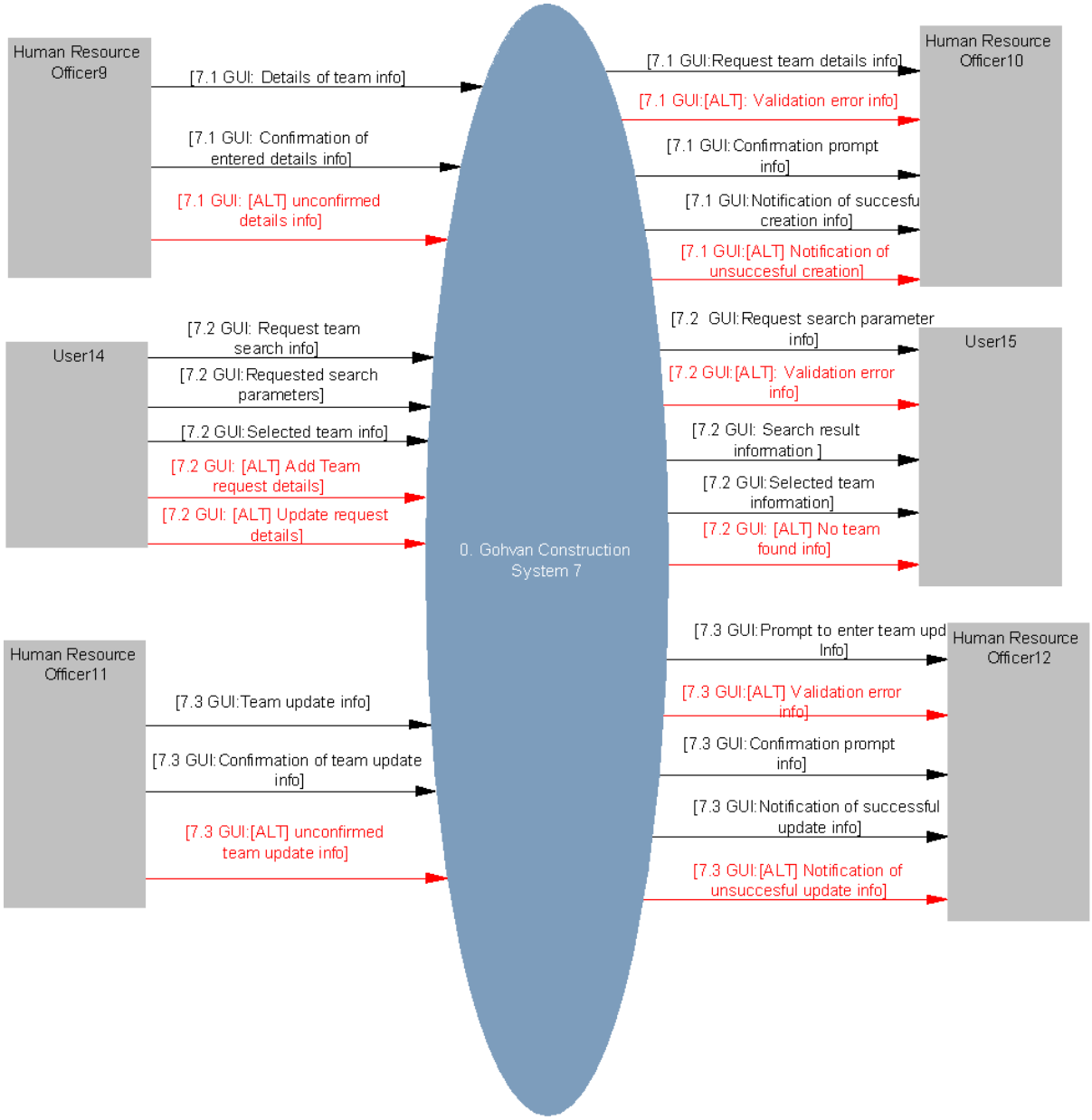


Figure 27: Context diagram Team Subsystem

2.1.8 Supplier Order Subsystem

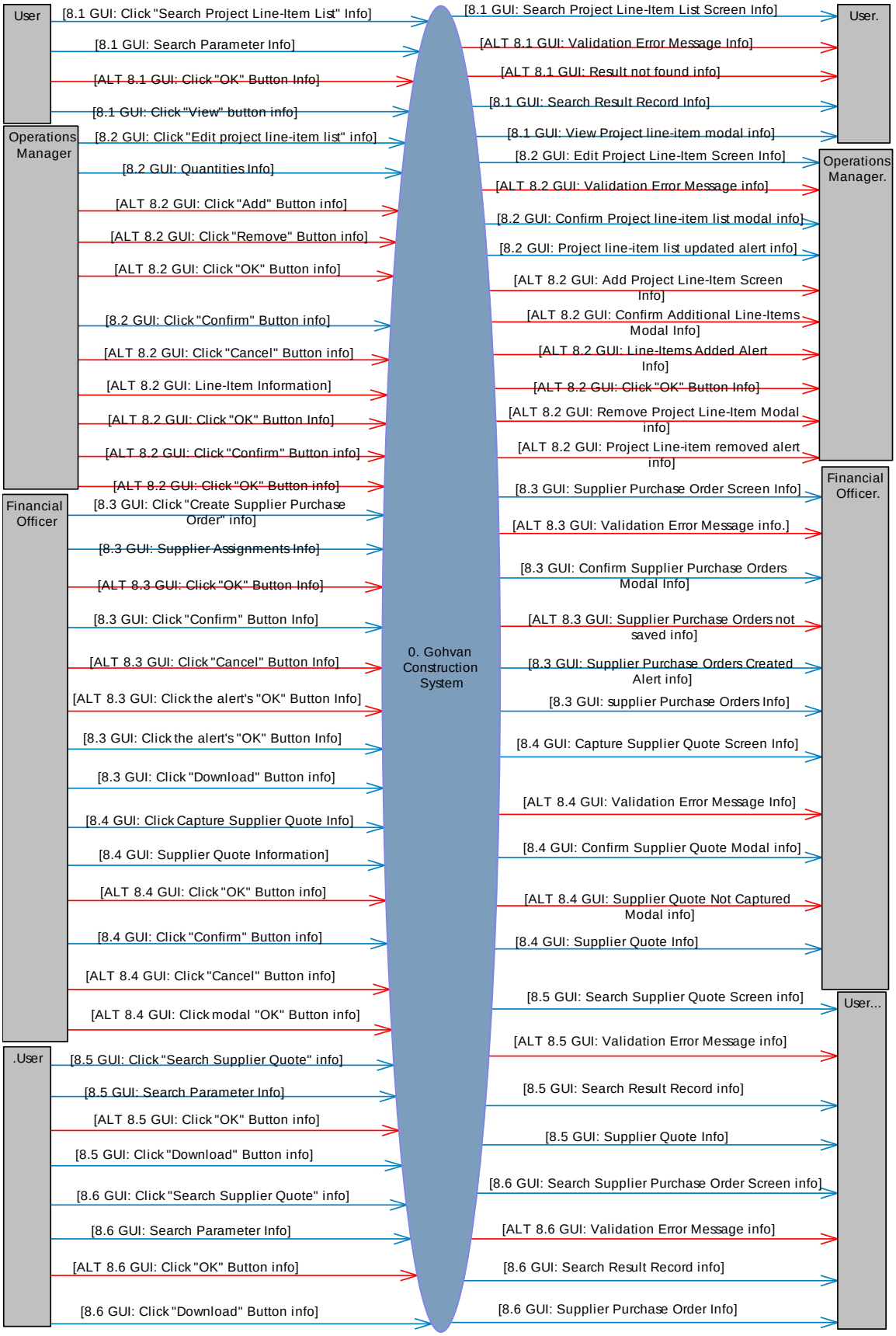


Figure 28: Context diagram Supplier Order Subsystem

2.1.9 Supplier Subsystem

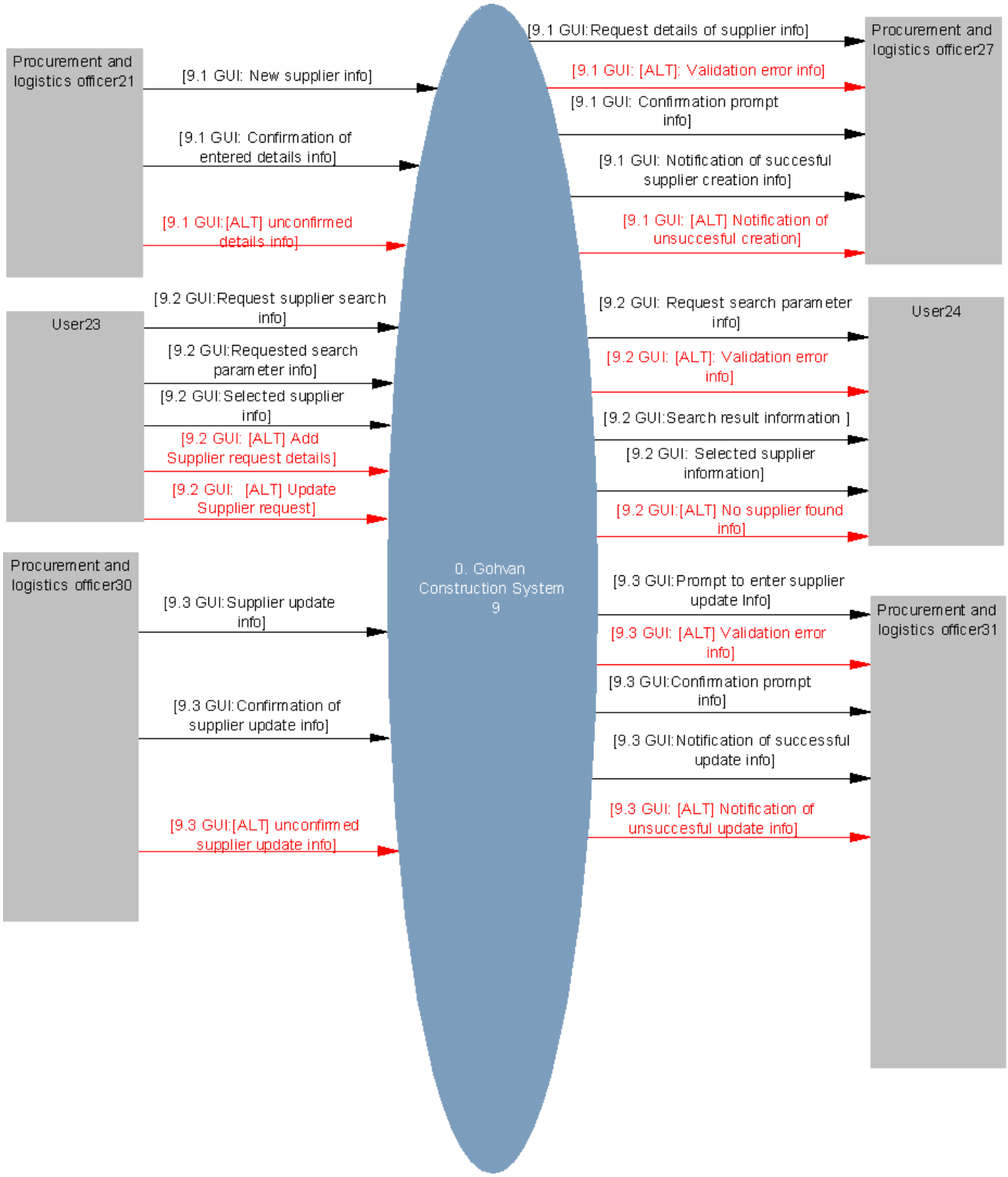


Figure 29: Context diagram Supplier Subsystem

2.1.10 Inventory Management Subsystem

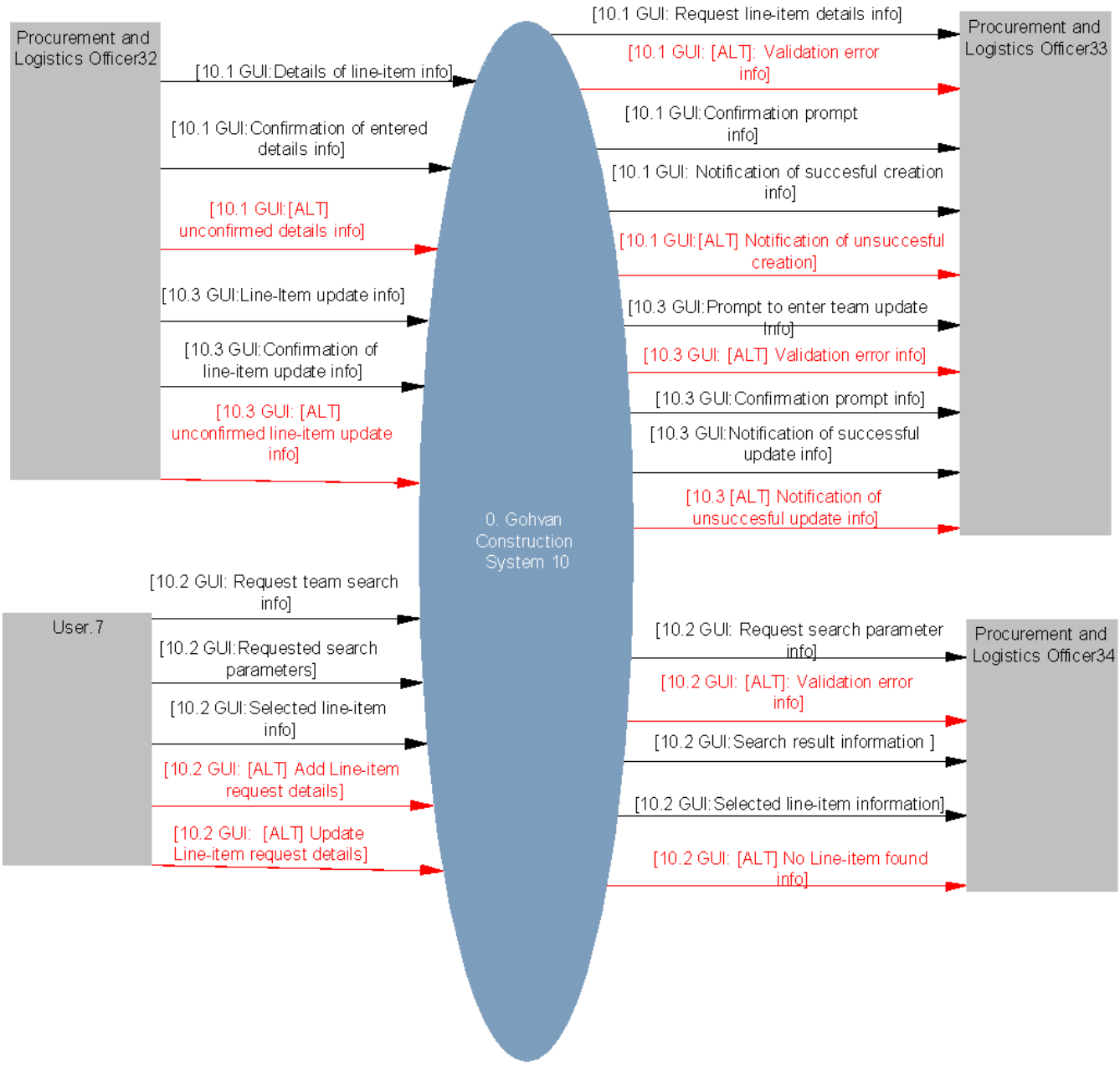


Figure 30: Context diagram Inventory Subsystem

2.1.11 Report Subsystem

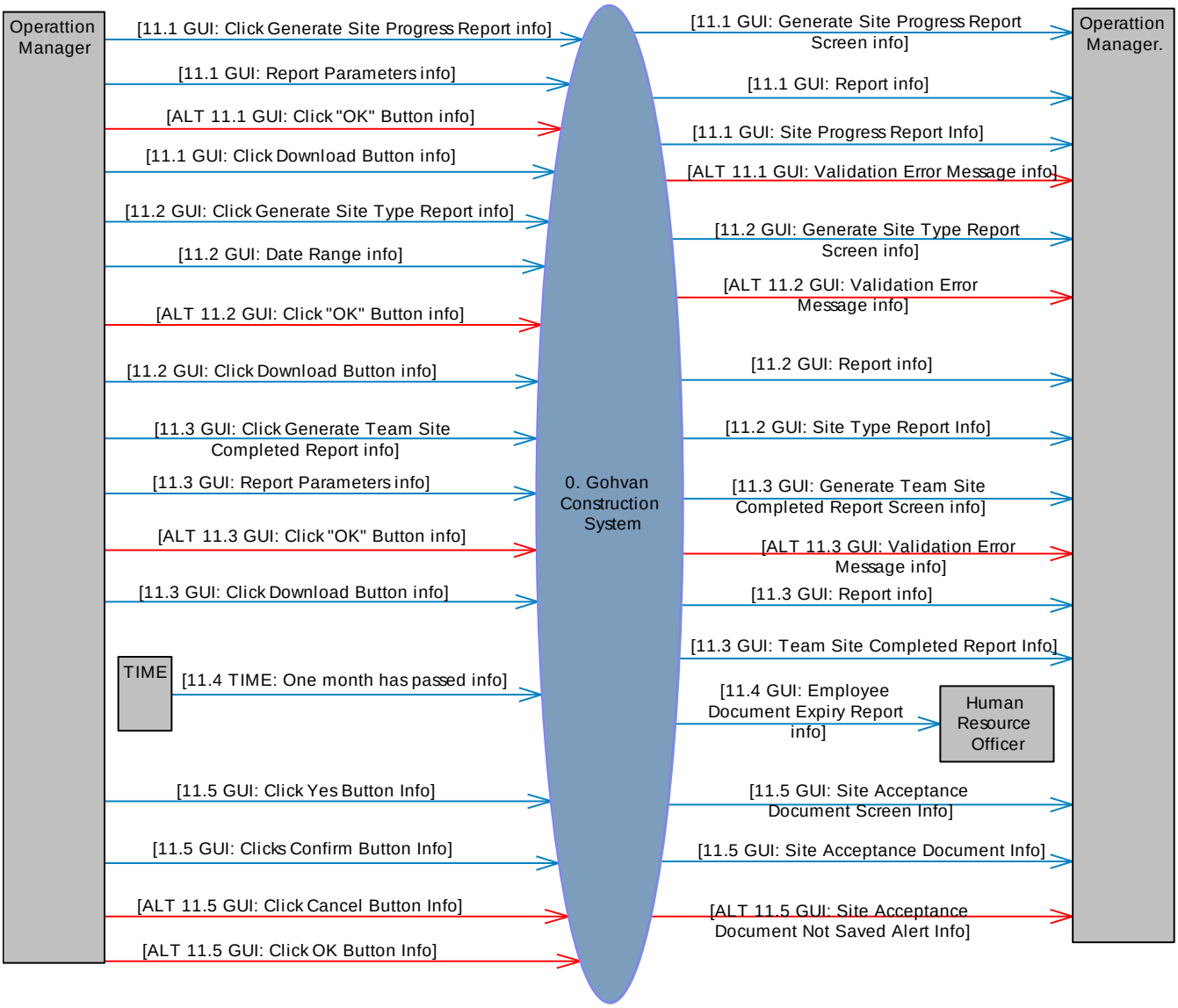


Figure 31: Context diagram Report Subsystem

2.1.12 User Management Subsystem

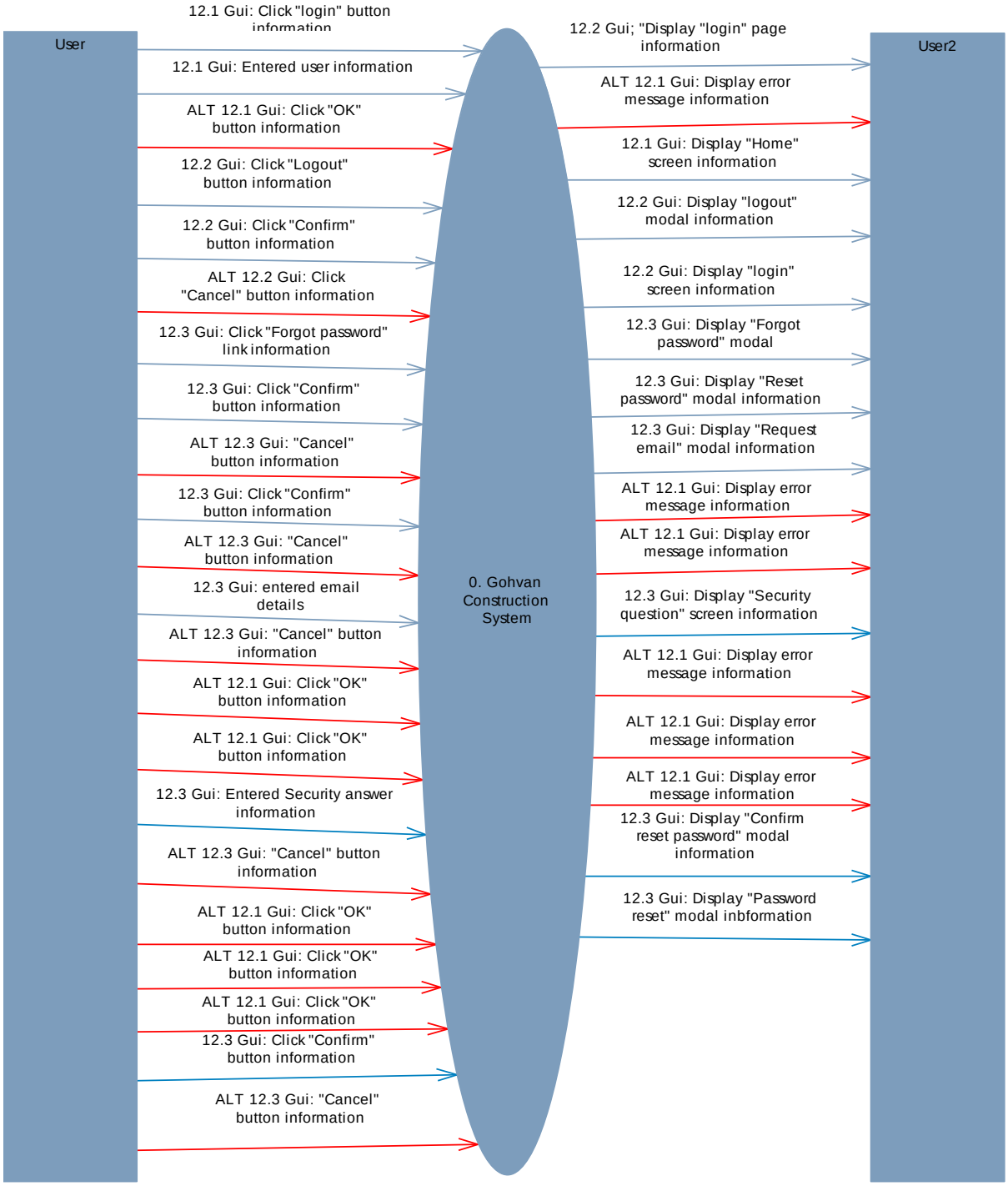


Figure 32: Context diagram User Management Subsystem

2.1.13 Client Subsystem

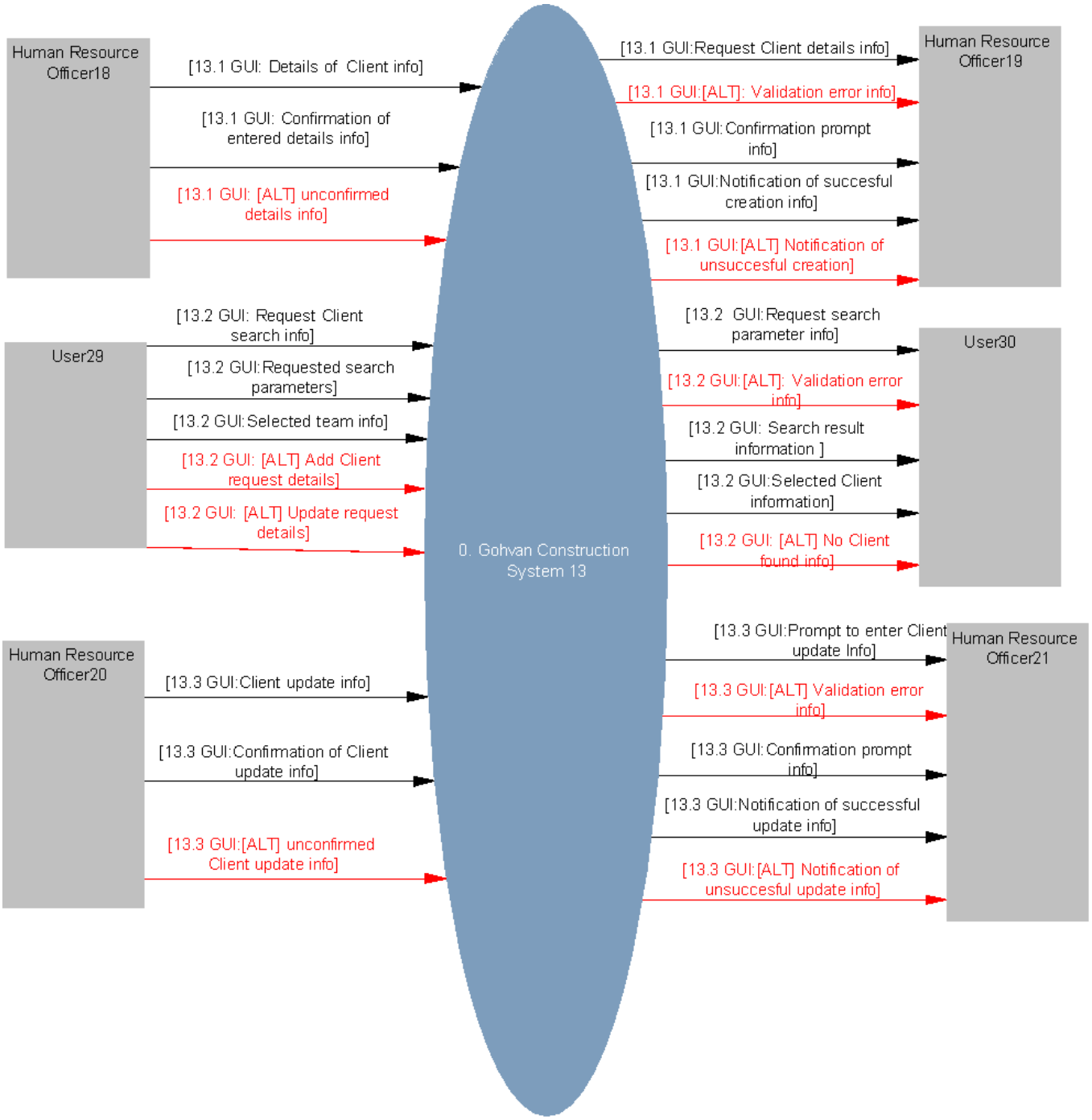


Figure 33: Context diagram Client Subsystem

2.2 Decomposition

2.2.1 Establishment Subsystem

2.2.2 Quotation Subsystem

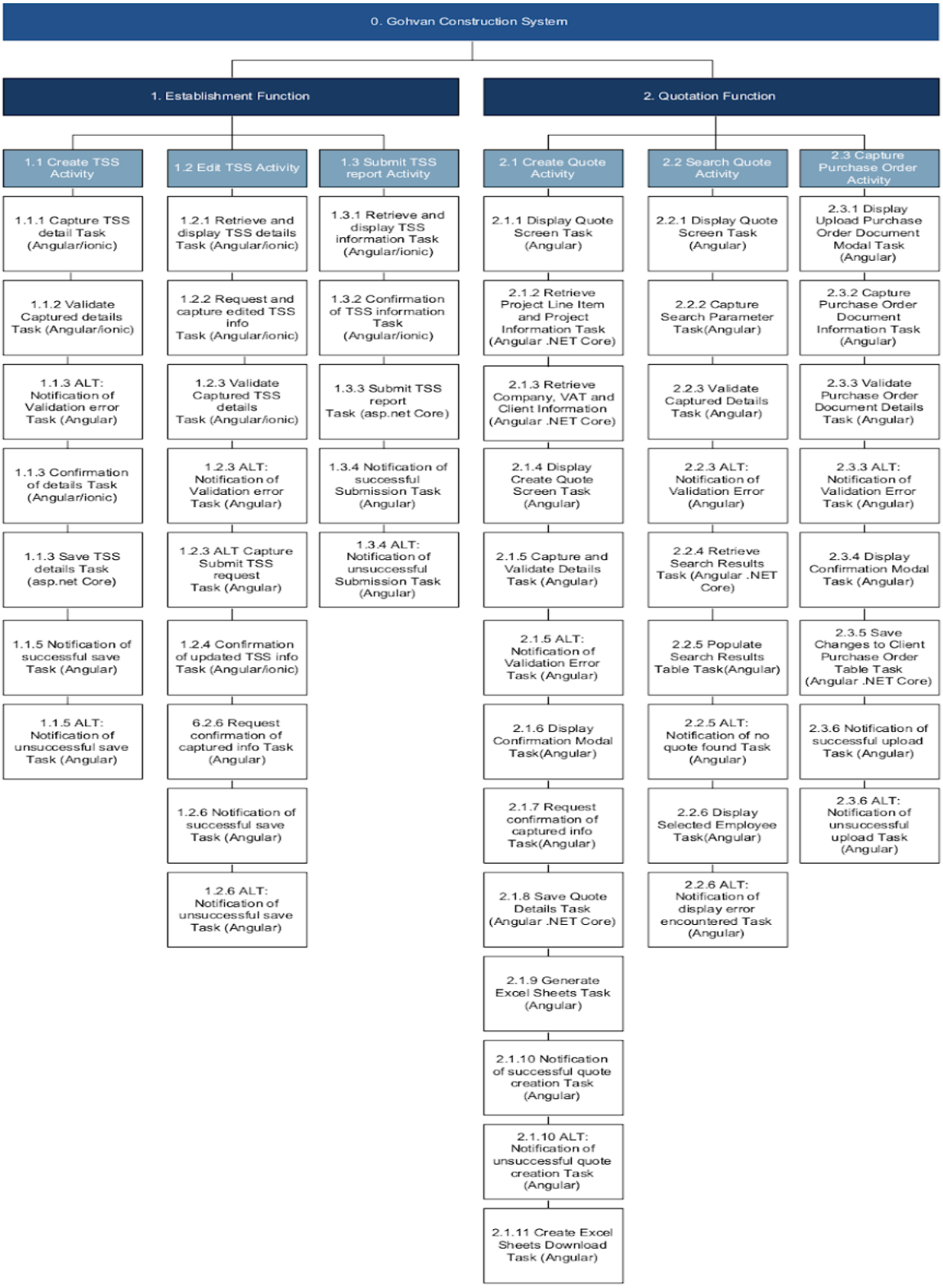


Figure 34: Decomposition diagram - 1. Establishment Function and 2. Quotation Function

2.2.3 Administration Subsystem

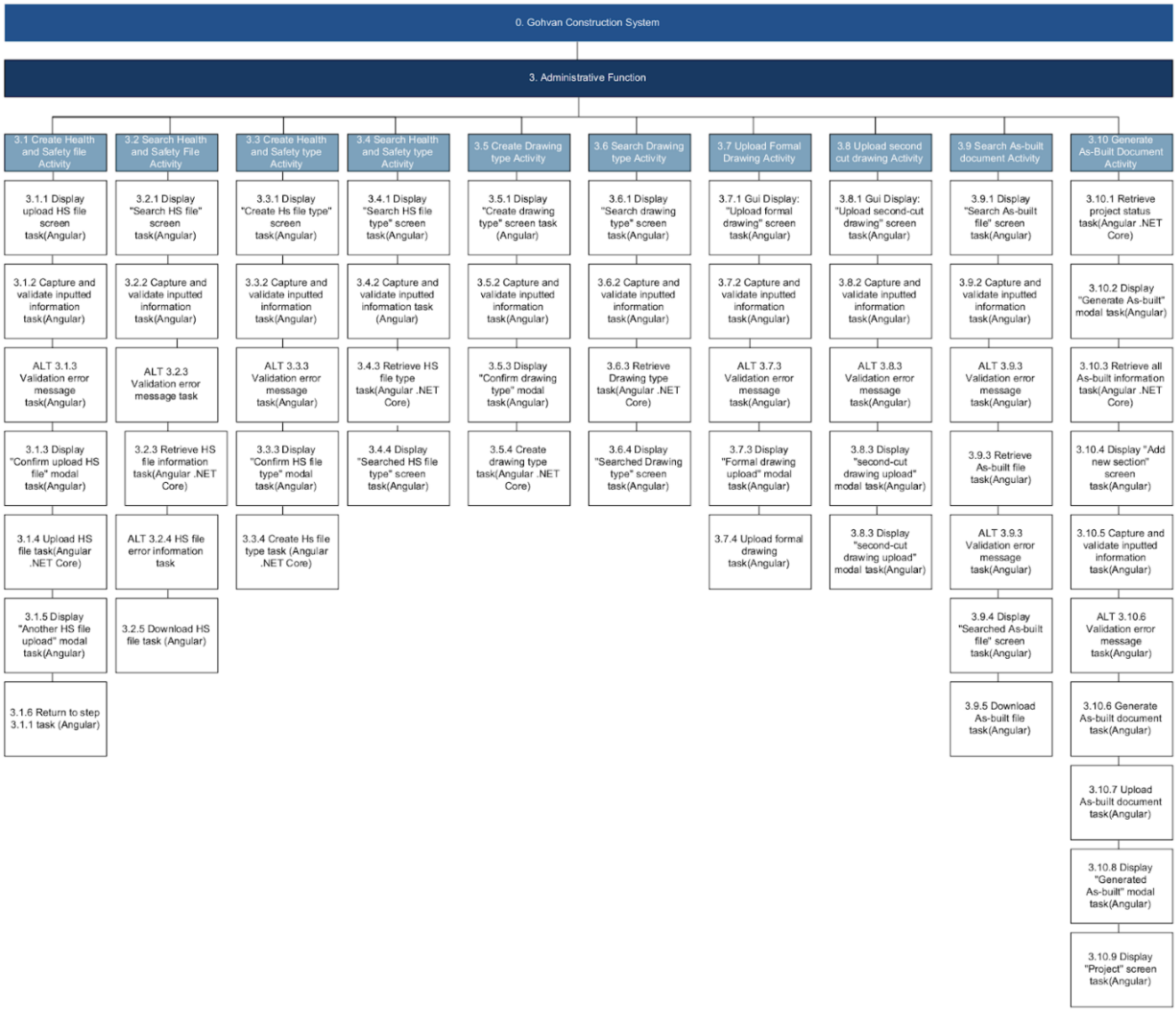


Figure 35: Decomposition diagram - 3. Administrative Function

2.2.4 Construction Subsystem

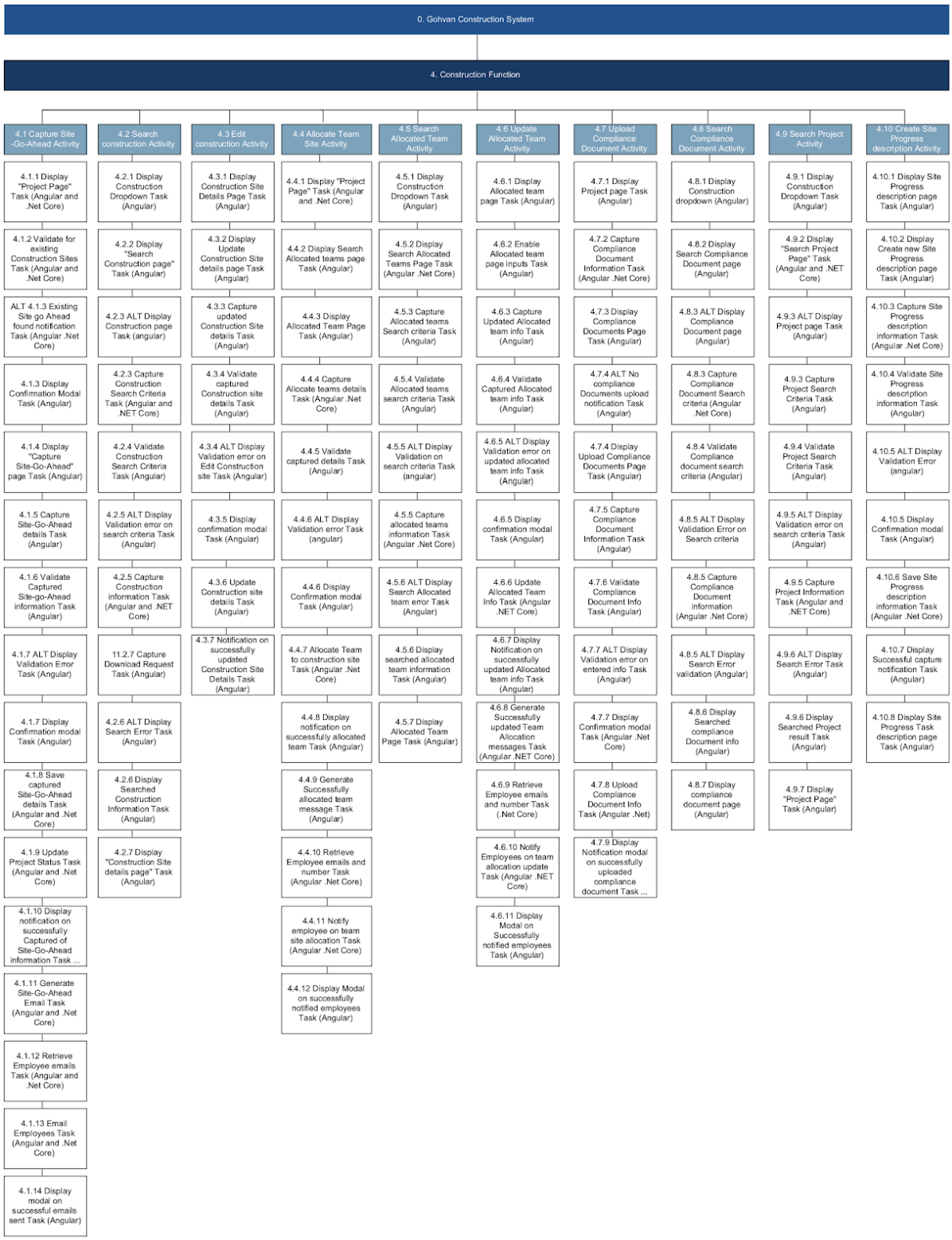


Figure 36: Decomposition diagram - 4. Construction Function 4.1-4.10

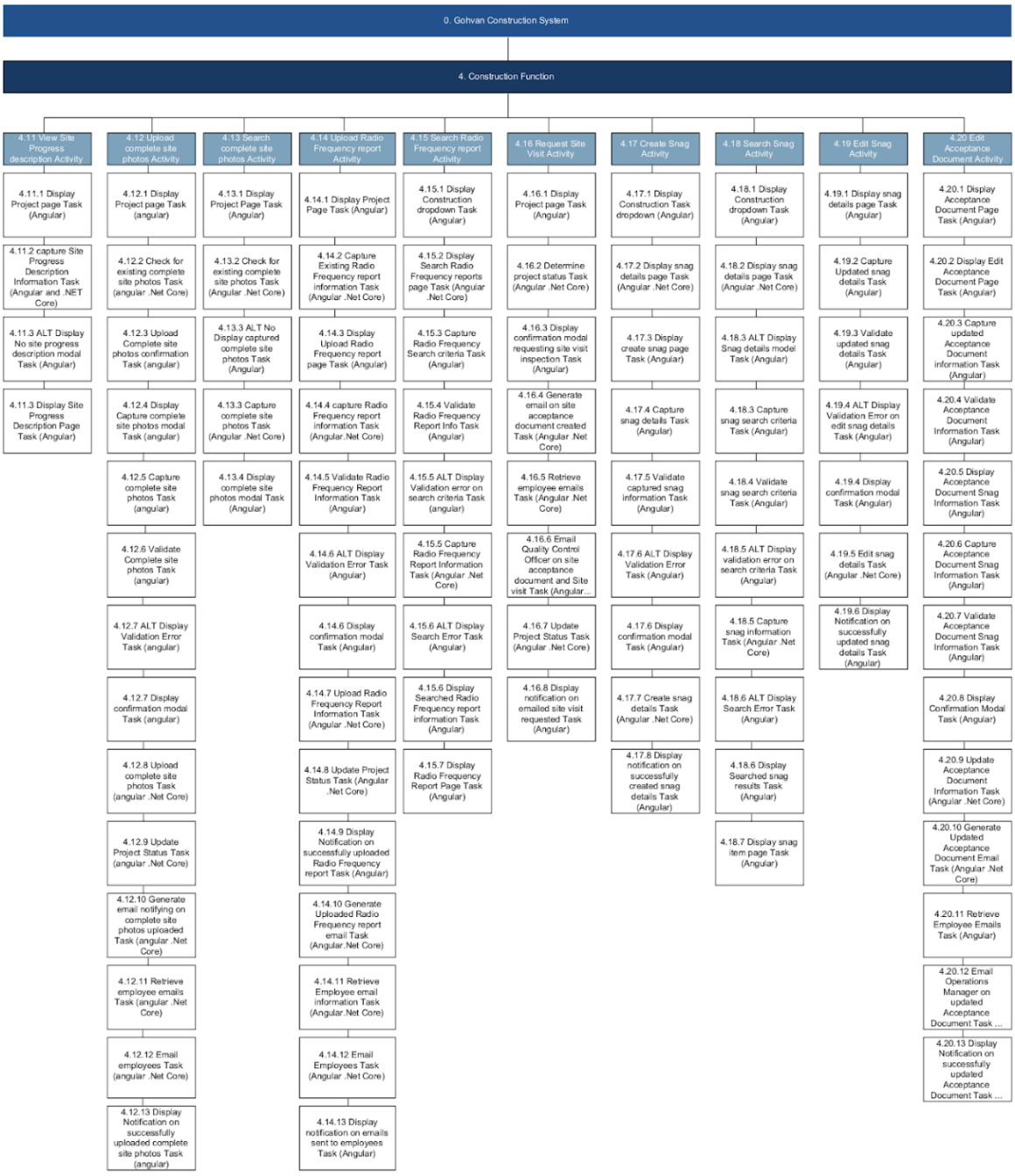


Figure 37: Decomposition diagram - 4. Construction Function 4.11-4.20



Figure 38: Decomposition diagram - 4. Construction Function 4.21-4.29

2.2.5 Invoicing Subsystem

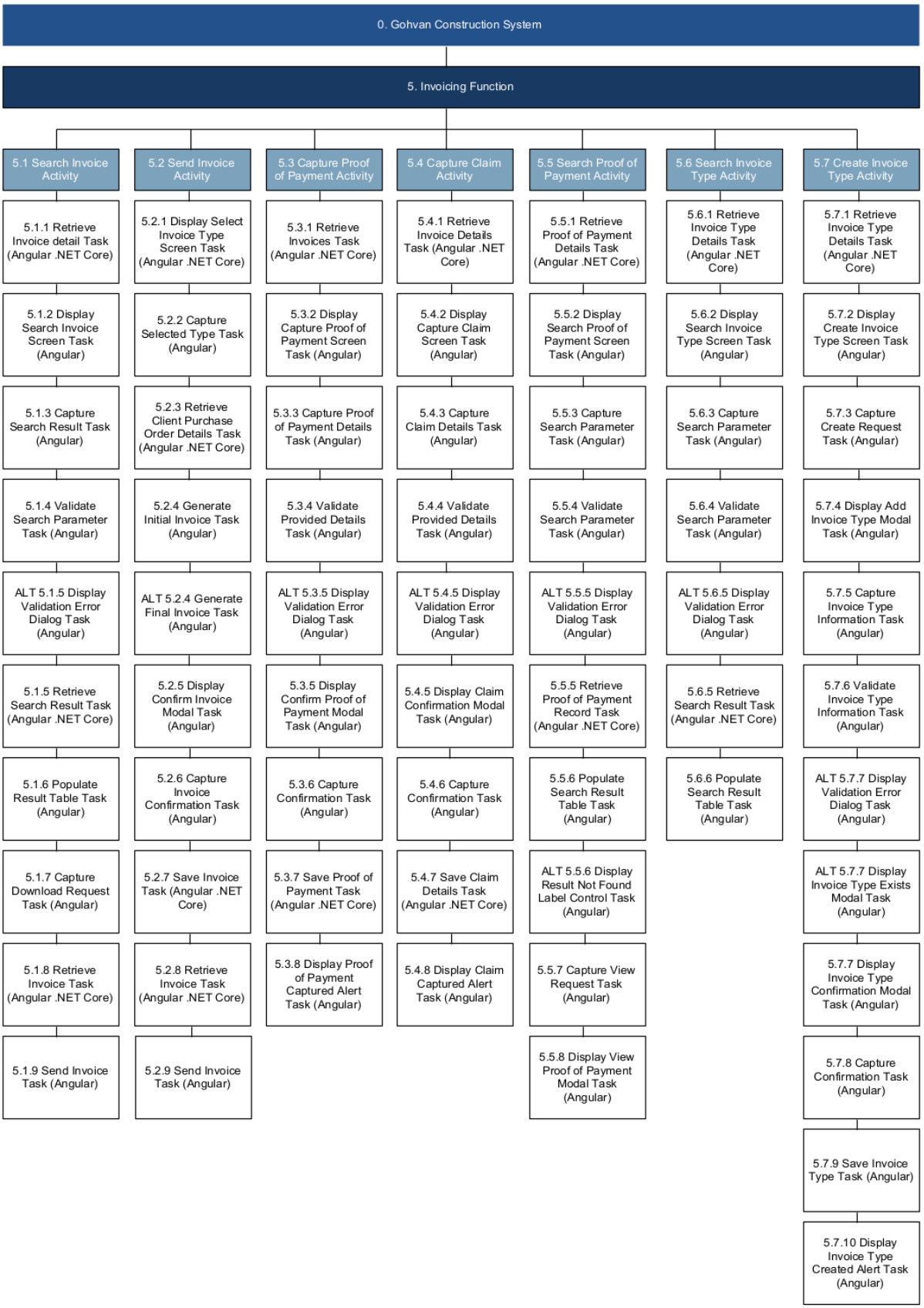


Figure 39: Decomposition diagram - 5. Invoicing Function

2.2.6 Human Resource Subsystem

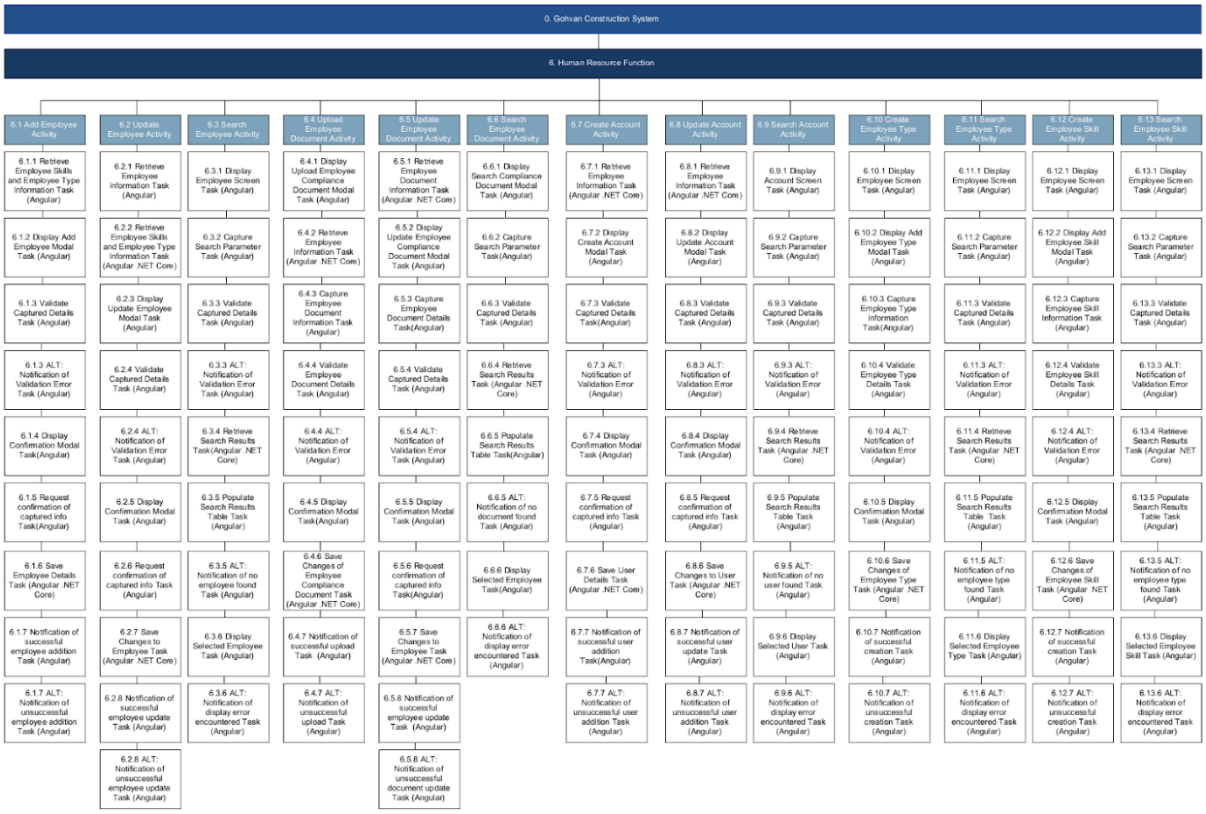


Figure 40: Decomposition diagram - 6. Human Resource Function

2.2.7 Teams Subsystem

2.2.8 Supplier Order Subsystem

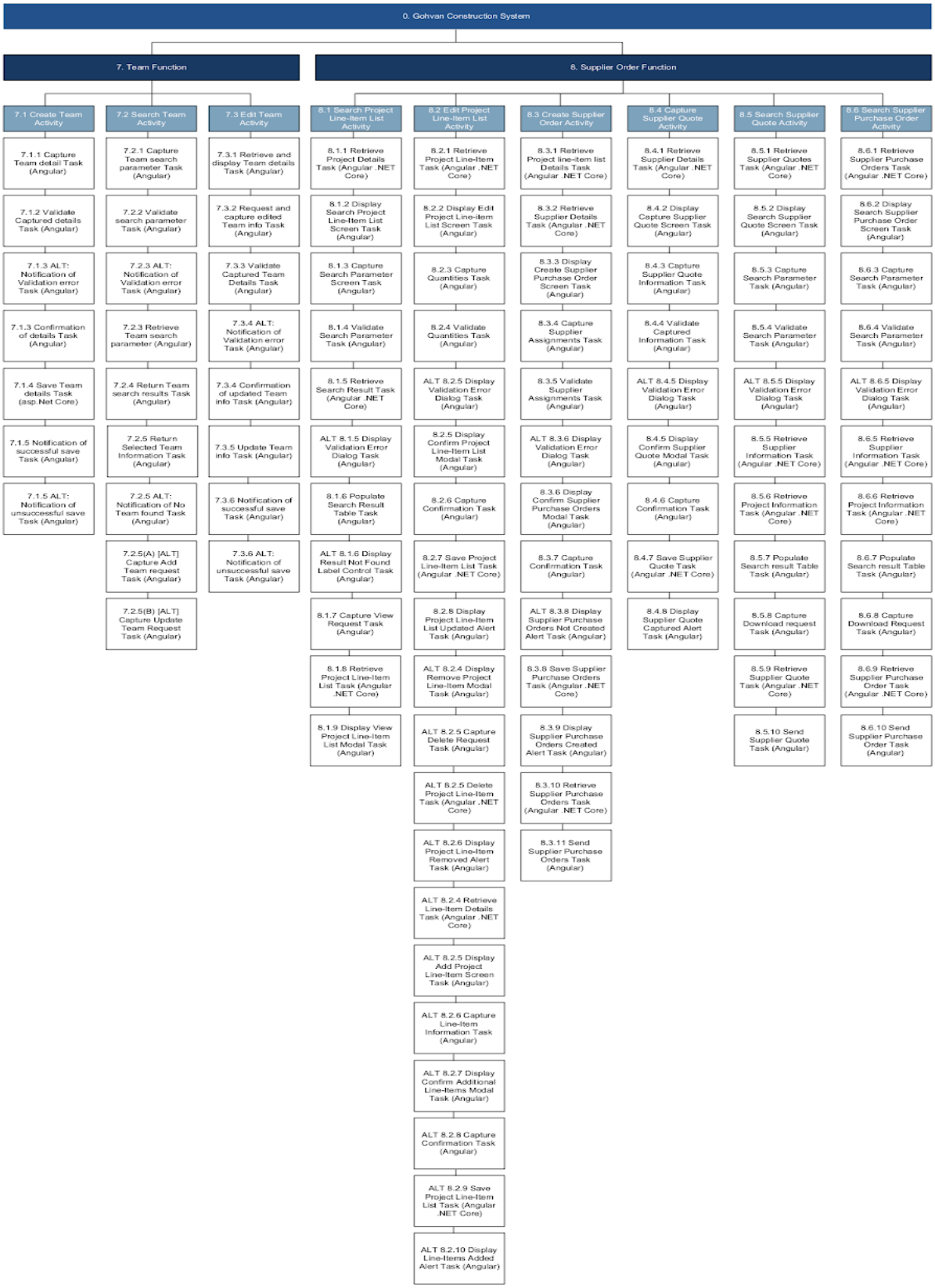


Figure 41: Decomposition diagram - 7. Team Function and 8. Supplier Order Function

2.2.9 Supplier Subsystem

2.2.10 Inventory Management Subsystem

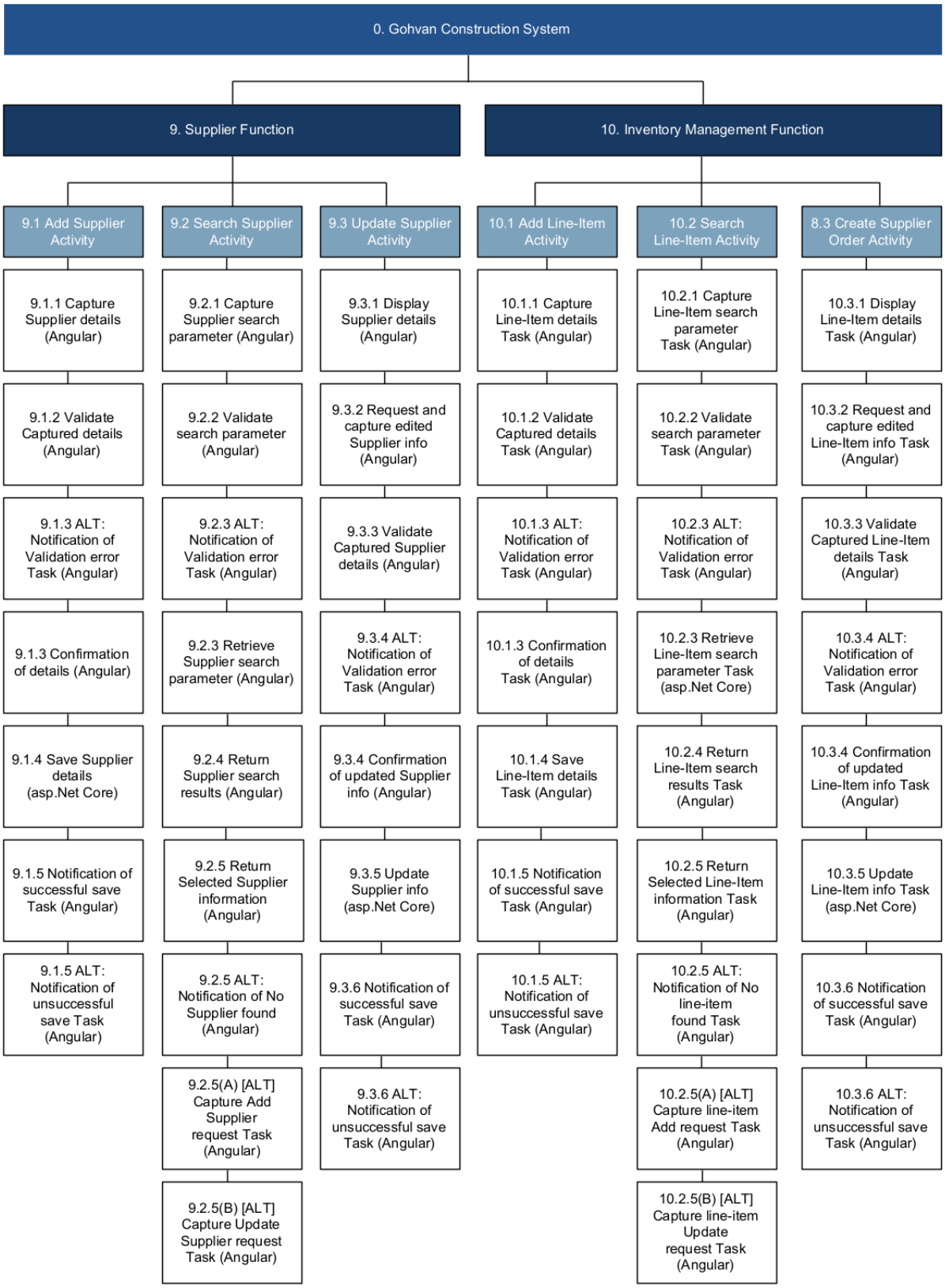


Figure 42: Decomposition diagram - 9. Supplier Function and 10. Inventory Management Function

2.2.11 Report Subsystem

2.2.12 User Management Subsystem

2.2.13 Client Subsystem

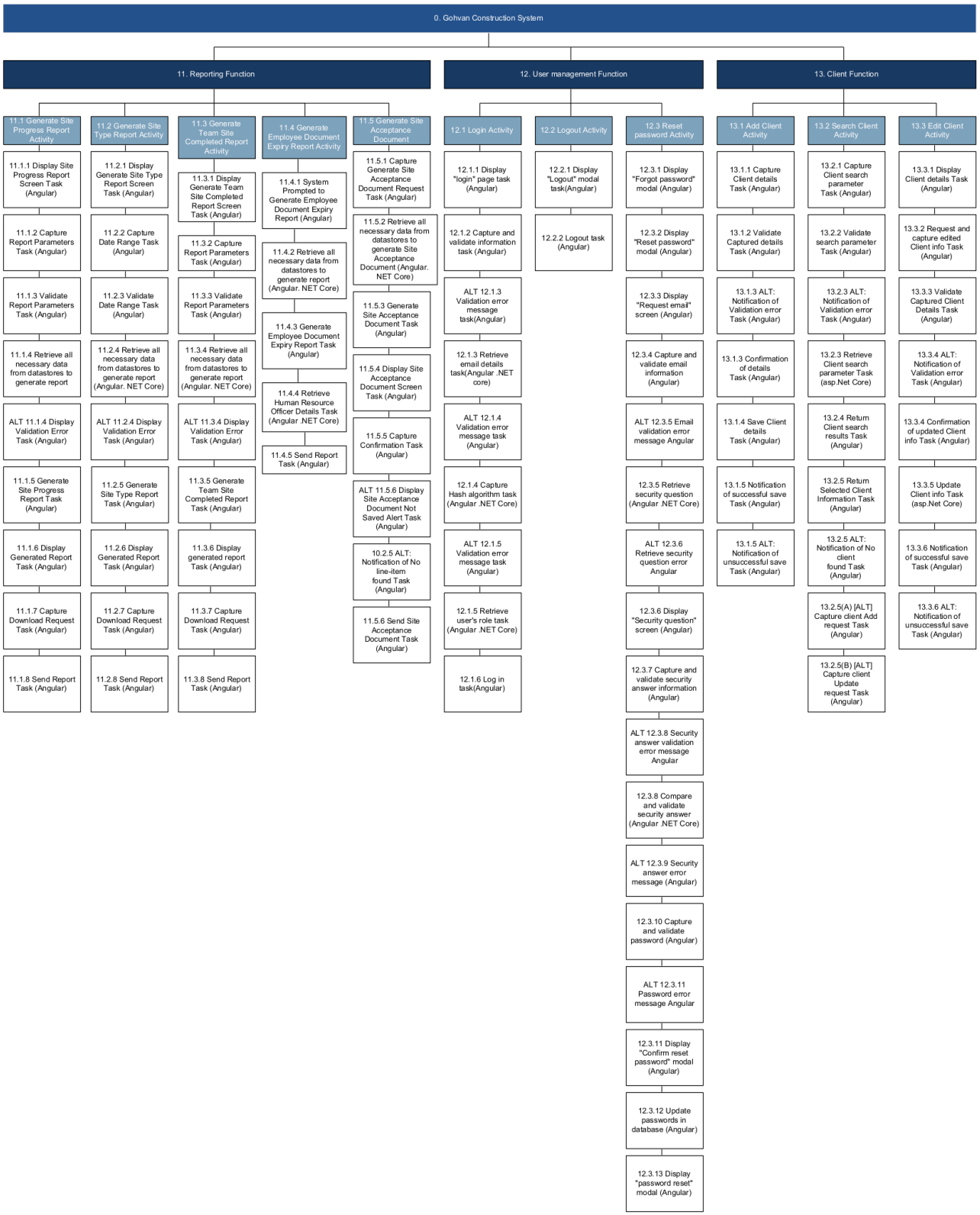


Figure 43: Decomposition diagram - 11. Reporting Function, 12. User Management Function and 13. Client Function

2.3 Physical Data Flow Diagrams

2.3.1 High Level Date flow diagrams

2.3.1.1 Establishment Subsystem

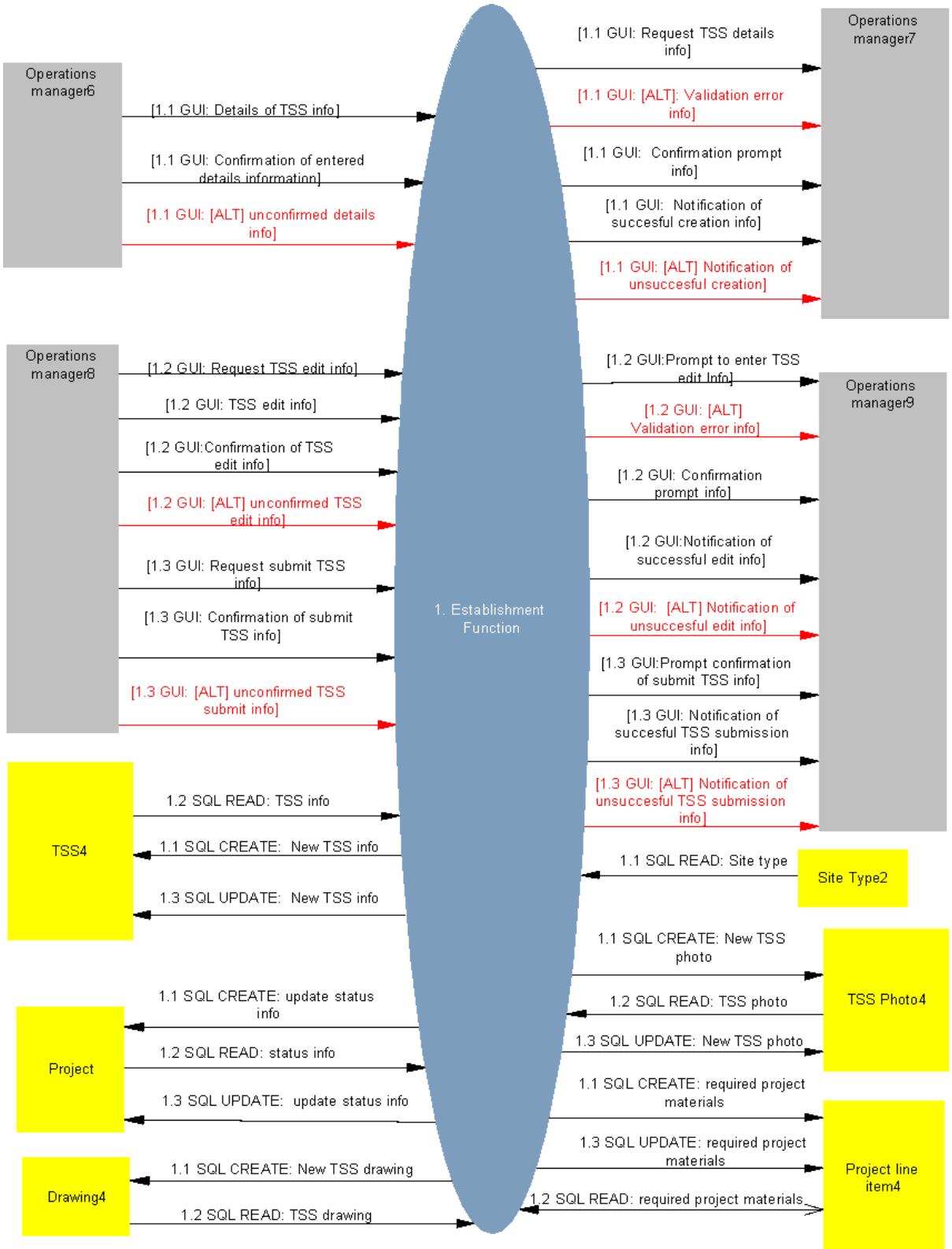


Figure 44: High level data flow diagram Establishment Subsystem

2.3.1.2 Quotation Subsystem

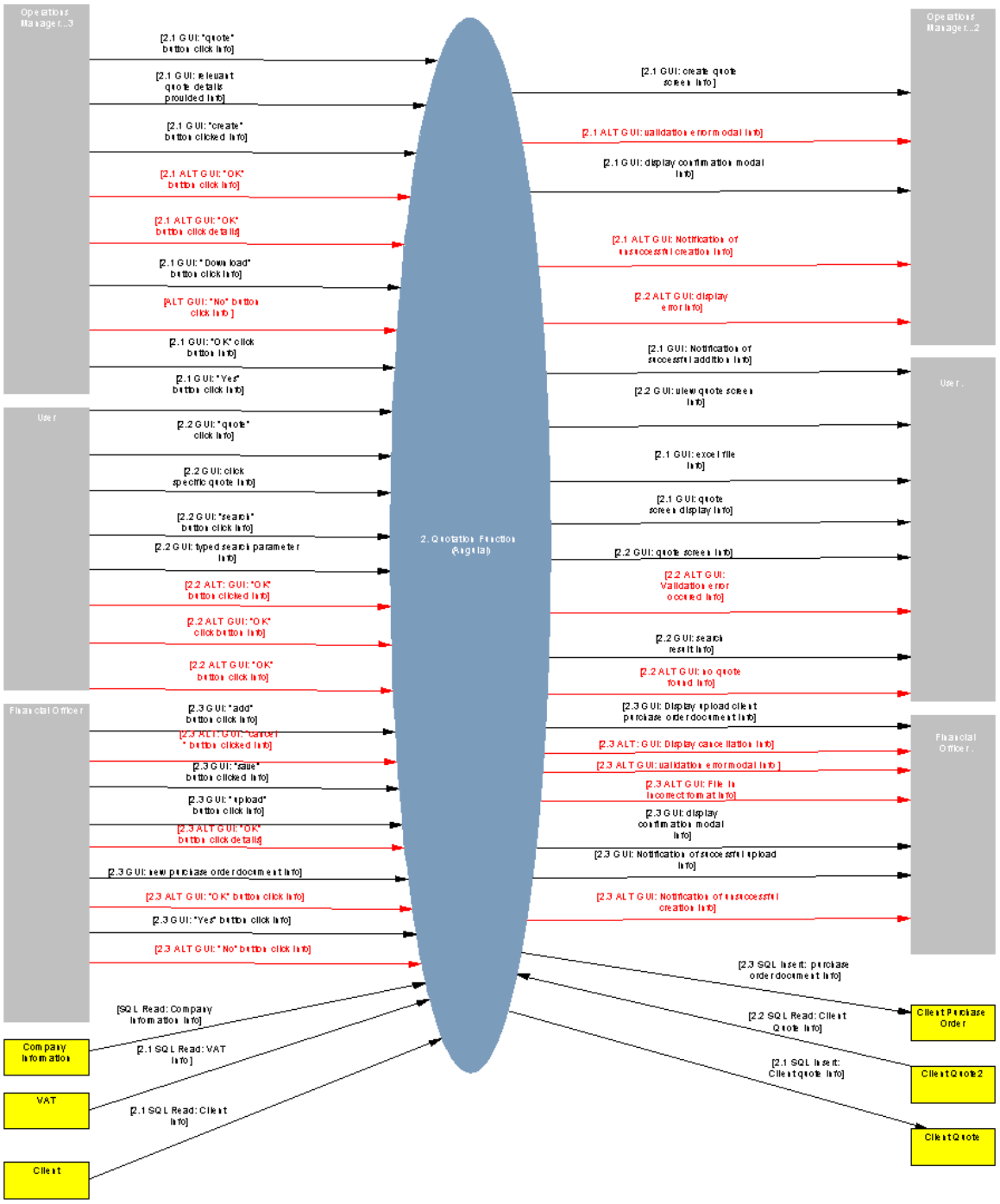


Figure 45: High level data flow diagram Quotation Subsystem

2.3.1.3 Administration Subsystem

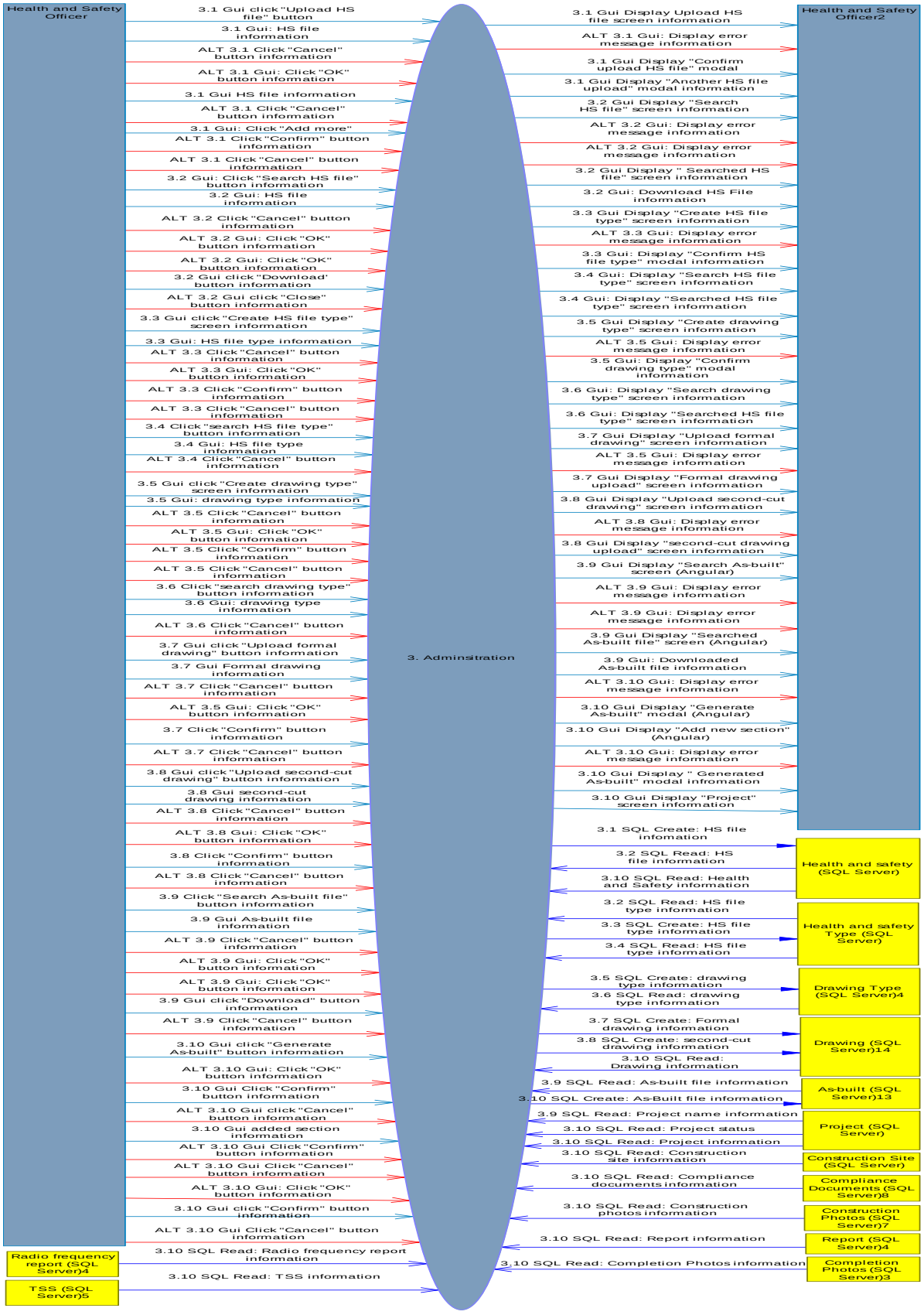


Figure 46: High level data flow diagram Administration Subsystem

2.3.1.4 Construction Subsystem

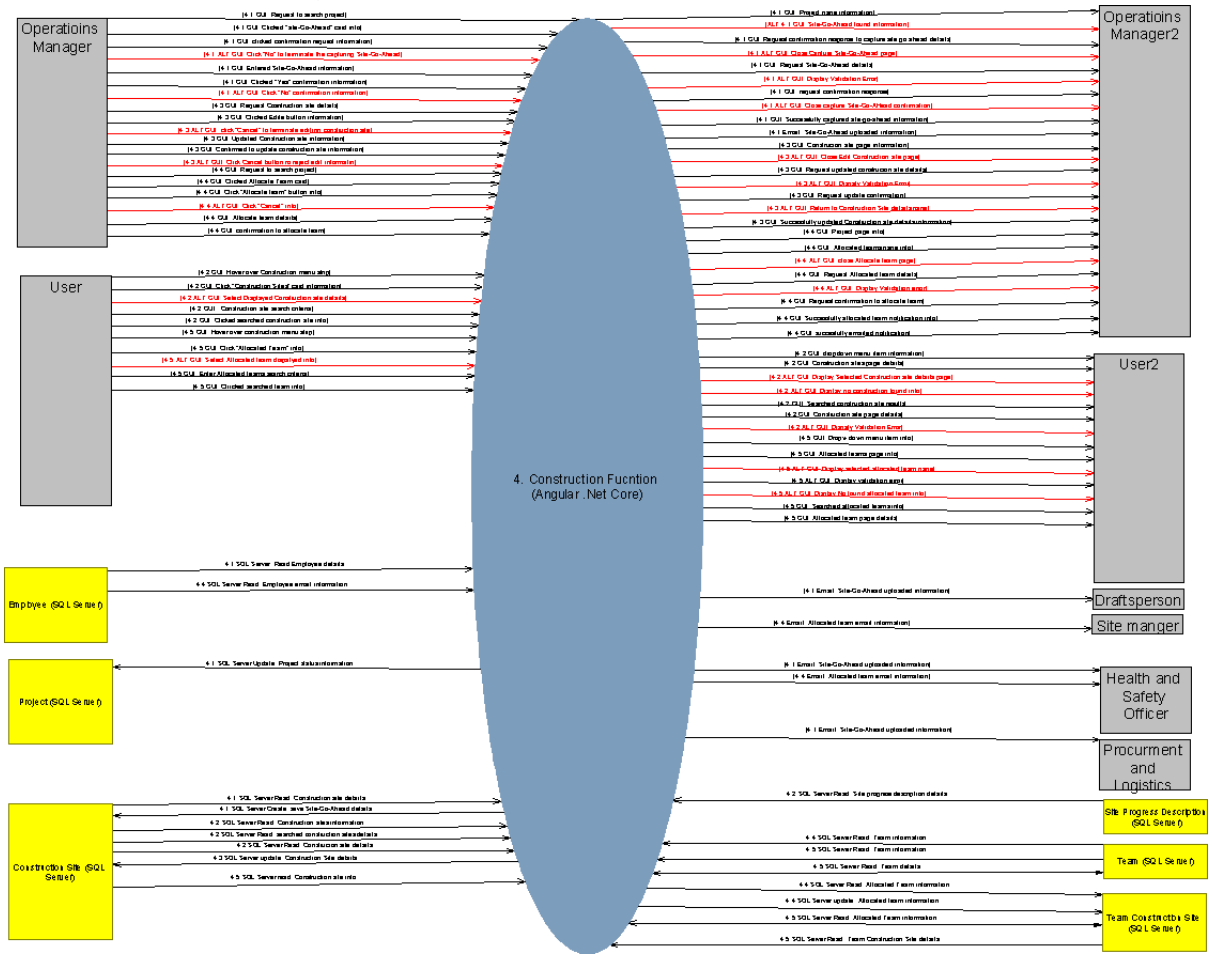


Figure 47: High level data flow diagram Construction Subsystem 4.1-4.5

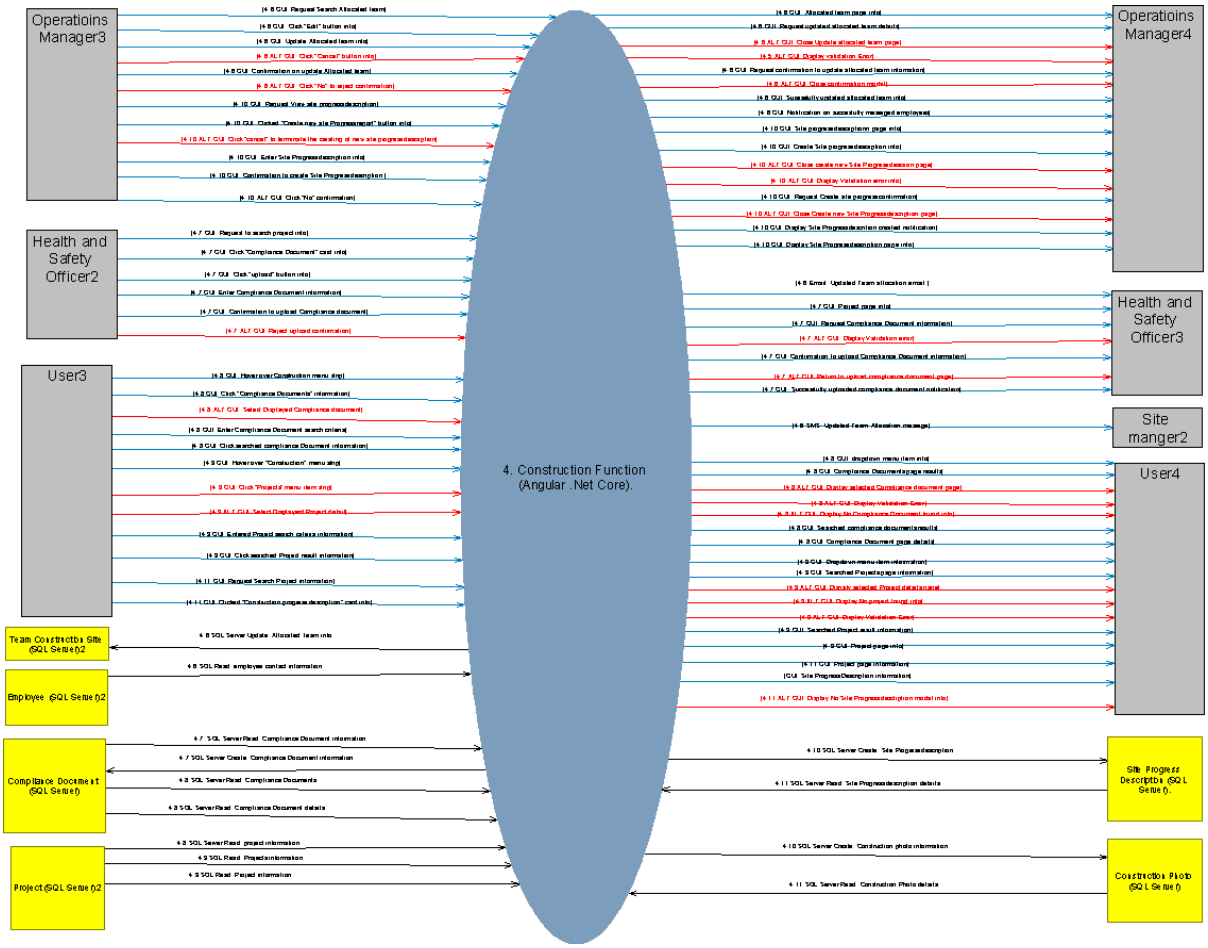


Figure 48: High level data flow diagram Construction Subsystem 4.6-4.11

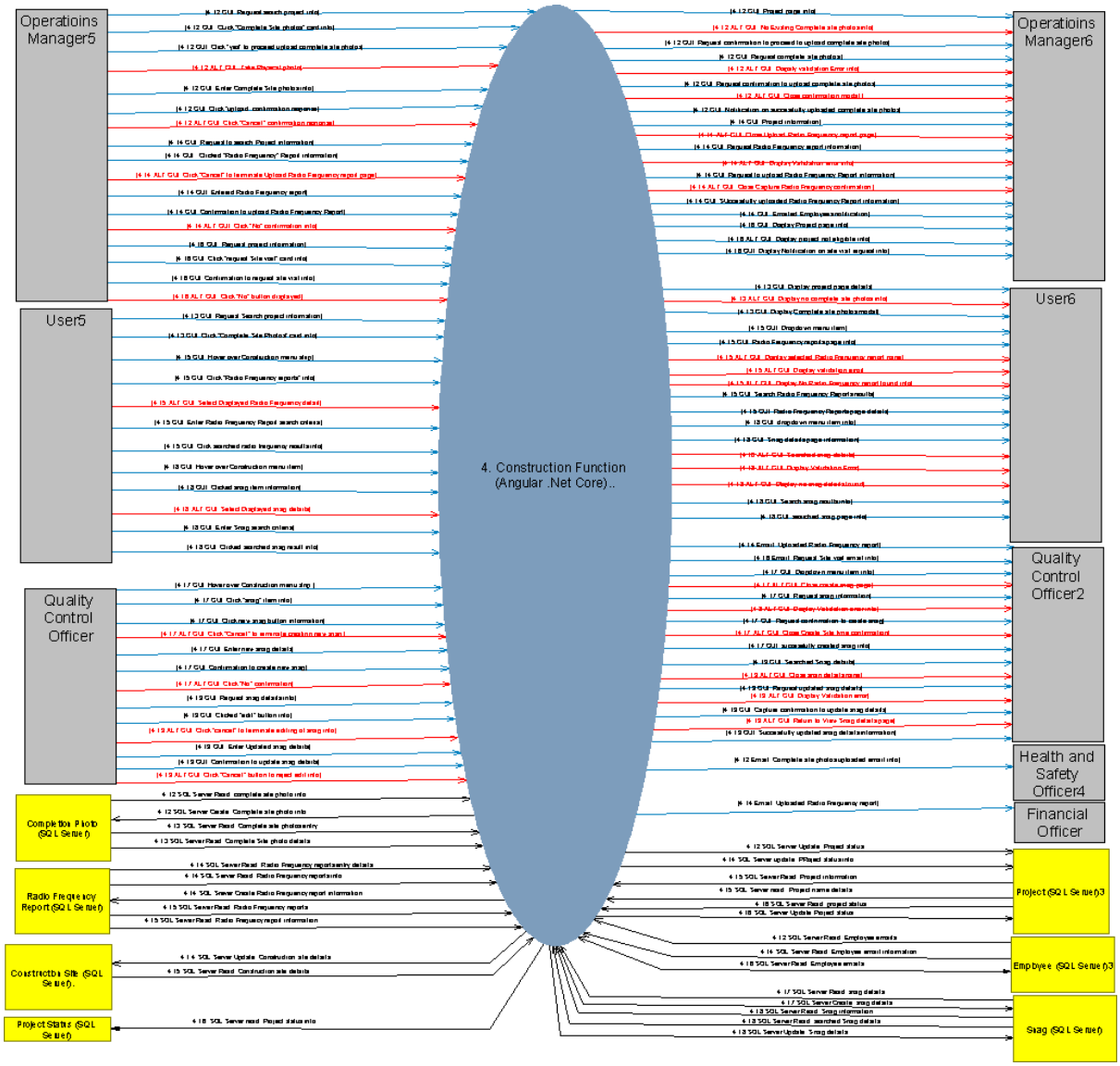


Figure 49: High level data flow diagram Construction Subsystem 4.12-4.19

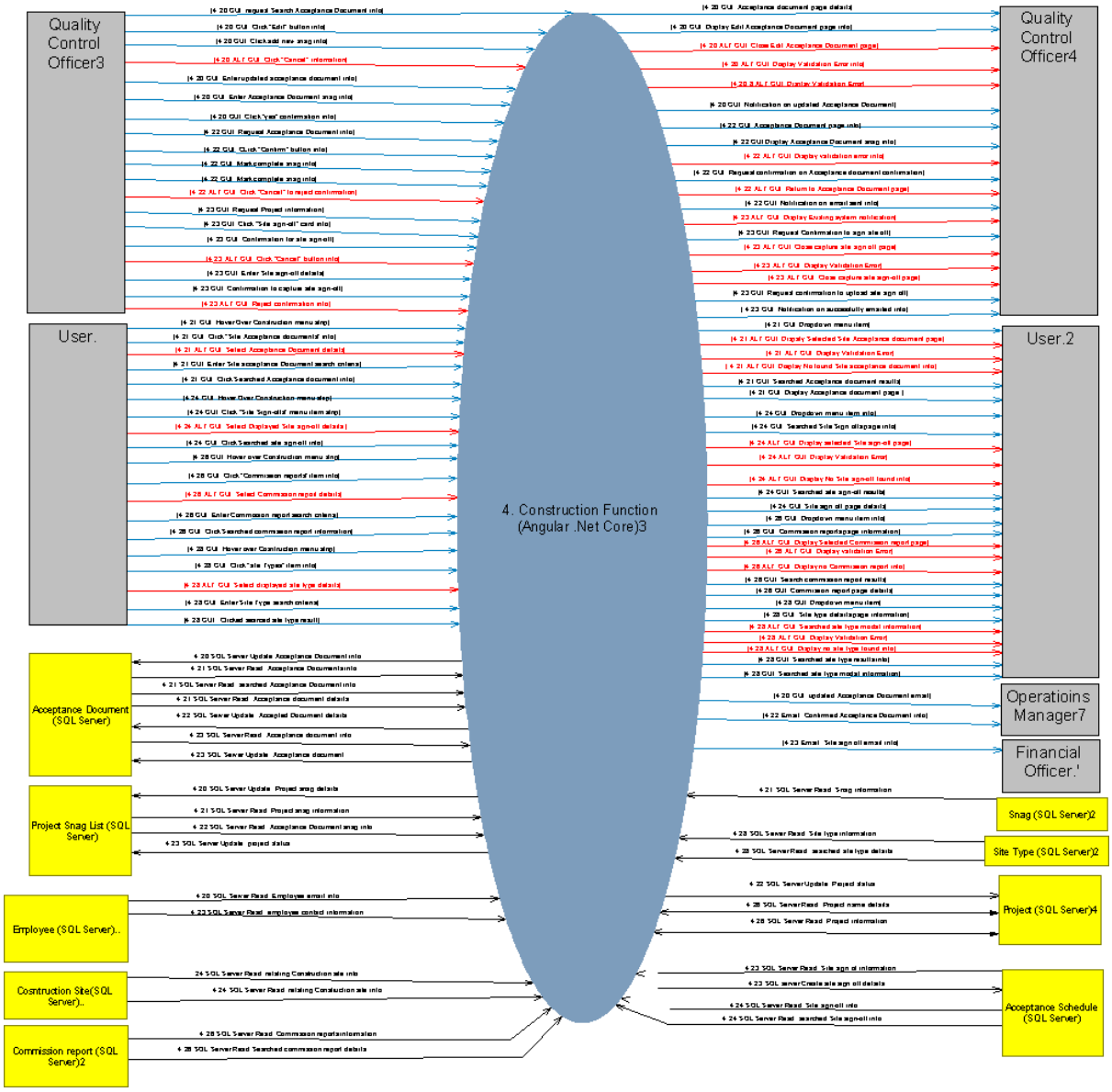


Figure 50: High level data flow diagram Construction Subsystem 4.20 -4.28

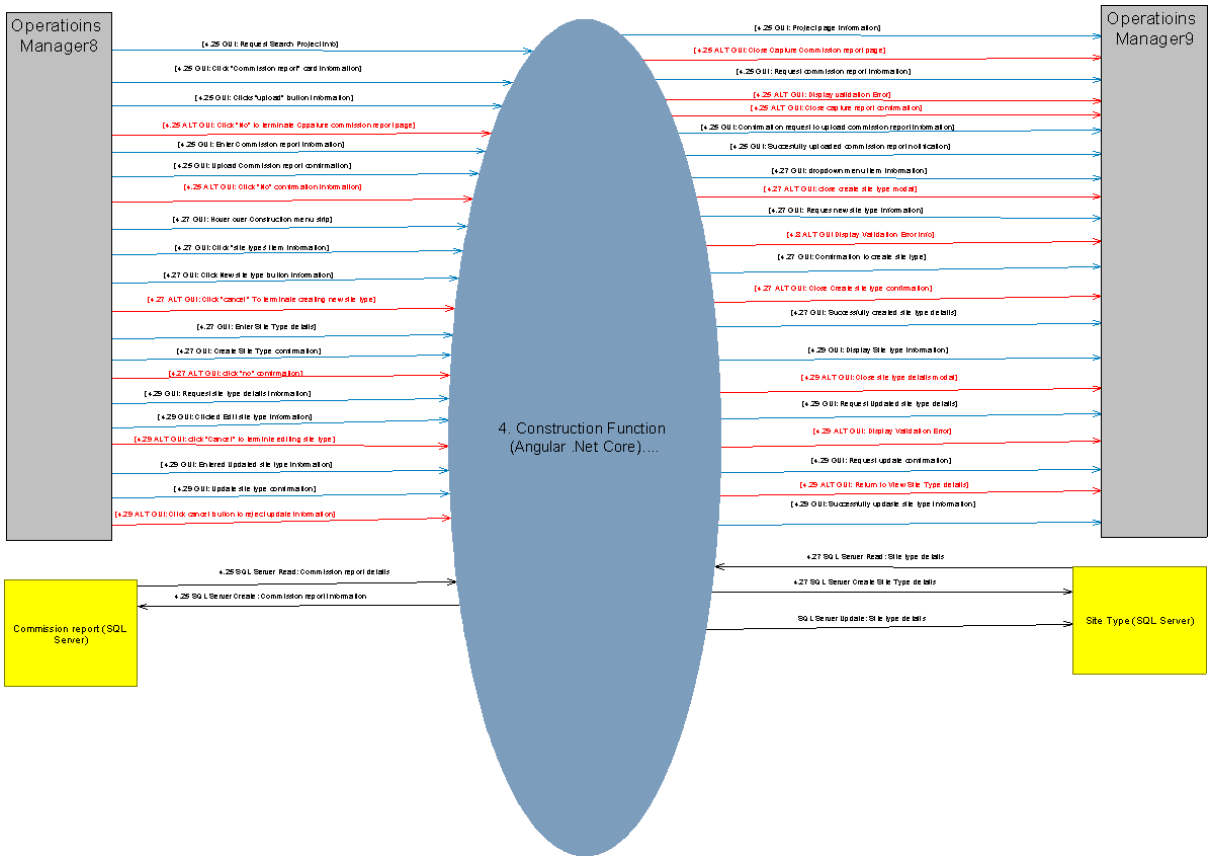


Figure 51: High level data flow diagram Construction Subsystem 4.25 -4.29

2.3.1.5 Invoicing Subsystem

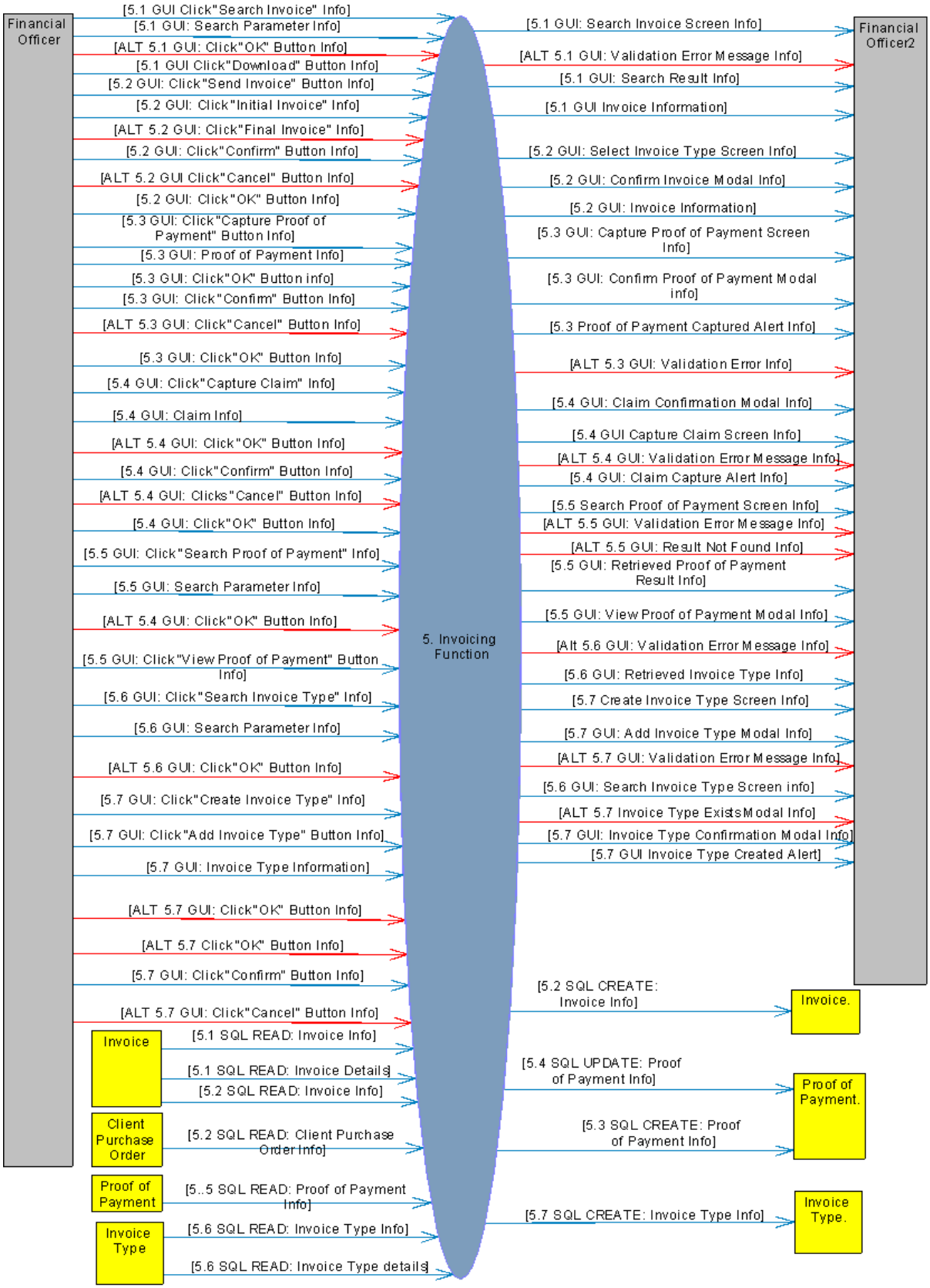


Figure 52: High level data flow diagram Invoicing Subsystem

2.3.1.6 Human Resource Subsystem

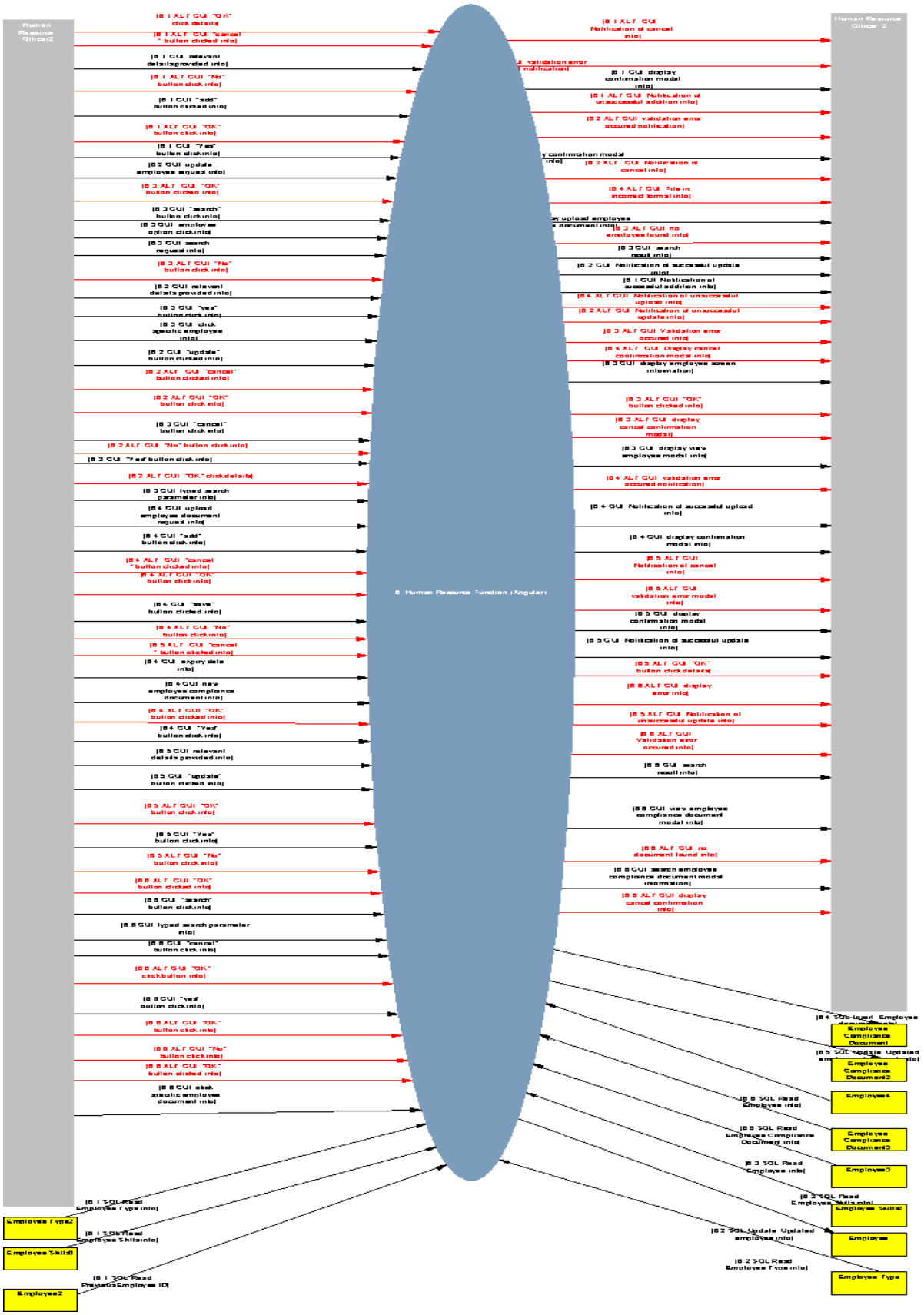


Figure 53: High level data flow diagram Human Resources Subsystem 6.1-6.6

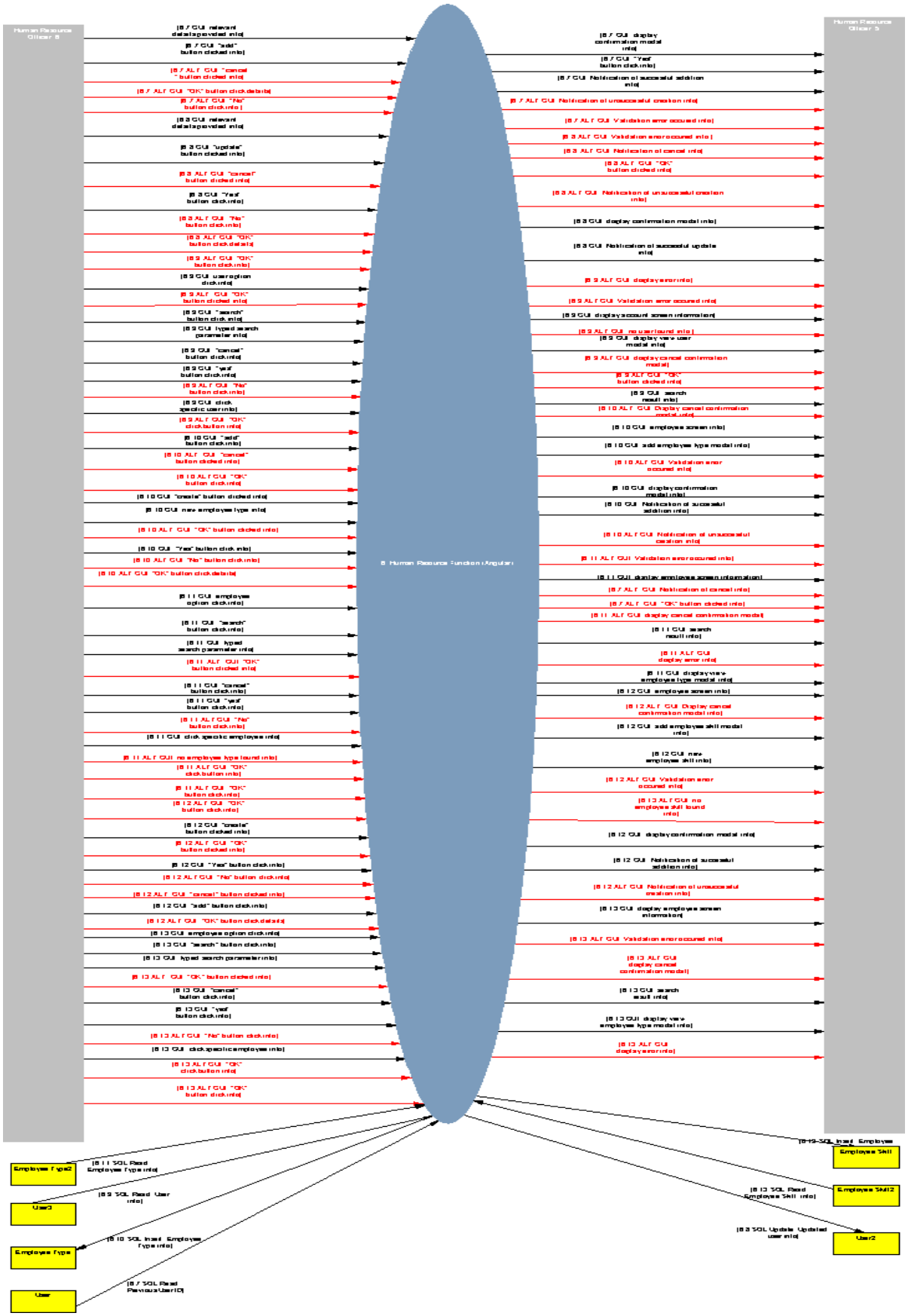


Figure 54: High level data flow diagram Human Resources Subsystem 6.7-6.13

2.3.1.7 Teams Subsystem

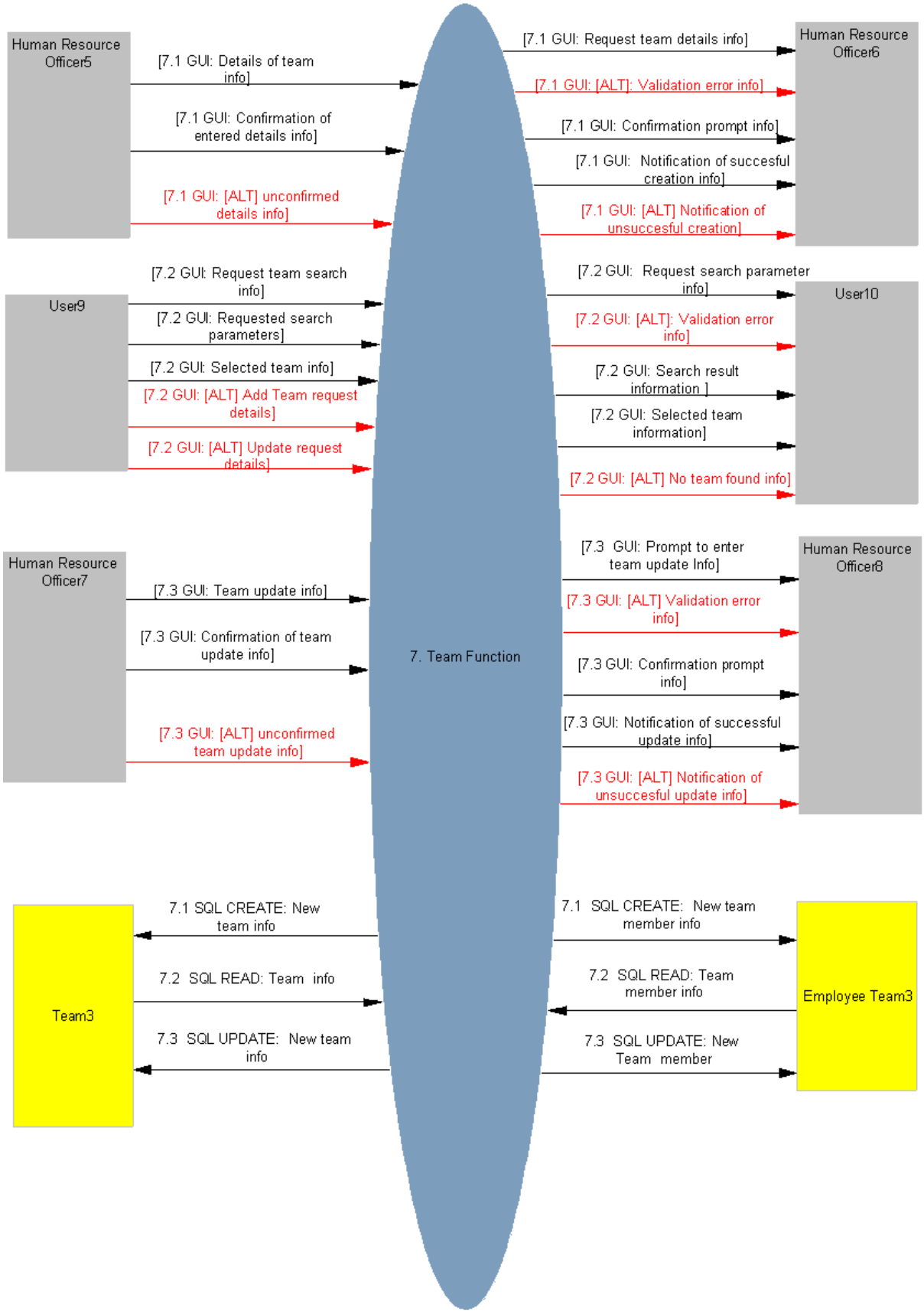


Figure 55: High level data flow diagram Teams Subsystem

2.3.1.8 Supplier Order Subsystem

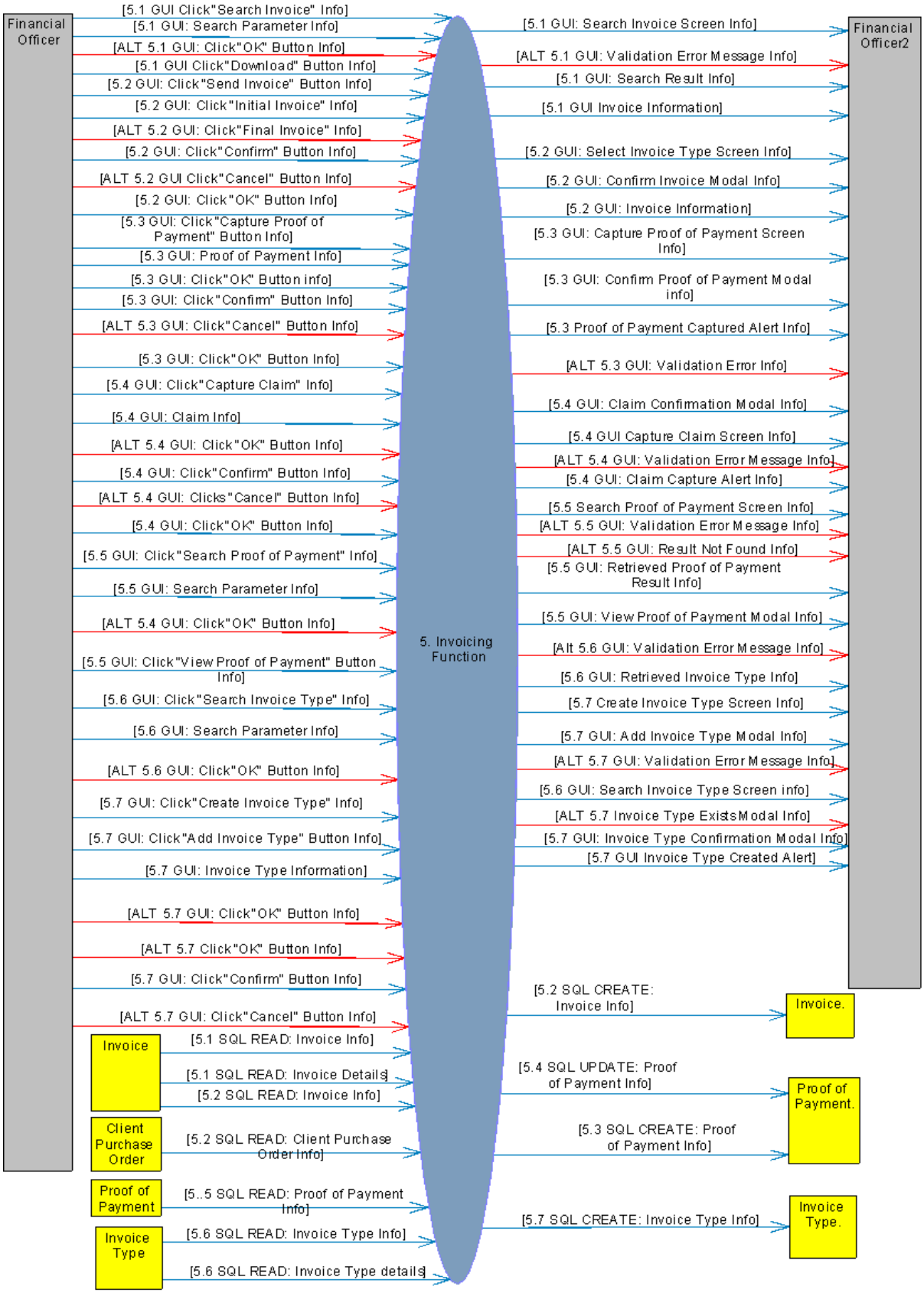


Figure 56: High level data flow diagram Supplier order Subsystem

2.3.1.9 Supplier Subsystem

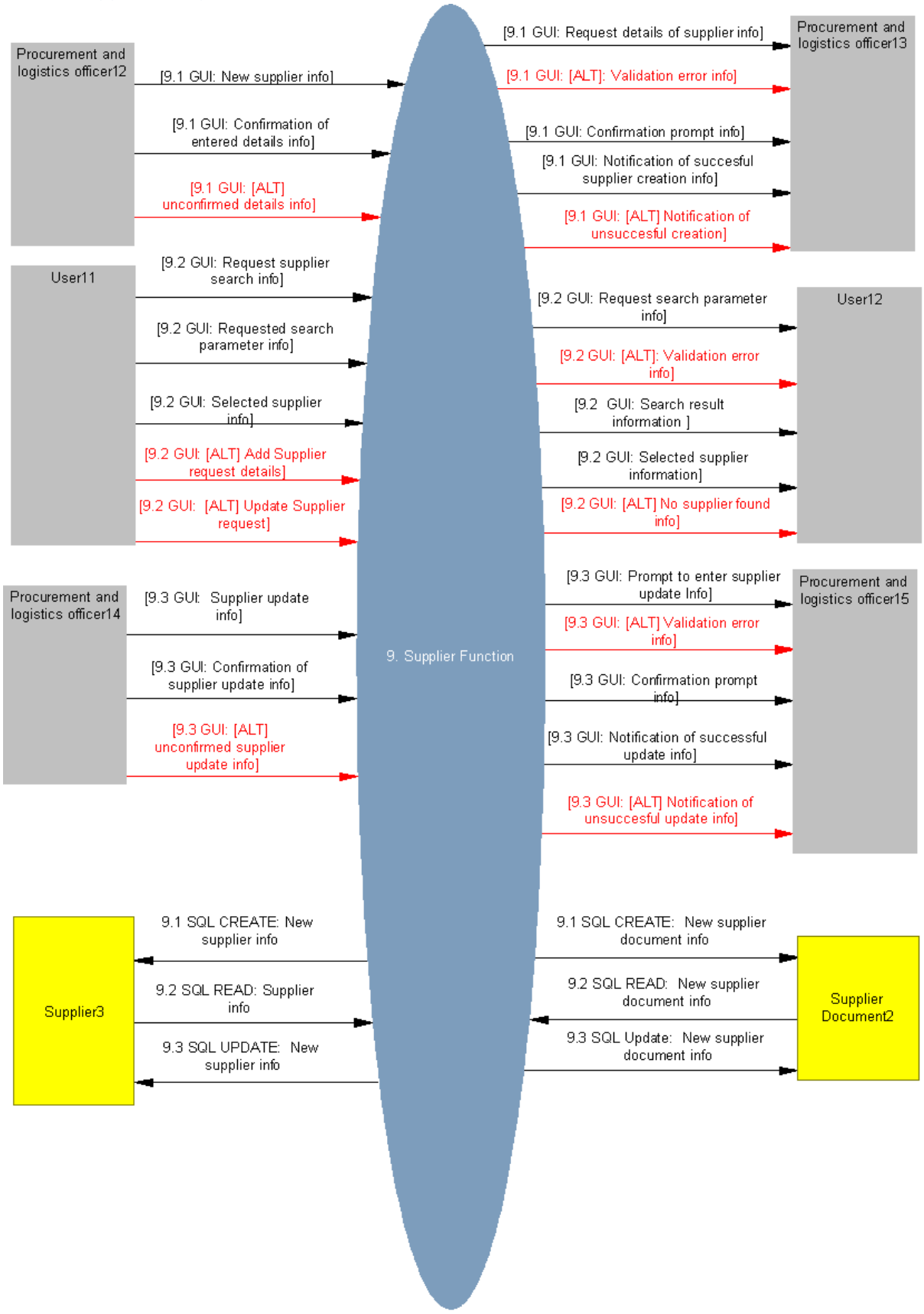


Figure 57: High level data flow diagram Supplier Subsystem

2.3.1.10 Inventory Management Subsystem

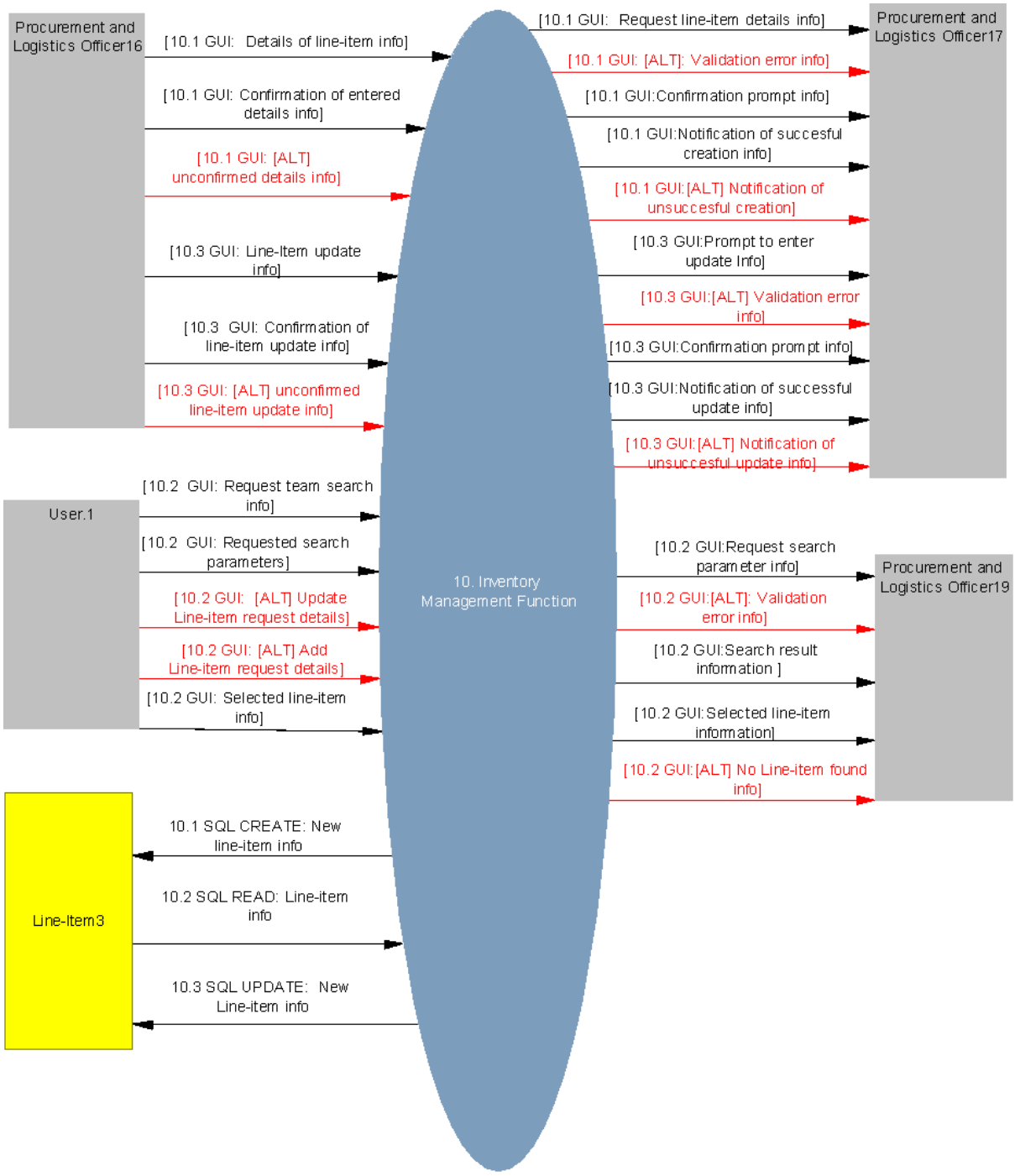


Figure 58: High level data flow diagram Inventory Management Subsystem

2.3.1.11 Report Subsystem

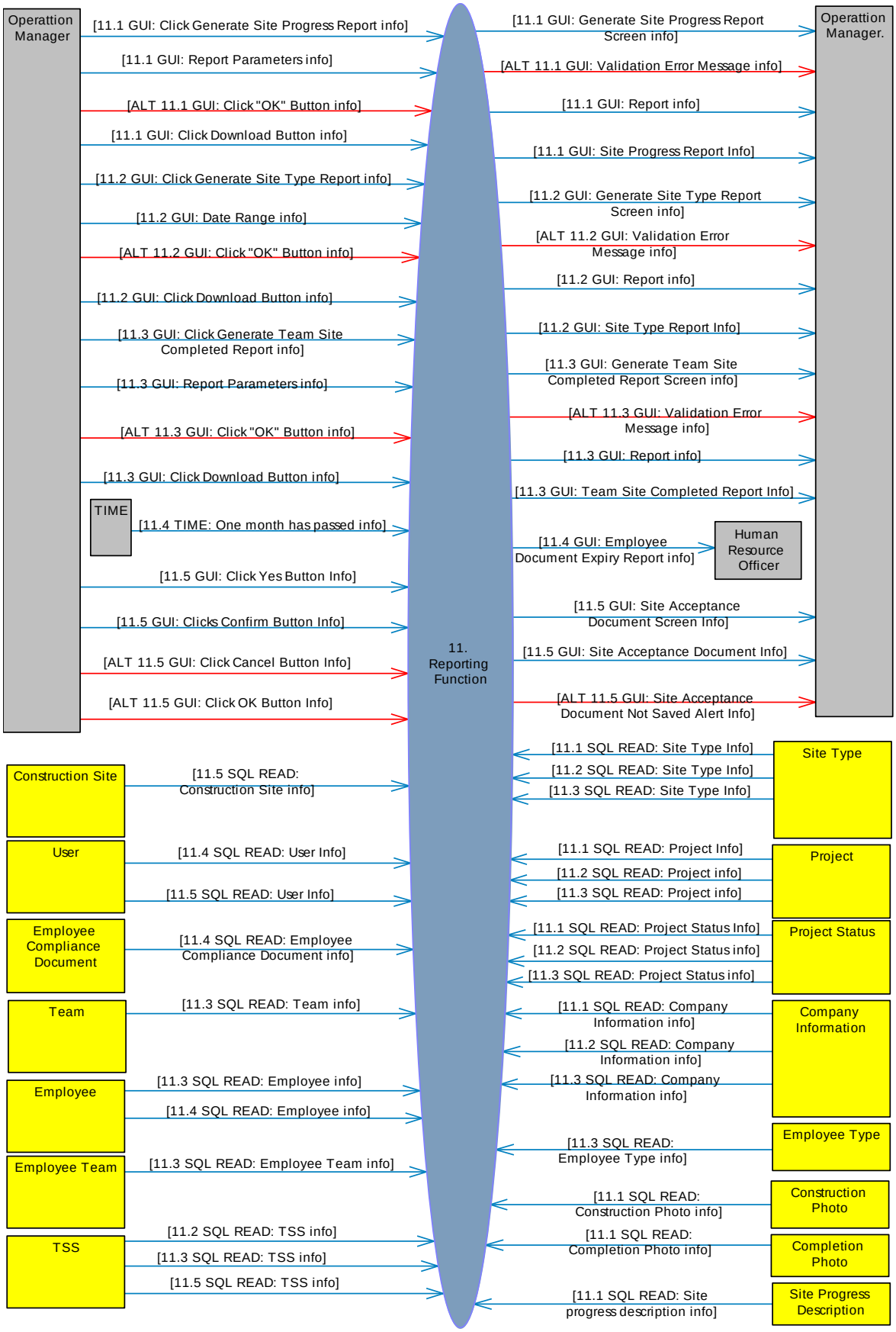


Figure 59: High level data flow diagram Report Subsystem

2.3.1.12 User Management Subsystem

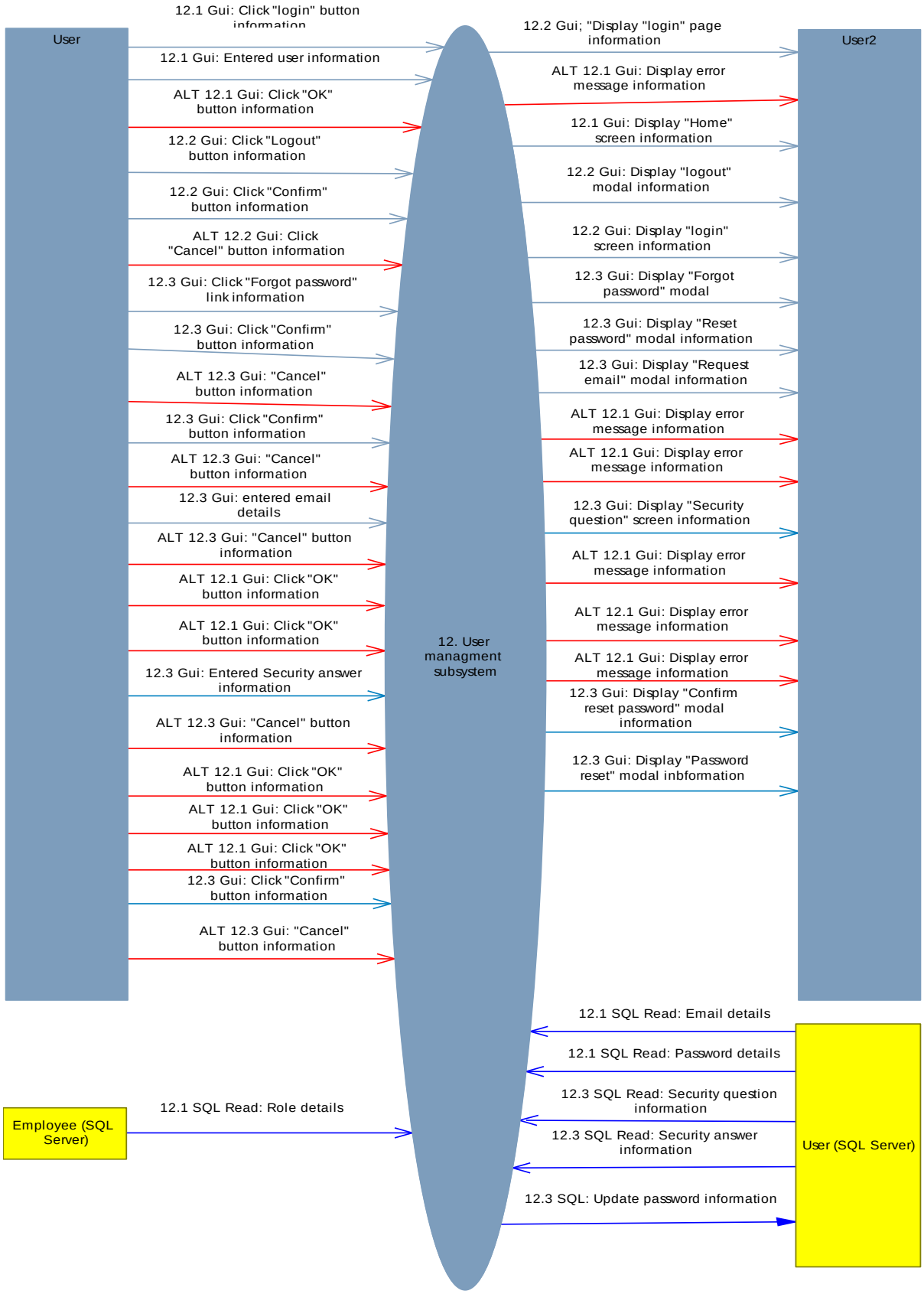


Figure 60: High level data flow diagram User Management Subsystem

2.3.1.13 Client Subsystem

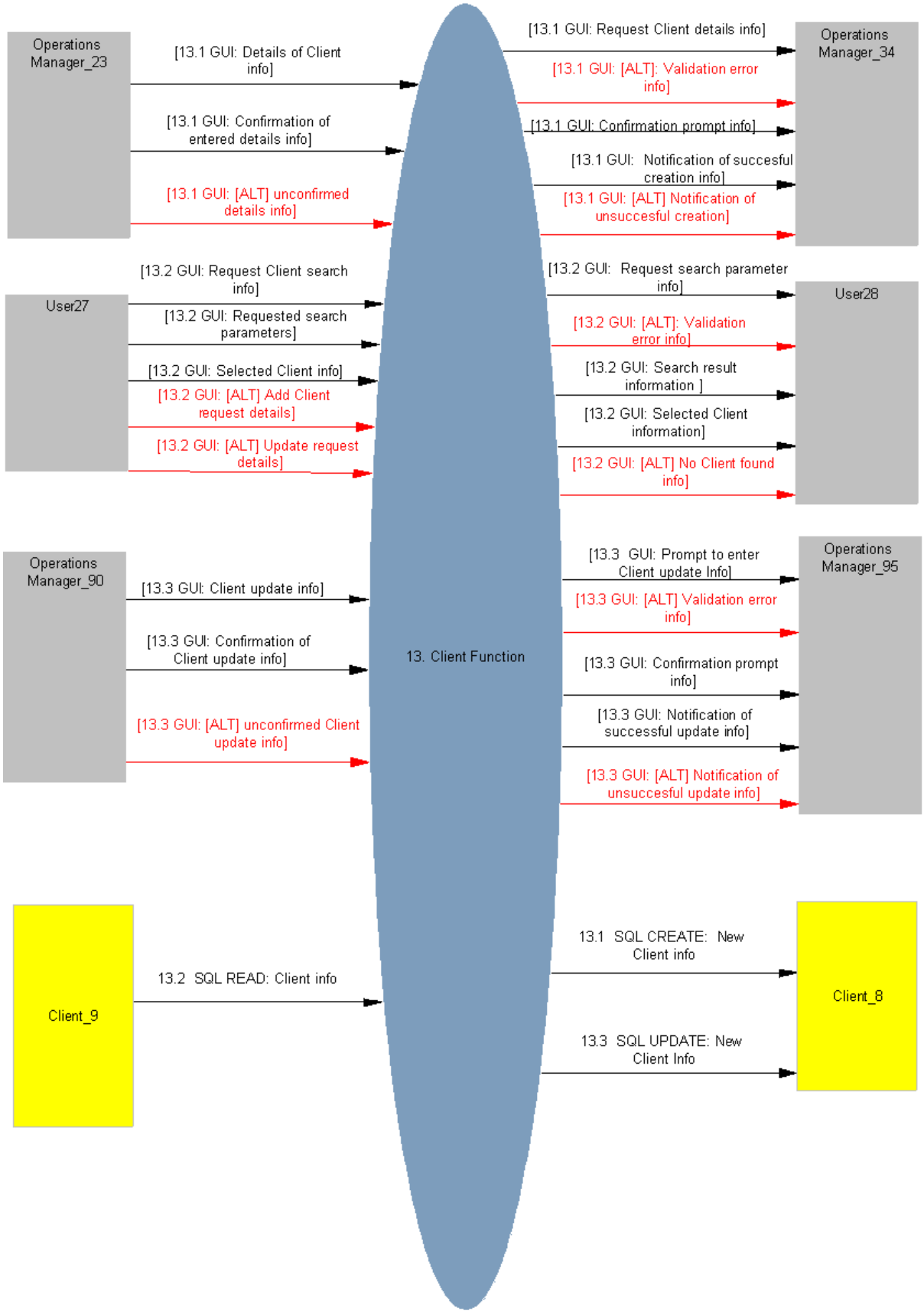


Figure 61: High level data flow diagram Client Subsystem

2.3.2 Mid-Level Data flow diagrams

2.3.2.1. Establishment Subsystem

2.3.2.1.1 Create TSS

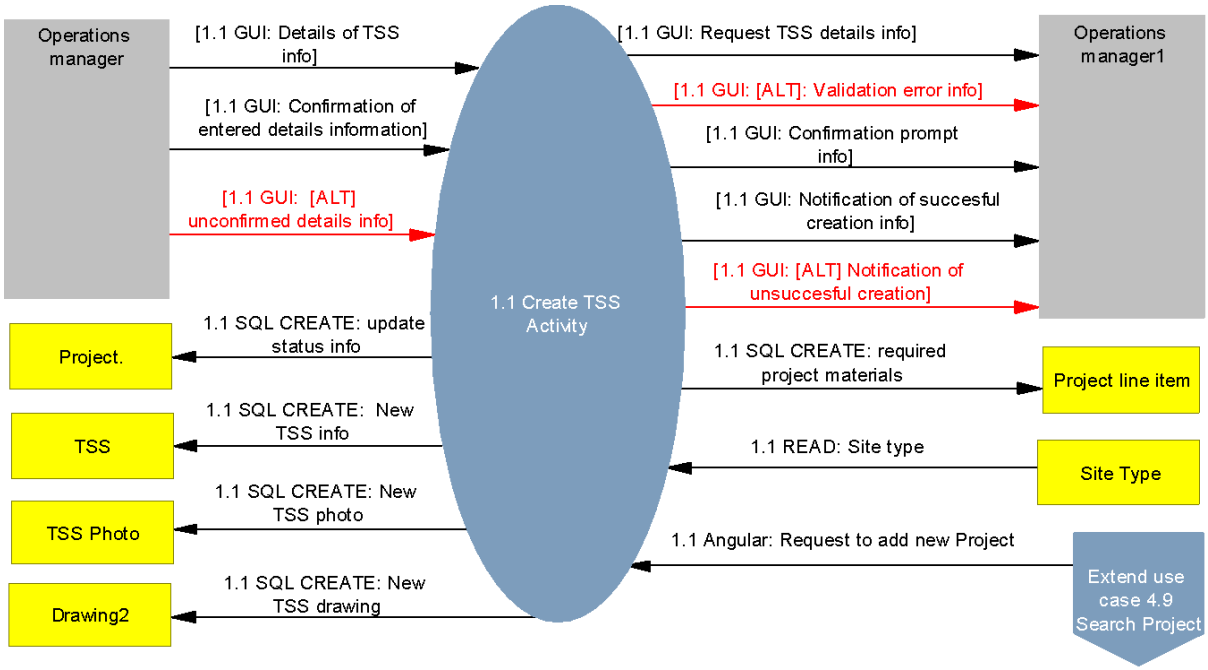


Figure 62: Middle level diagram 1.1 Create TSS

2.3.2.1.2 Edit TSS

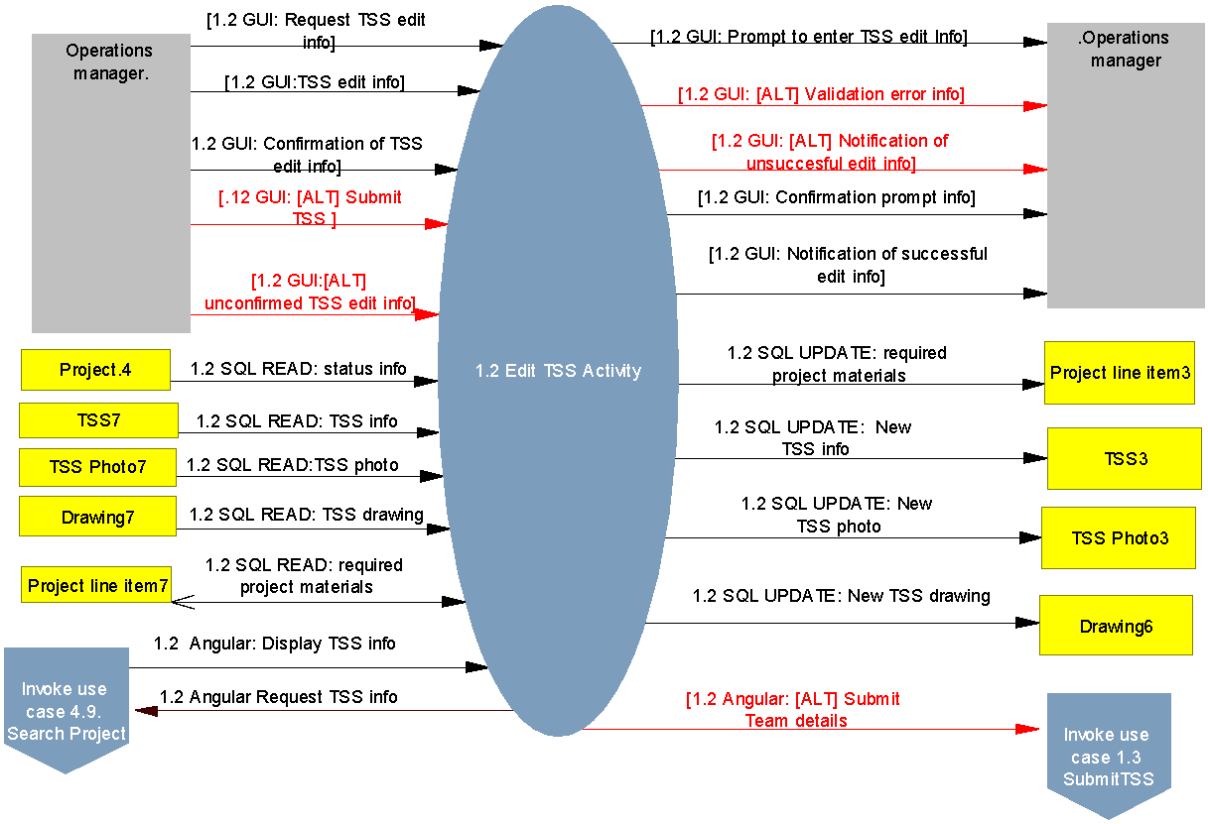


Figure 63: Middle level diagram 1.2 Edit TSS

2.3.2.1.3 Submit TSS Report

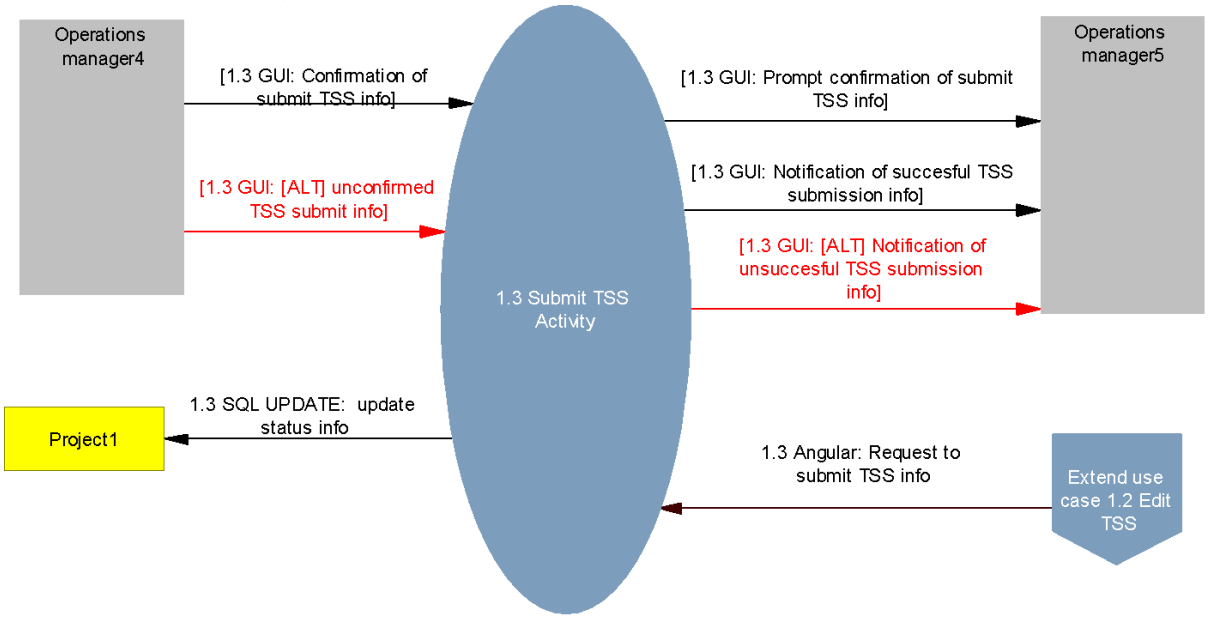


Figure 64: Middle level diagram 1.3 Submit TSS

2.3.2.2. Quotation Subsystem

2.3.2.2.1 Create Quote

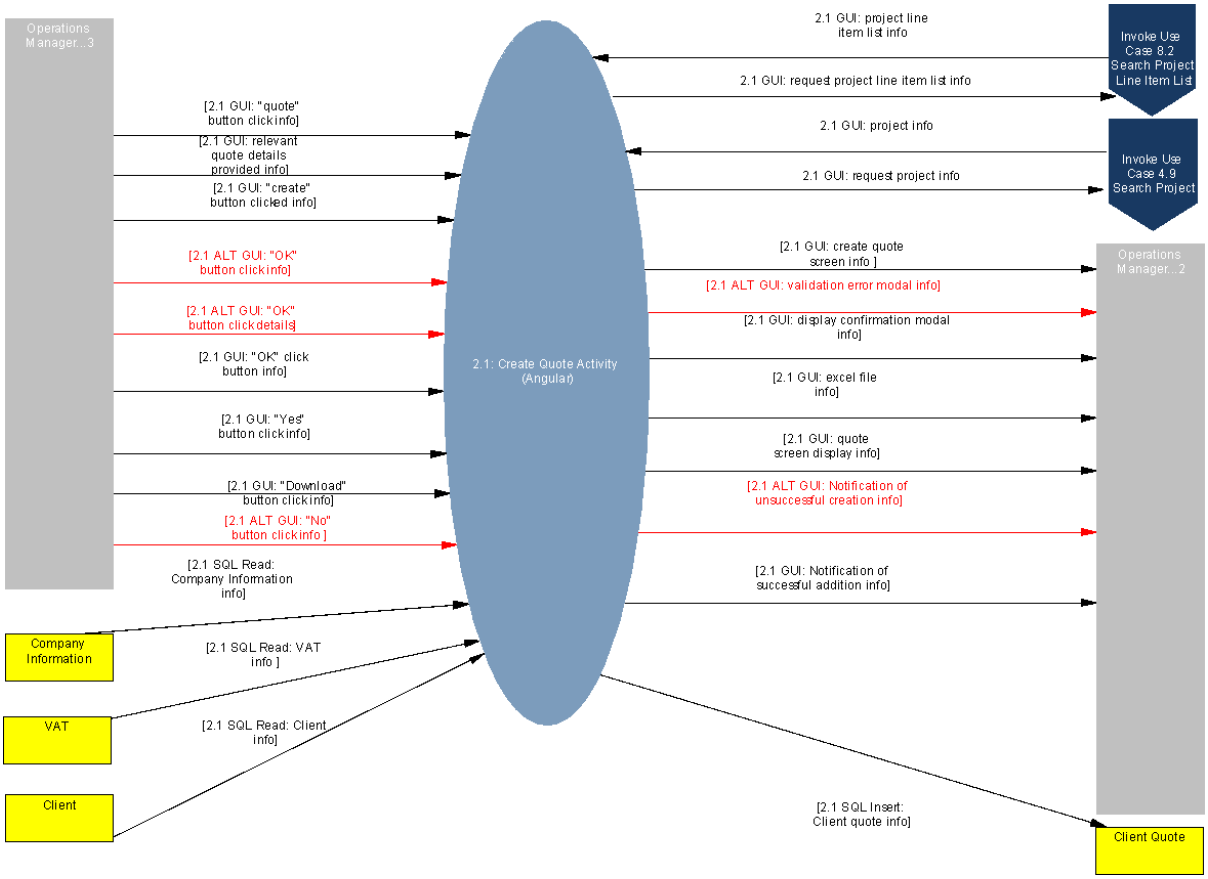


Figure 65: Middle level diagram 2.1 Create Quote

2.3.2.2.2 Search Quote

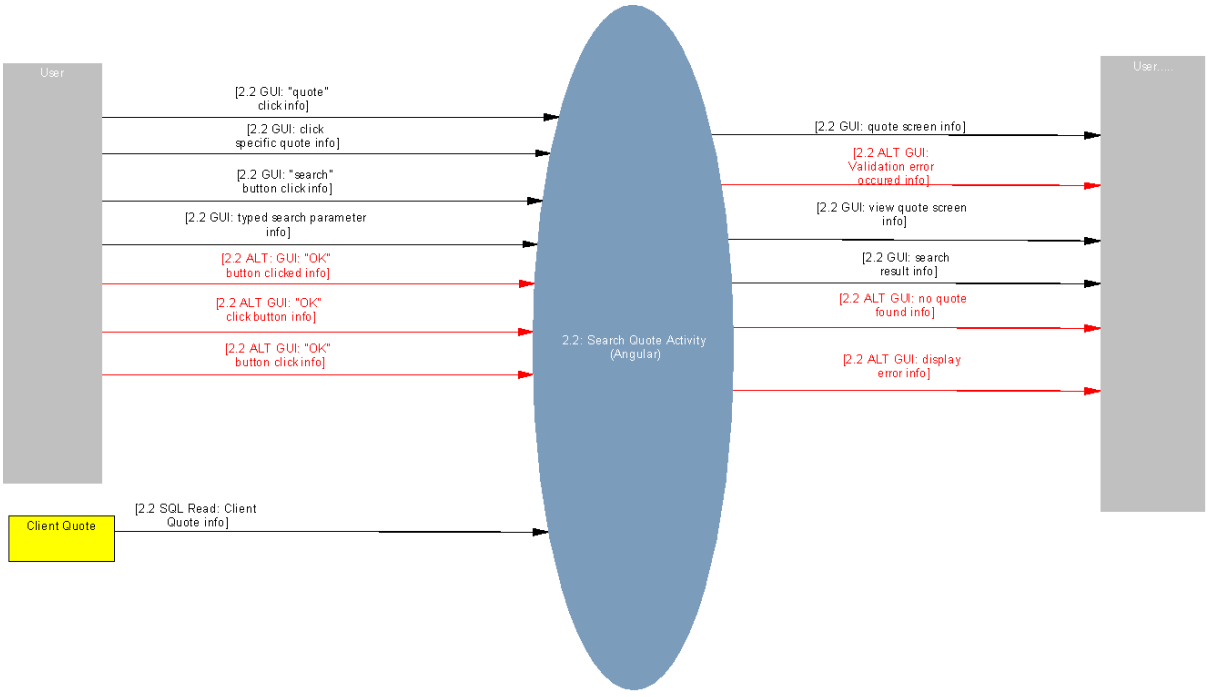


Figure 66: Middle level diagram 2.2 Search Quote

2.3.2.2.3 Capture Purchase Order

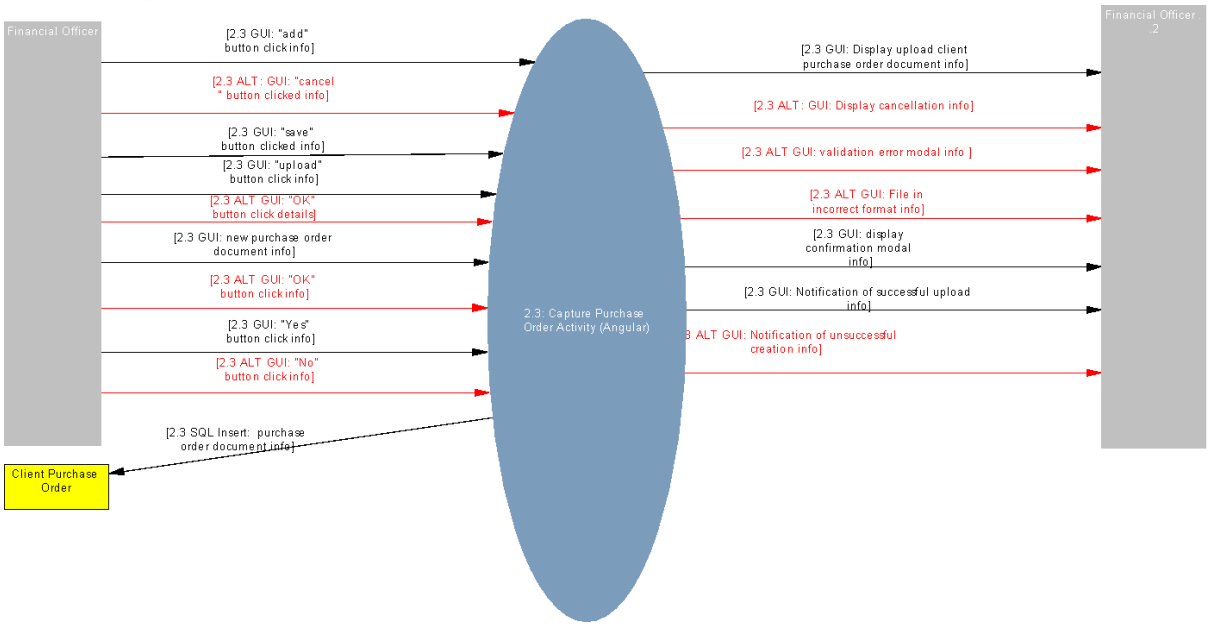


Figure 67: Middle level diagram 2.3 Capture Purchase order

2.3.2.3. Administration Subsystem

2.3.2.3.1 Create Health and Safety file

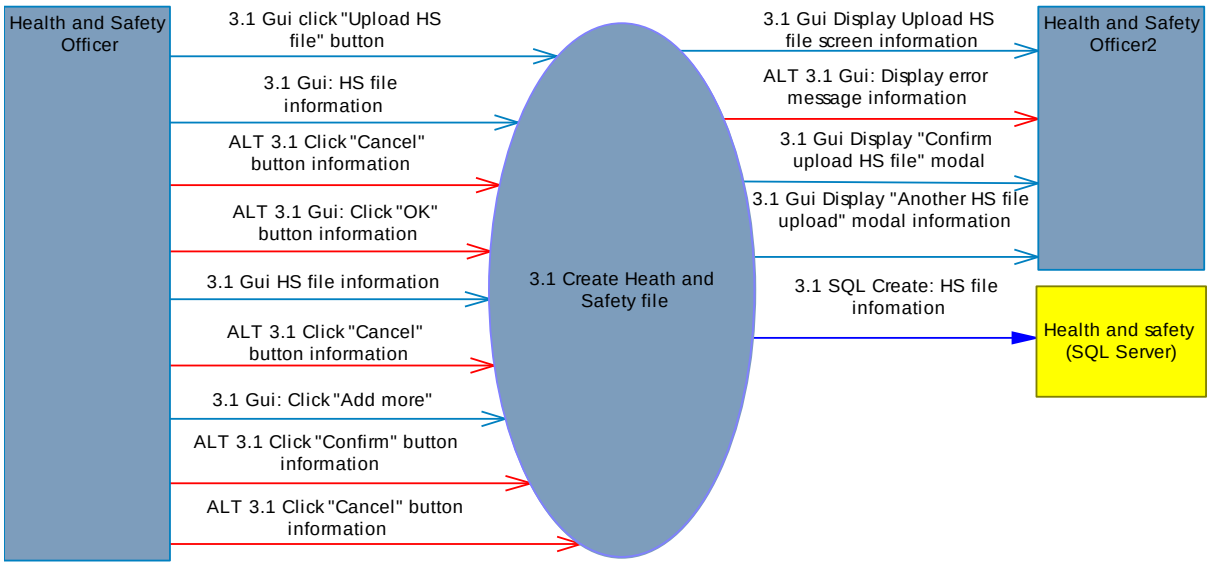


Figure 68: Middle level diagram 3.1 Create HS file

2.3.2.3.2 Search Health and Safety file

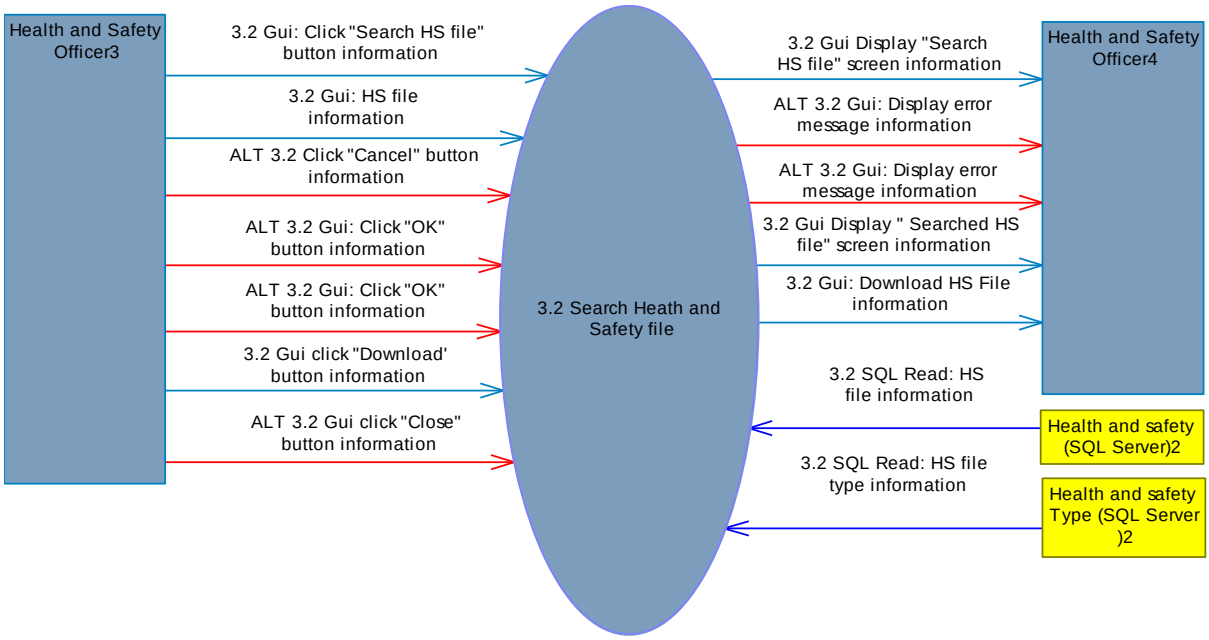


Figure 69: Middle level diagram 3.2 Search HS file

2.3.2.3.3 Create Health and Safety Type

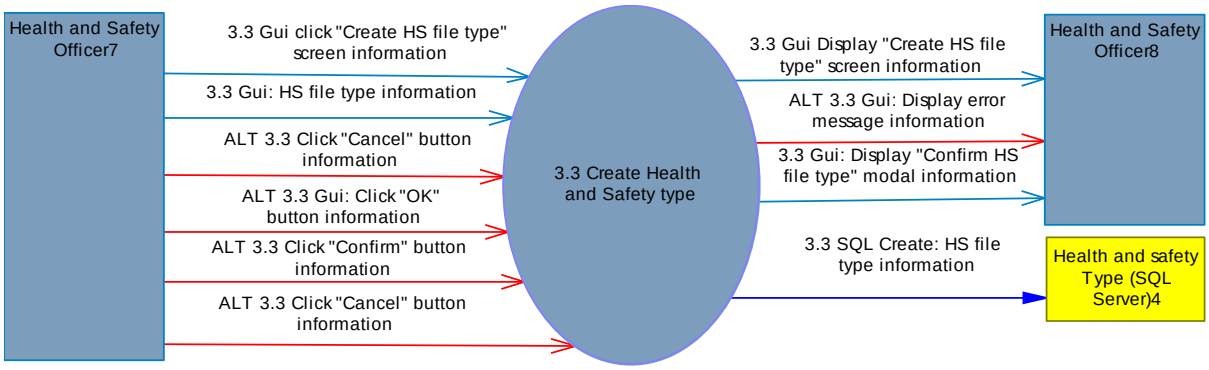


Figure 70: Middle level diagram 3.3 Create HS type

2.3.2.3.4 Search Health and Safety Type

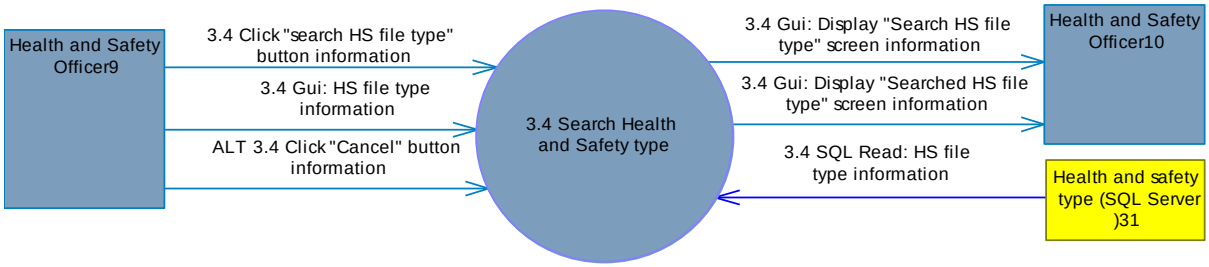


Figure 71: Middle level diagram 3.4 Search HS type

2.3.2.3.5 Create Drawing Type

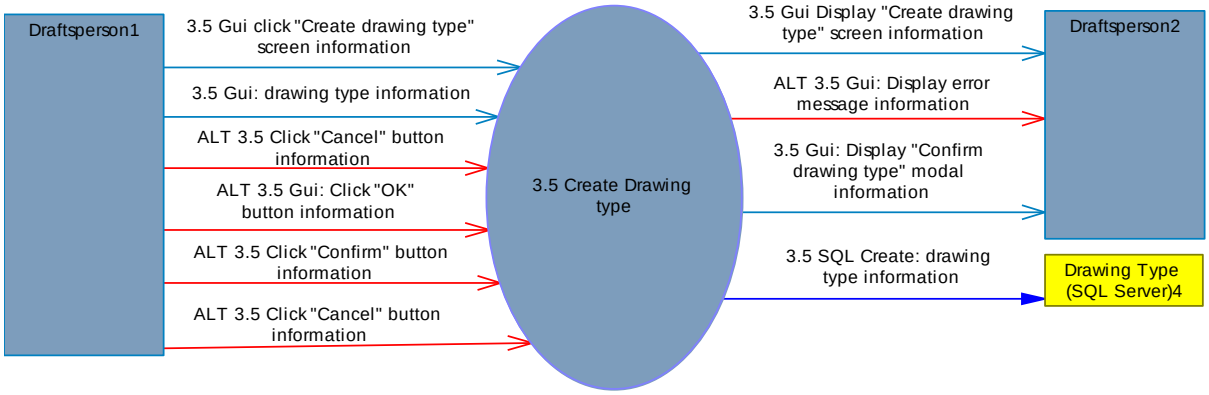


Figure 72: Middle level diagram 3.5 Create drawing type

2.3.2.3.6 Search Drawing Type

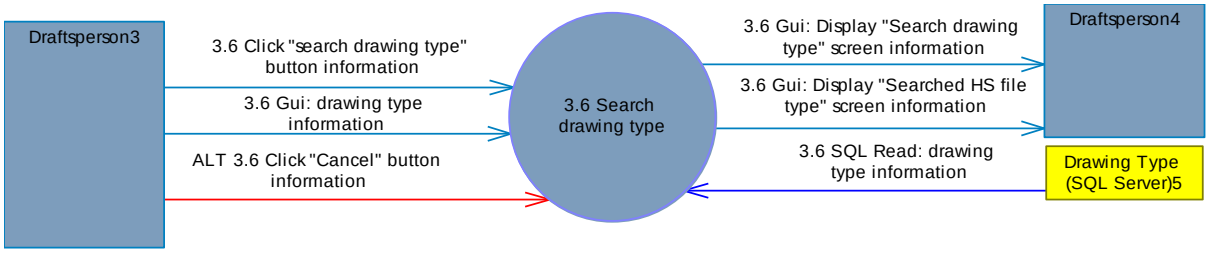


Figure 73: Middle level diagram 3.6 Search Drawing type

2.3.2.3.7 Upload Formal Drawing

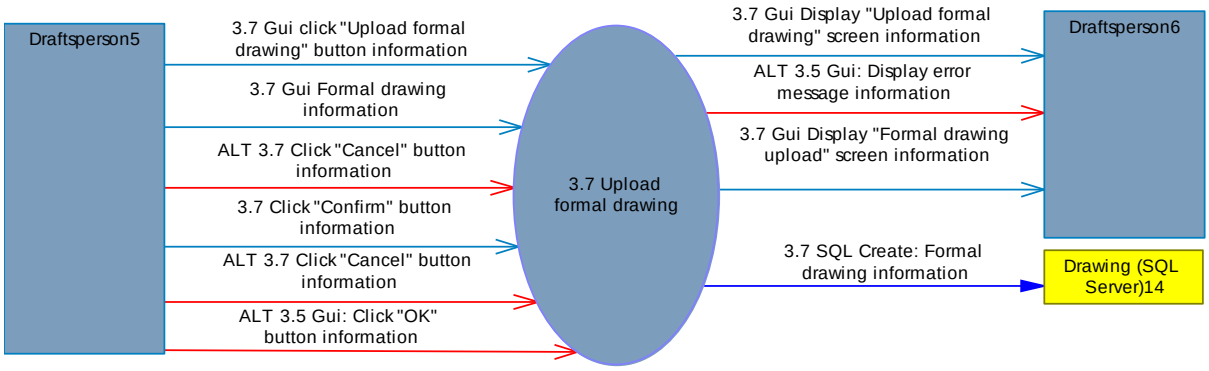


Figure 74: Middle level diagram 3.7 Upload formal drawing

2.3.2.3.8 Upload Second cut Drawing

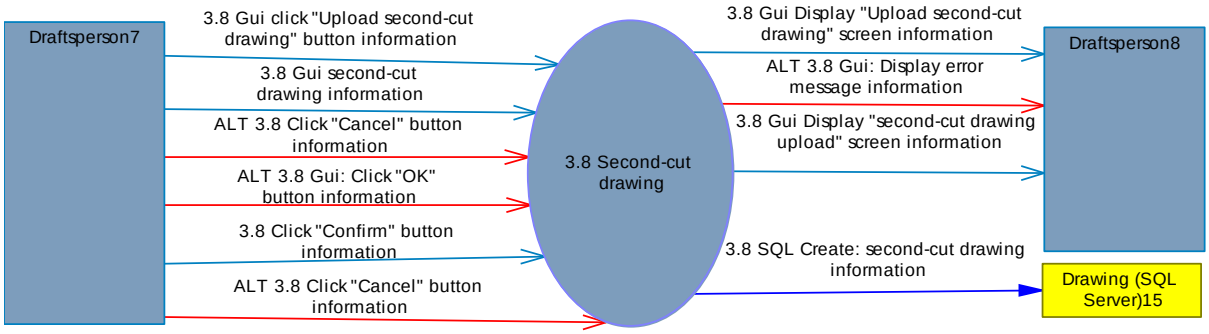


Figure 75: Middle level diagram 3.8 Upload second-cut drawing

2.3.2.3.9 Search As-built document

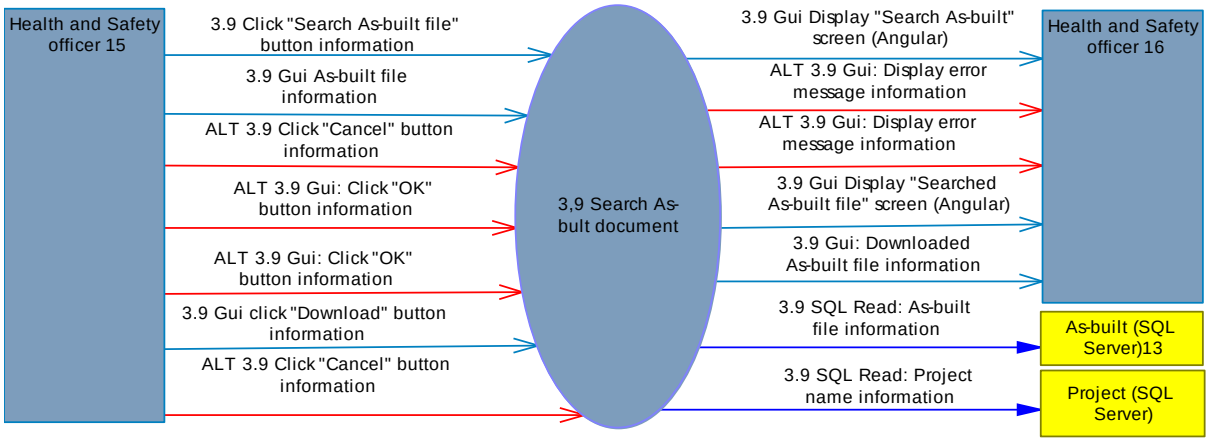


Figure 76: Middle level diagram 3.9 Search As-built document

2.3.2.3.10 Generate As-Built Document

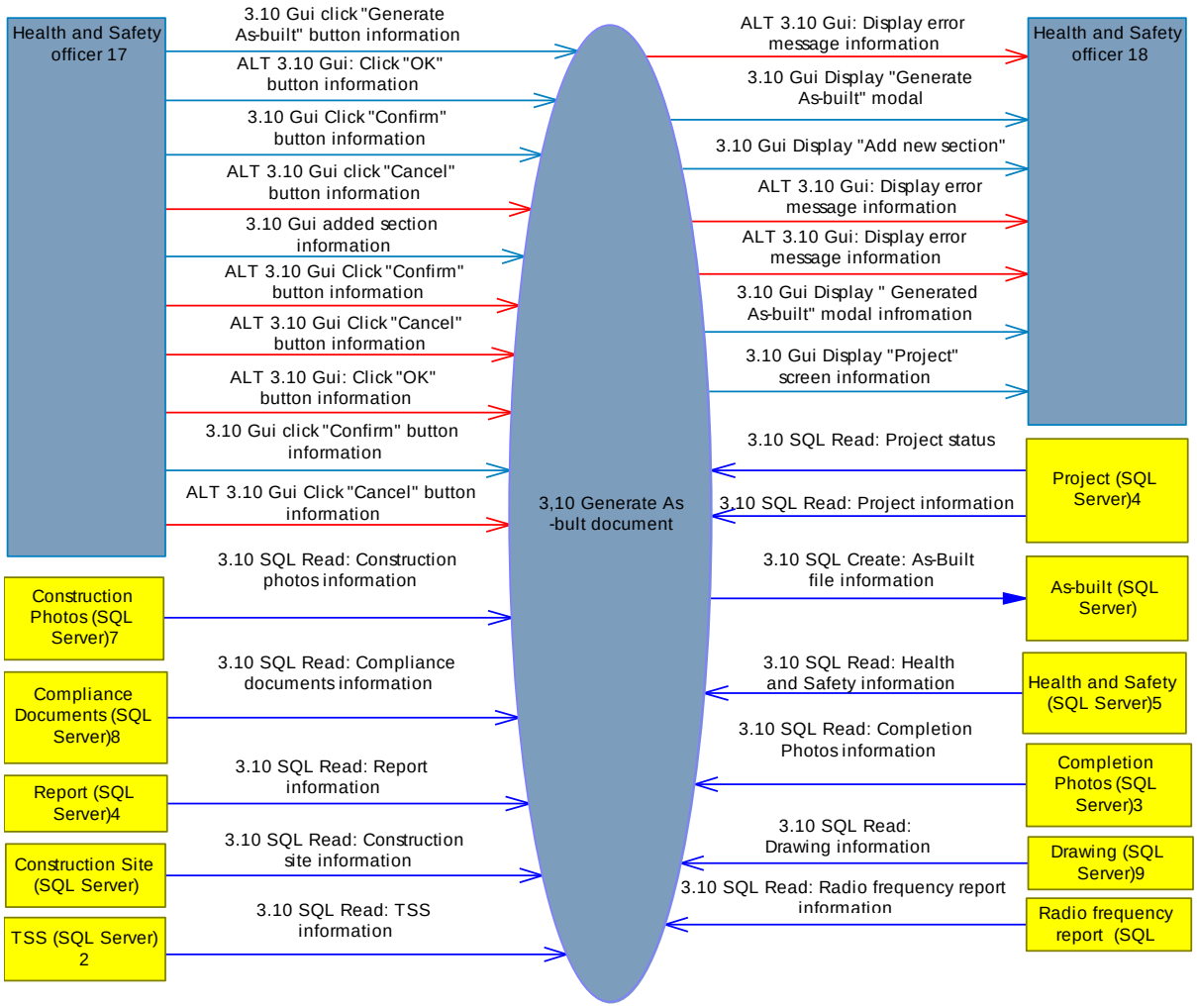


Figure 77: Middle level diagram 3.10 Generate As-built document

2.3.2.4. Construction Subsystem

2.3.2.4.1 Capture Site-Go-Ahead

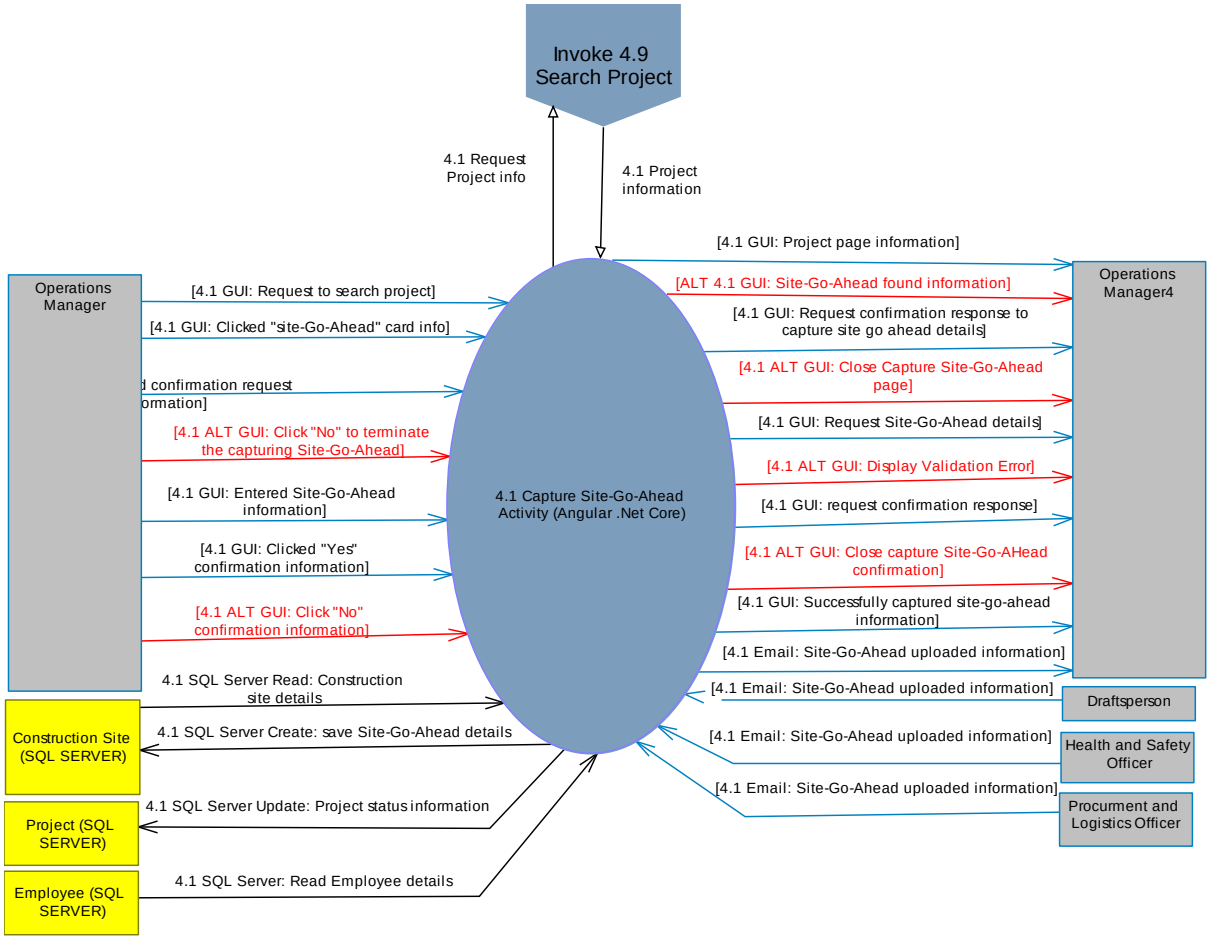


Figure 78: Middle level diagram 4.1 Capture Site-Go-Ahead

2.3.2.4.2 Search construction

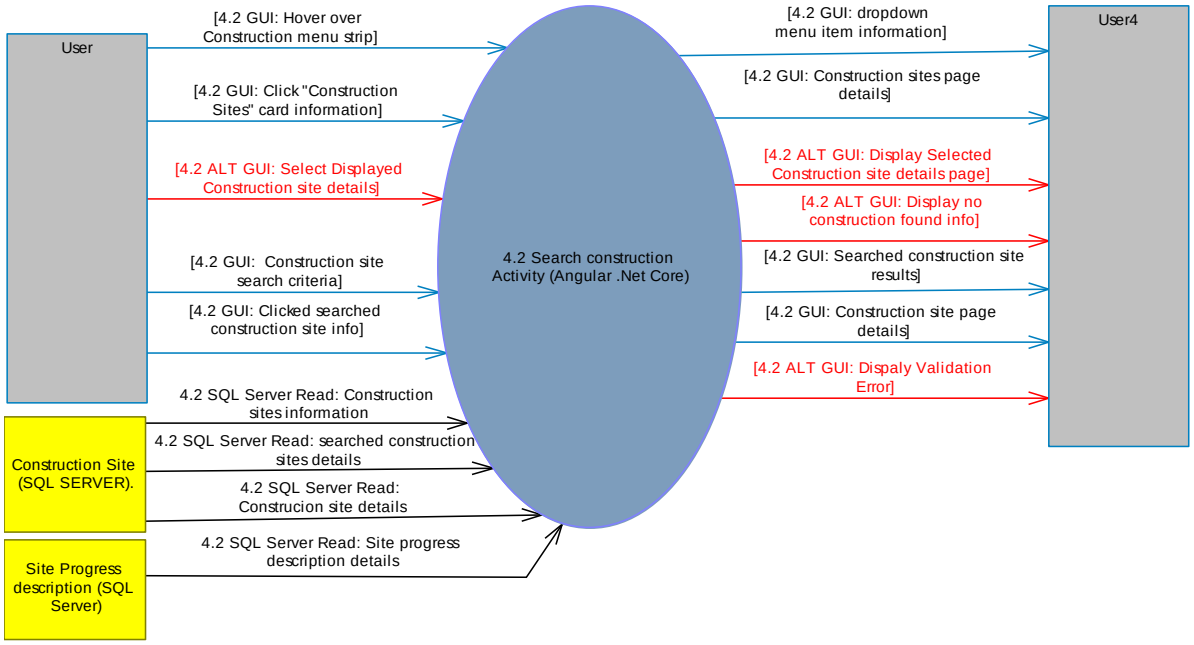


Figure 79: Middle level diagram 4.2 Search construction

2.3.2.4.3 Edit construction

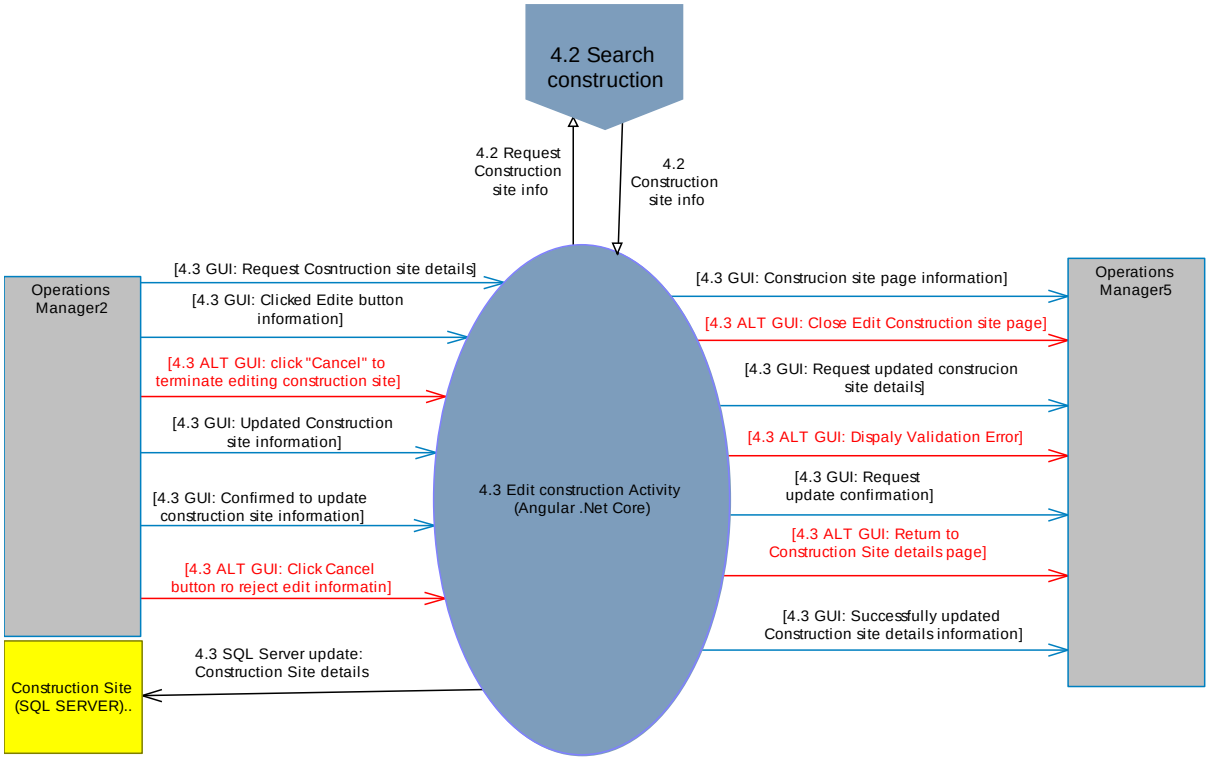


Figure 80: Middle level diagram 4.3 Edit construction

2.3.2.4.4 Allocate Team Site

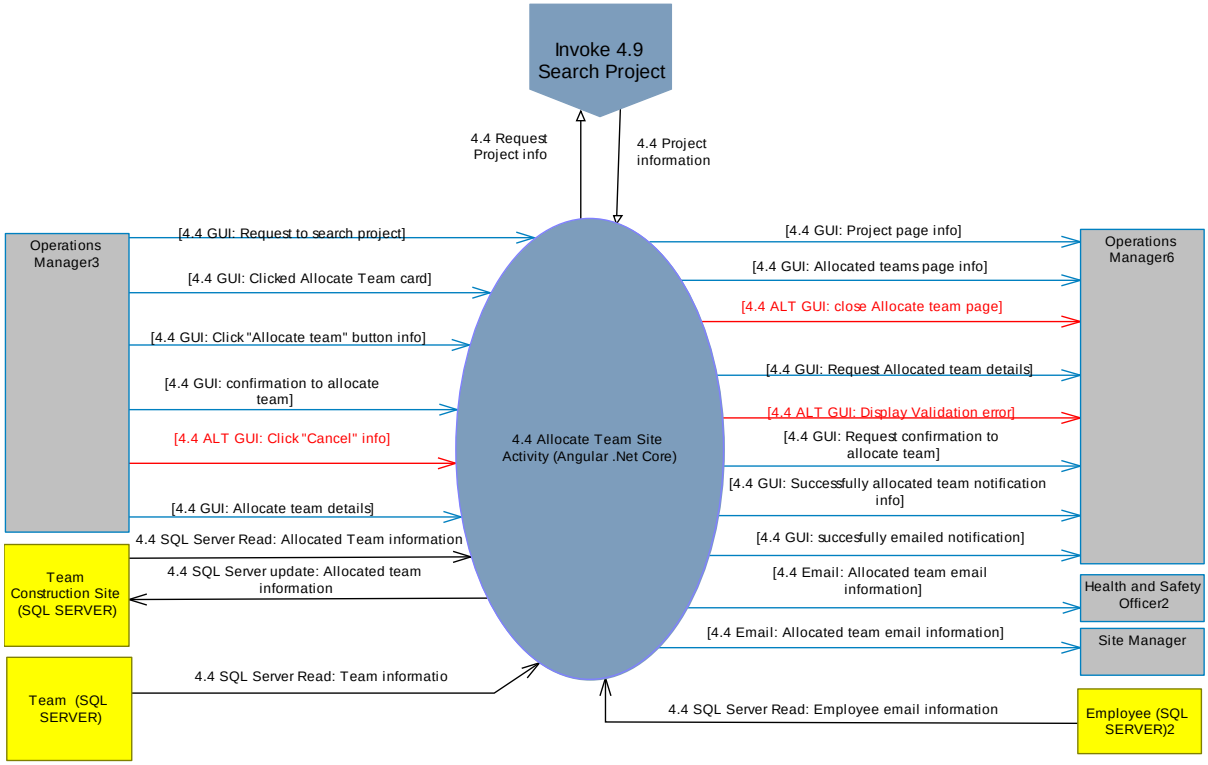


Figure 81: Middle level diagram 4.4 Allocate Team Site

2.3.2.4.5 Search Allocated Team

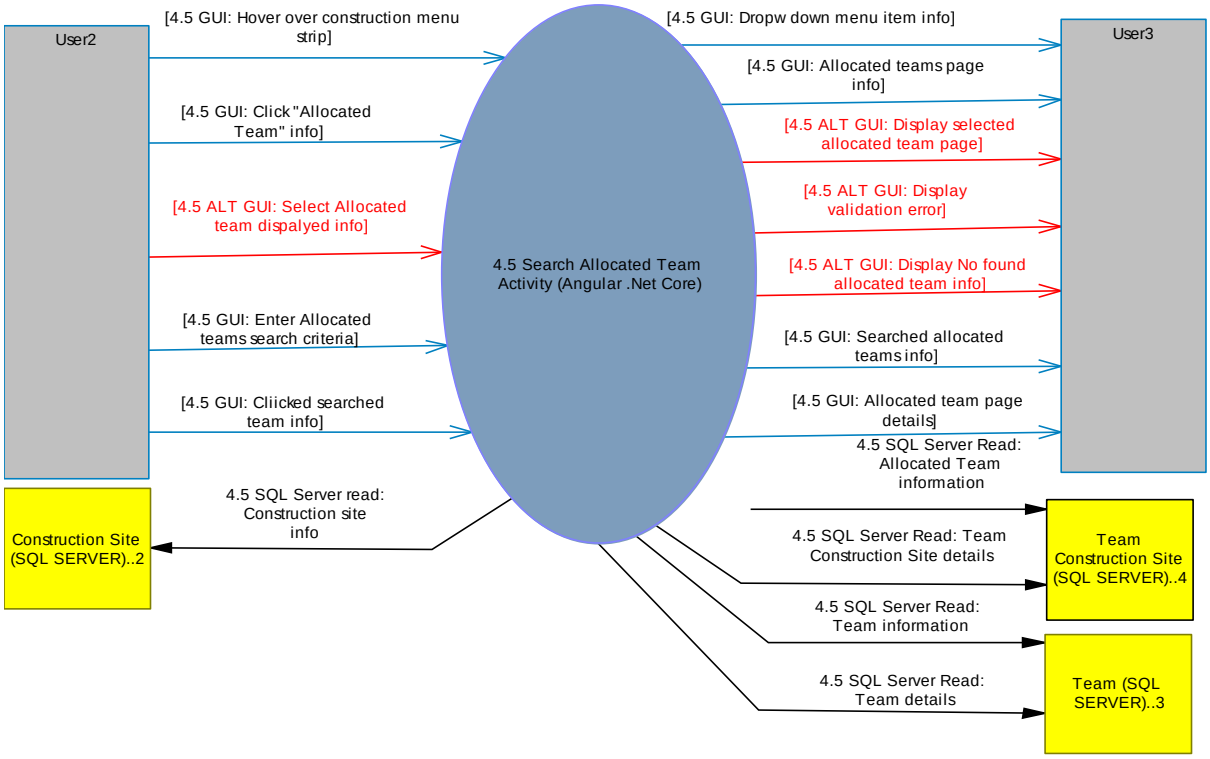


Figure 82: Middle level diagram 4.5 Search Allocated Team

2.3.2.4.6 Update Allocated Team

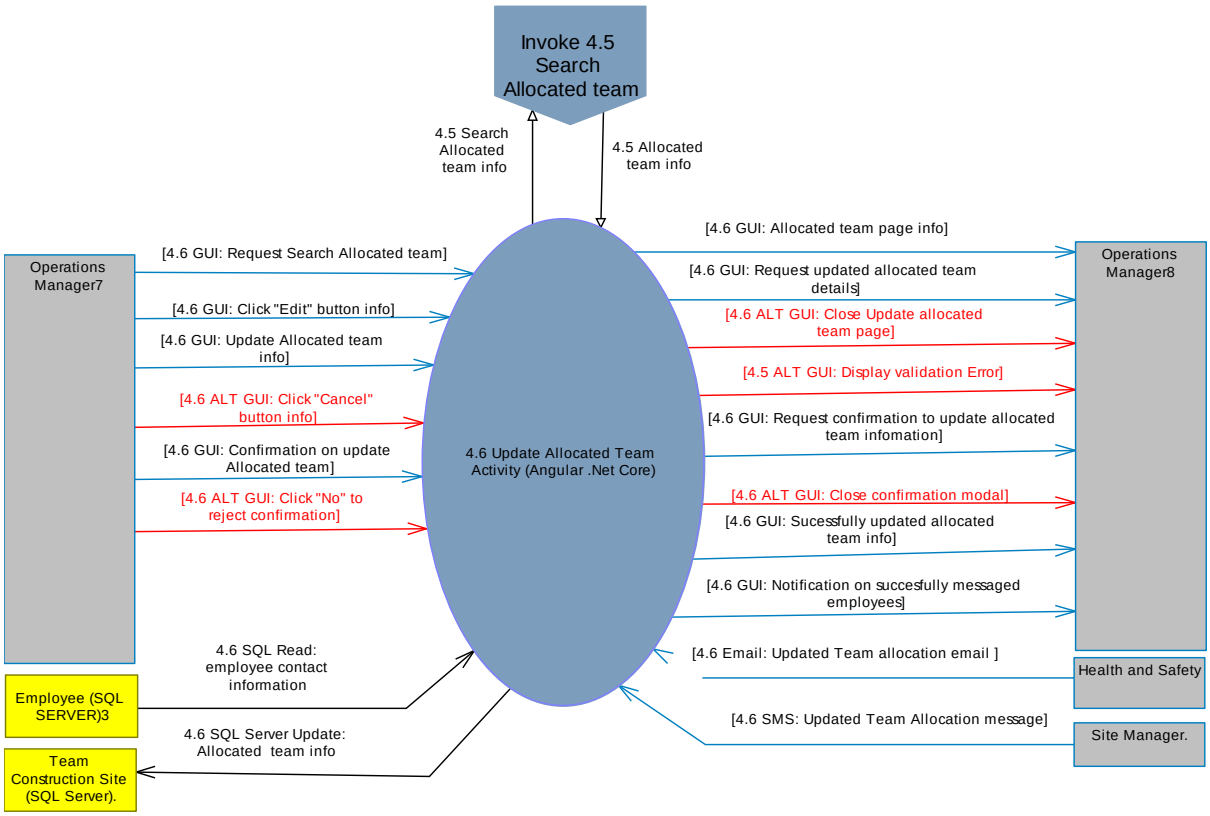


Figure 83: Middle level diagram 4.6 Update Allocated Team

2.3.2.4.7 Upload Compliance Document

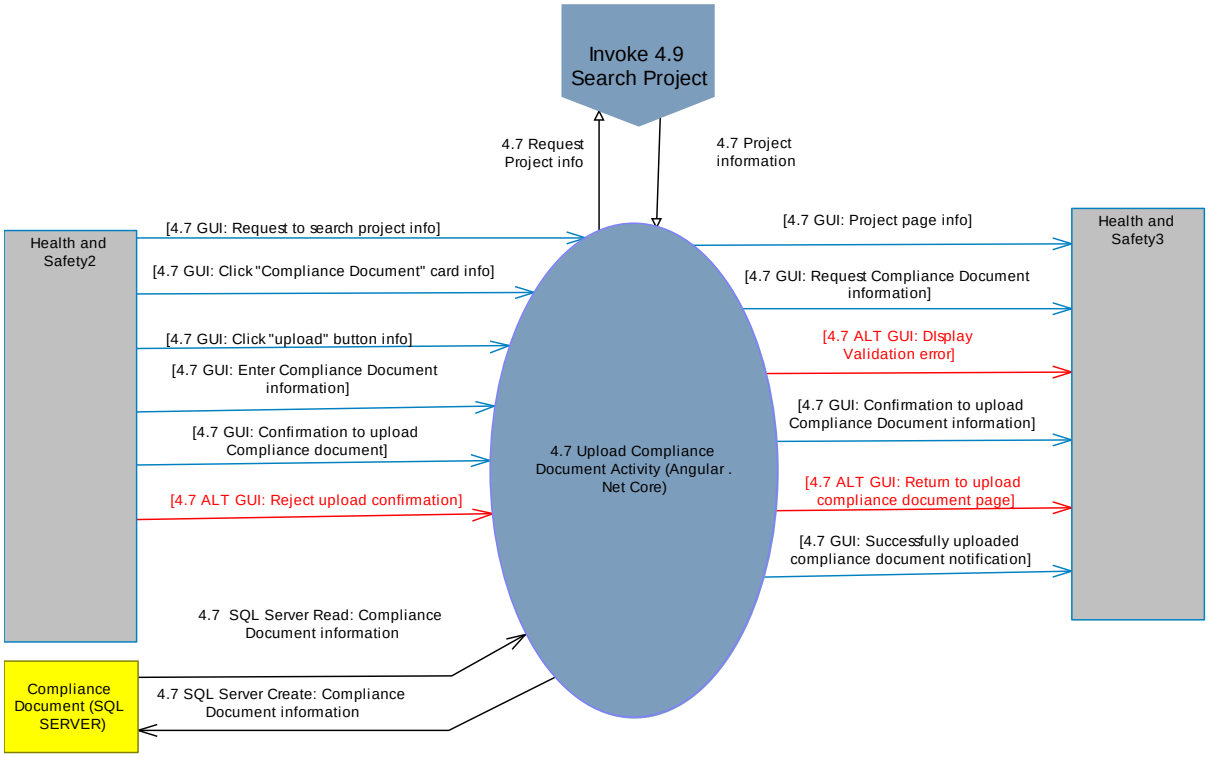


Figure 84: Middle level diagram 4.7 Upload Compliance Document

2.3.2.4.8 Search Compliance Document

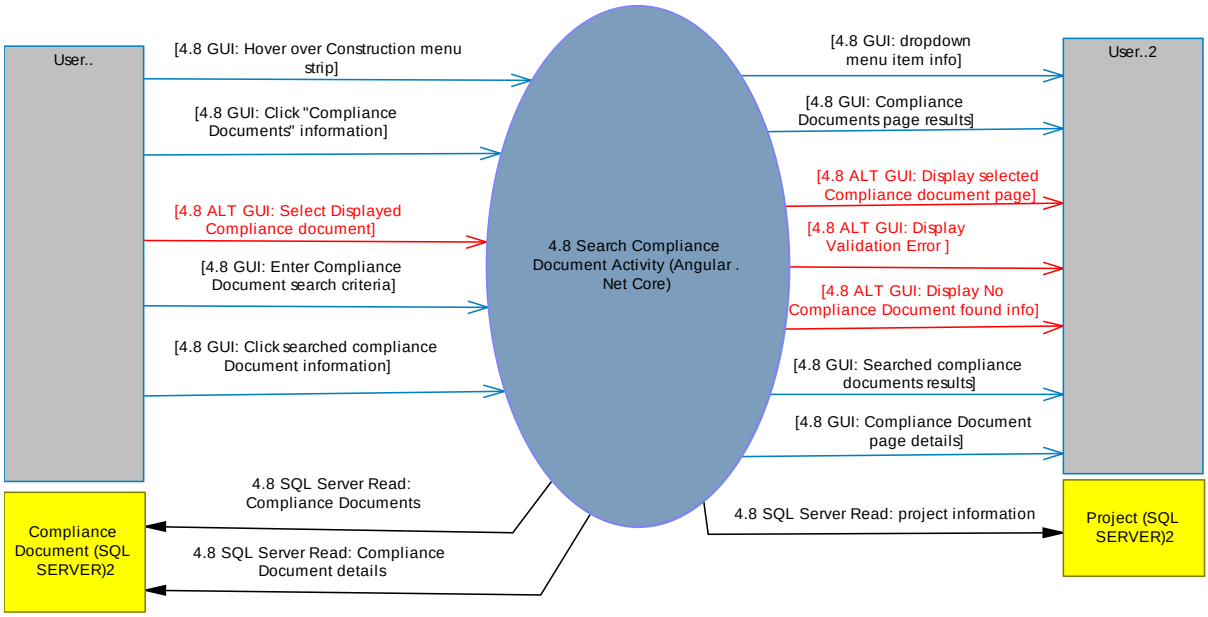


Figure 85: Middle level diagram 4.8 Search Compliance Document

2.3.2.4.9 Search Project

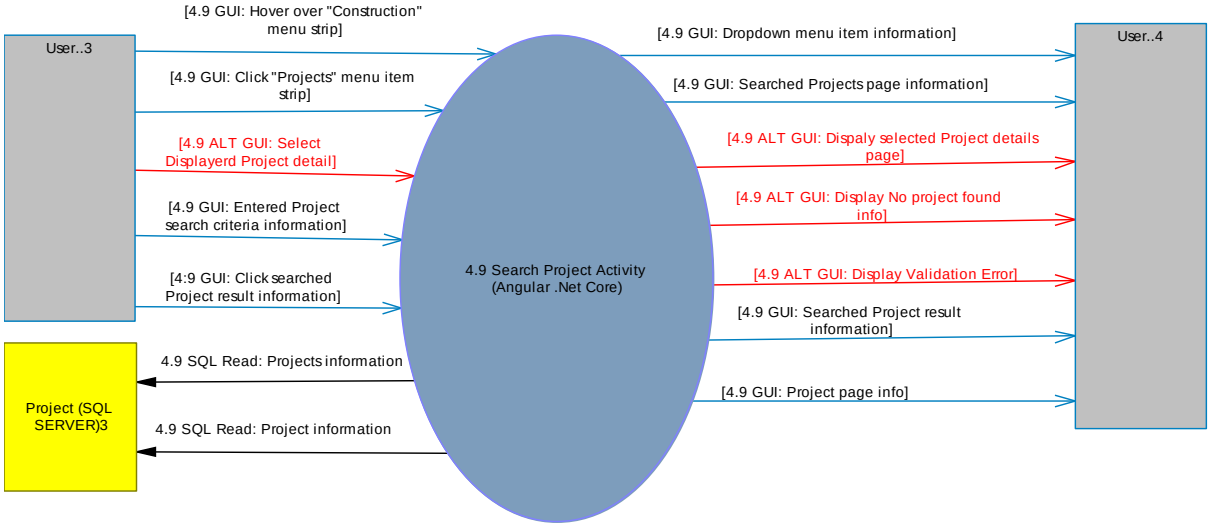


Figure 86: Middle level diagram 4.9 Search Project

2.3.2.4.10 Create Site Progress description

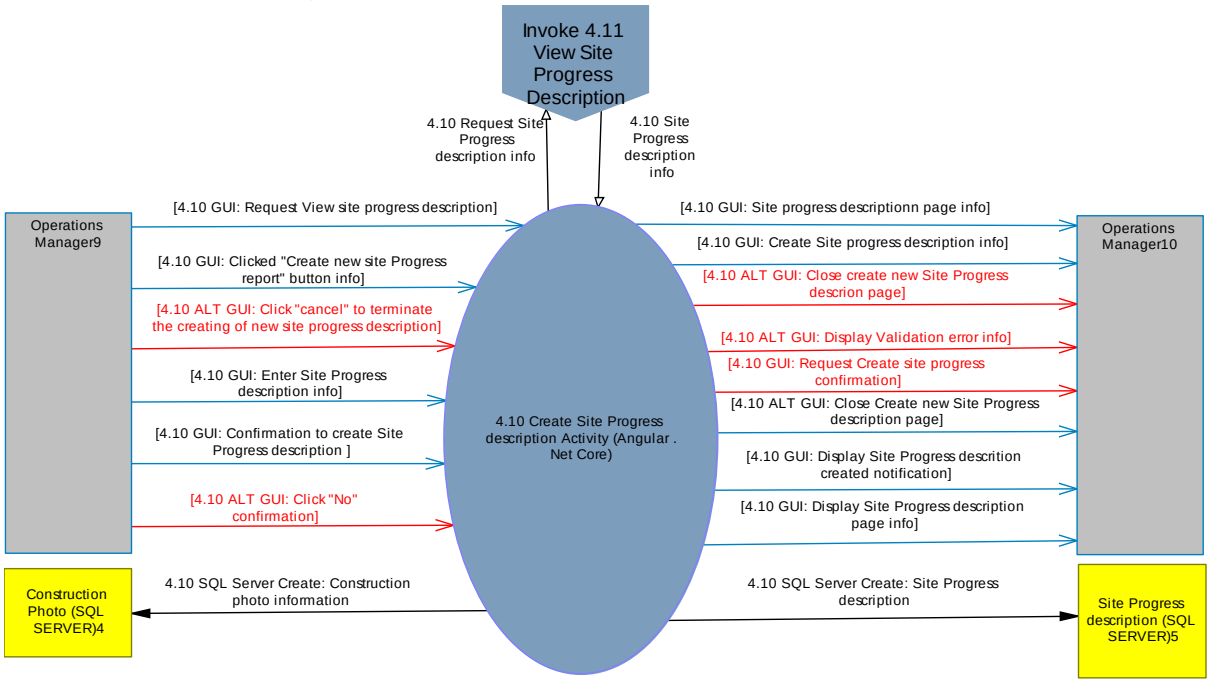


Figure 87: Middle level diagram 4.10 Create Site Progress description

2.3.2.4.11 View Site Progress description.

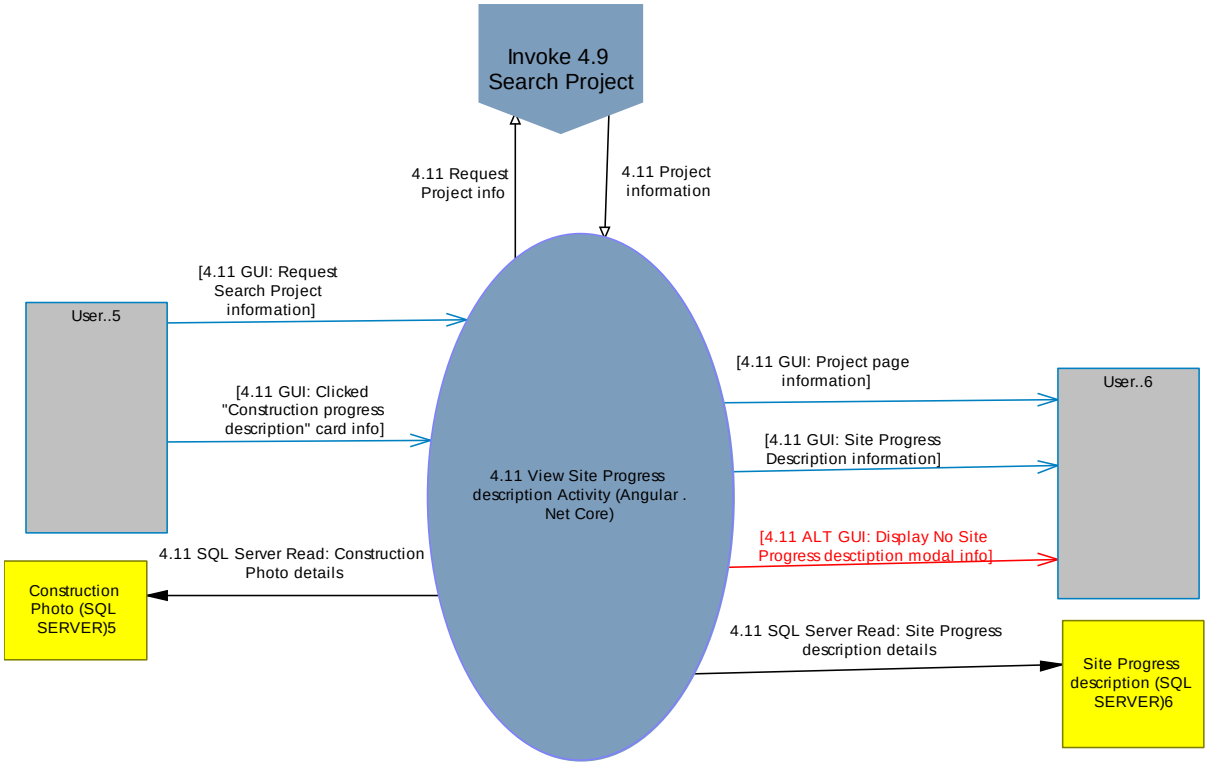


Figure 88: Middle level diagram 11 View Site Progress description

2.3.2.4.12 Upload complete site photos

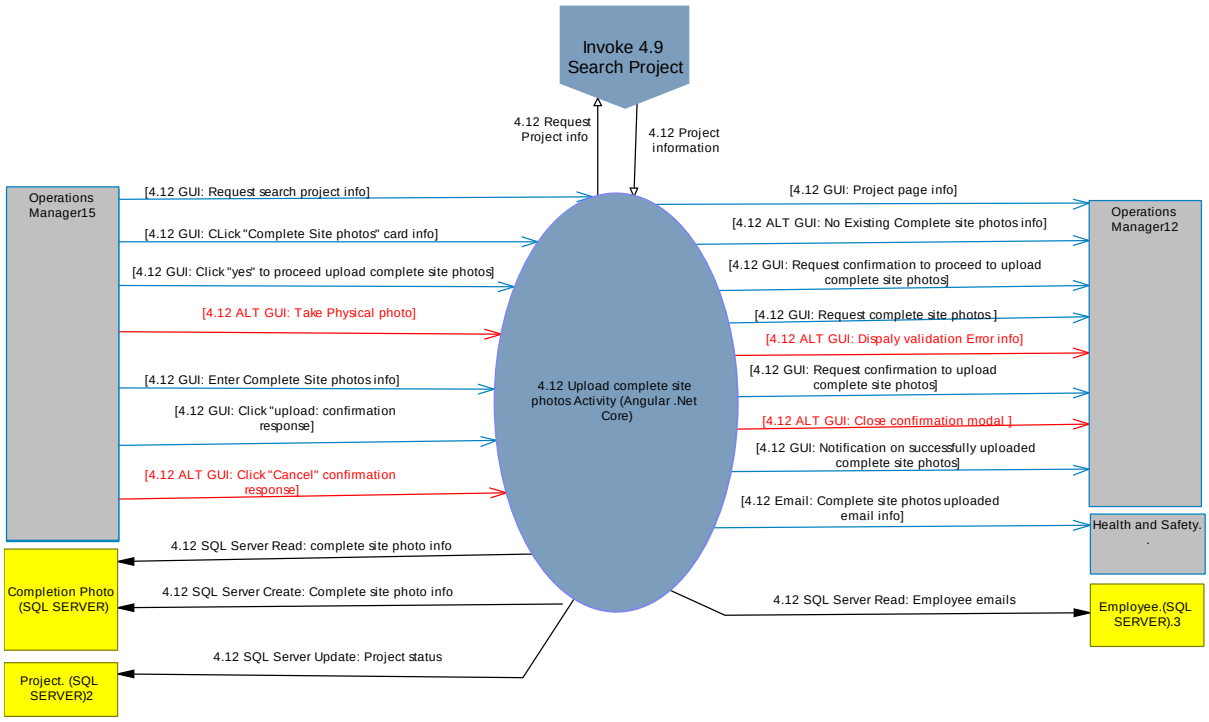


Figure 89: Middle level diagram 4.12 Upload complete site photos

2.3.2.4.13 Search complete site photos

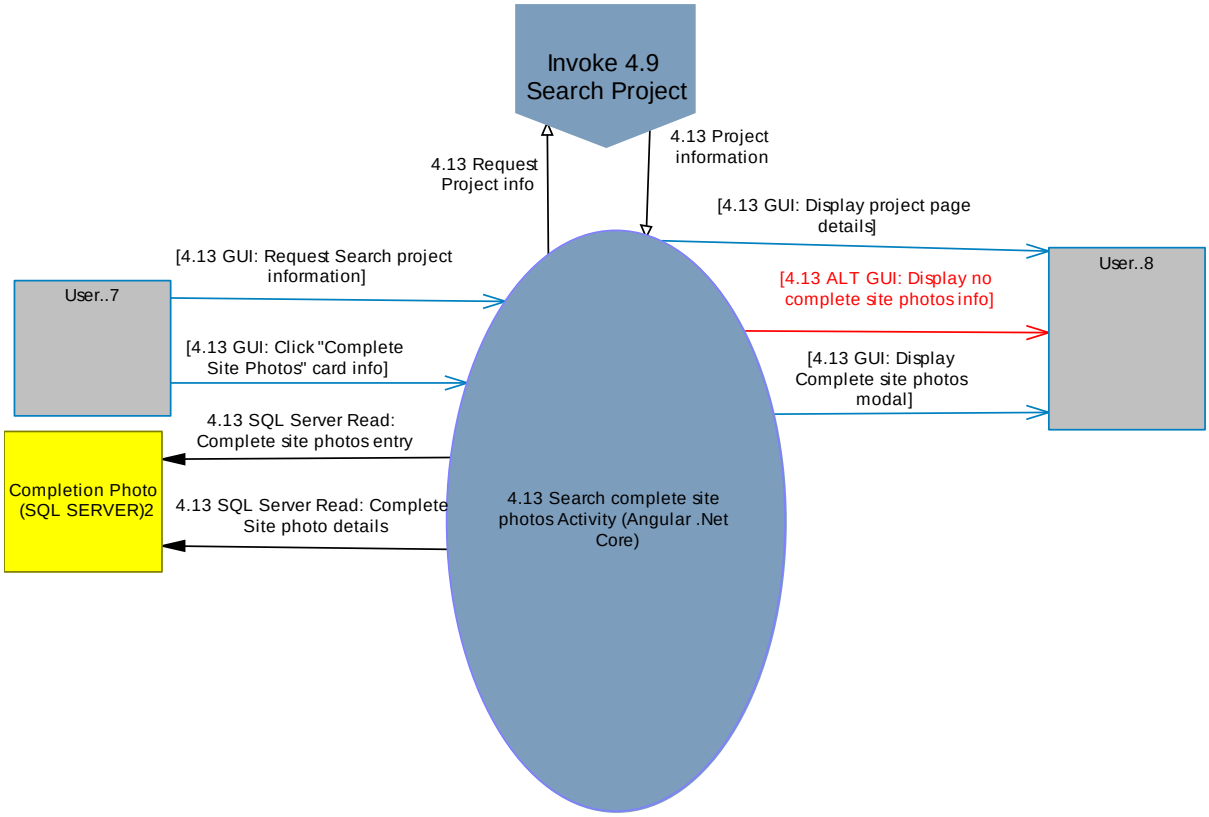


Figure 90: Middle level diagram 4.13 Search complete site photos

2.3.2.4.14 Upload Radio Frequency report

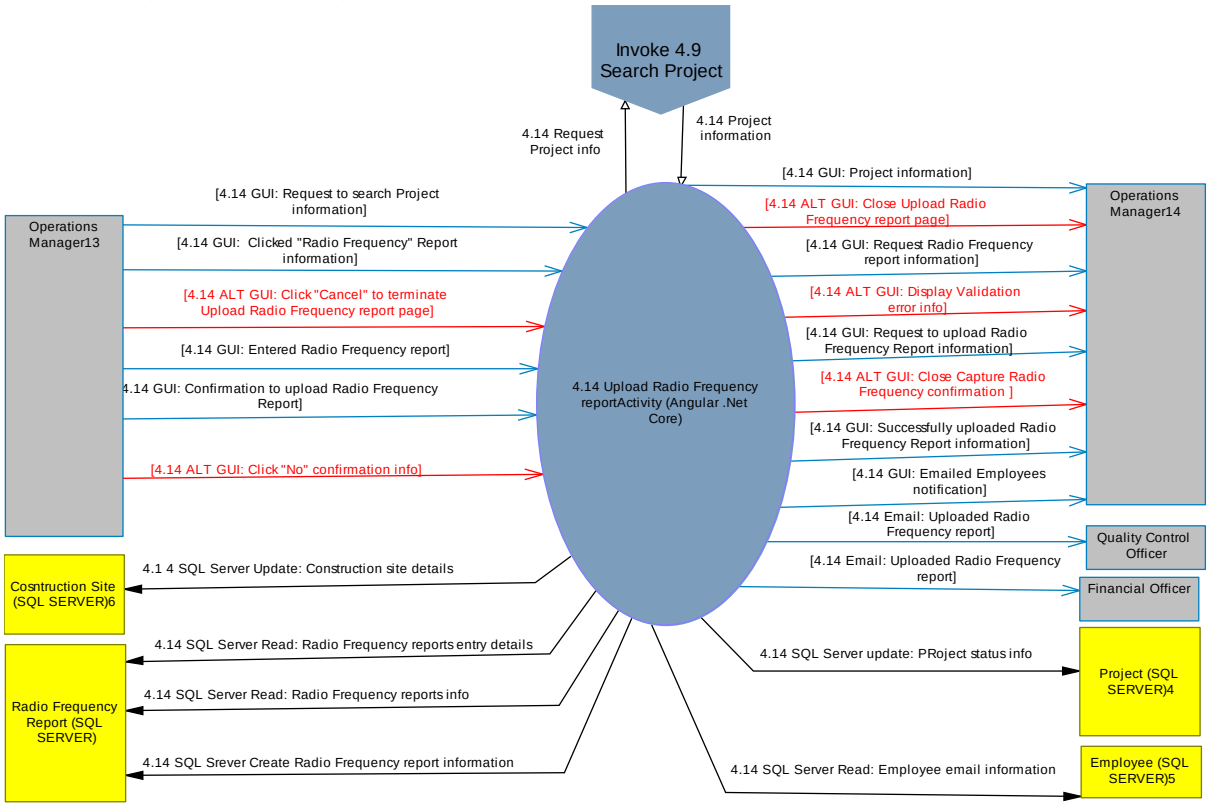


Figure 91: Middle level diagram 4.14 Upload Radio Frequency report

2.3.2.4.15 Search Radio Frequency report

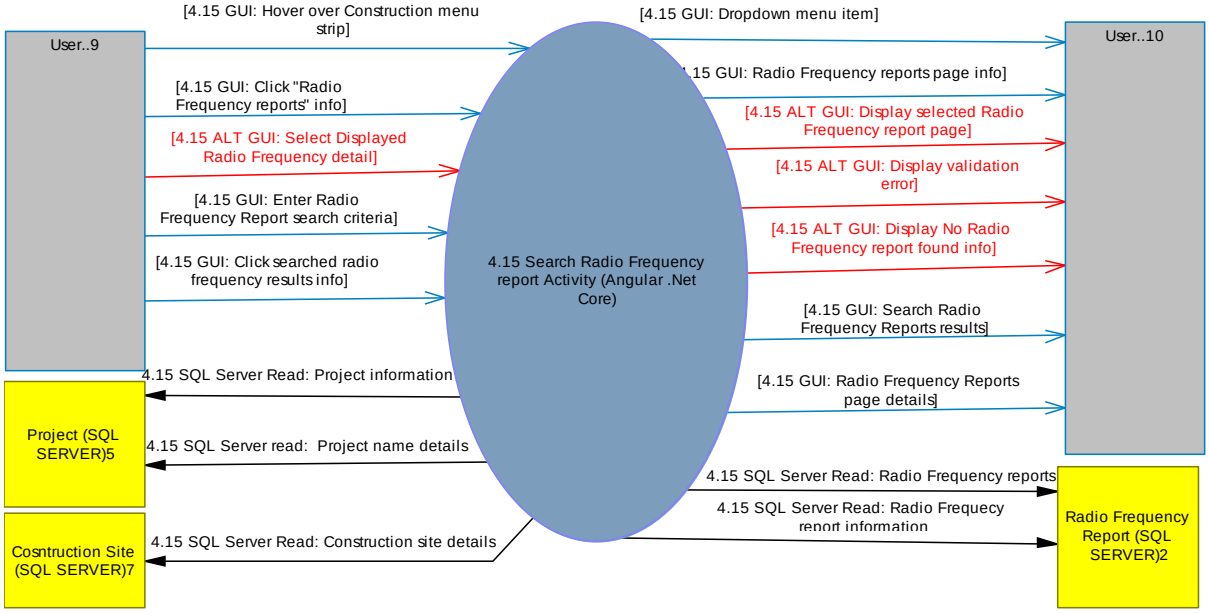


Figure 92: Middle level diagram 4.15 Search Radio Frequency report

2.3.2.4.16 Request Site Visit

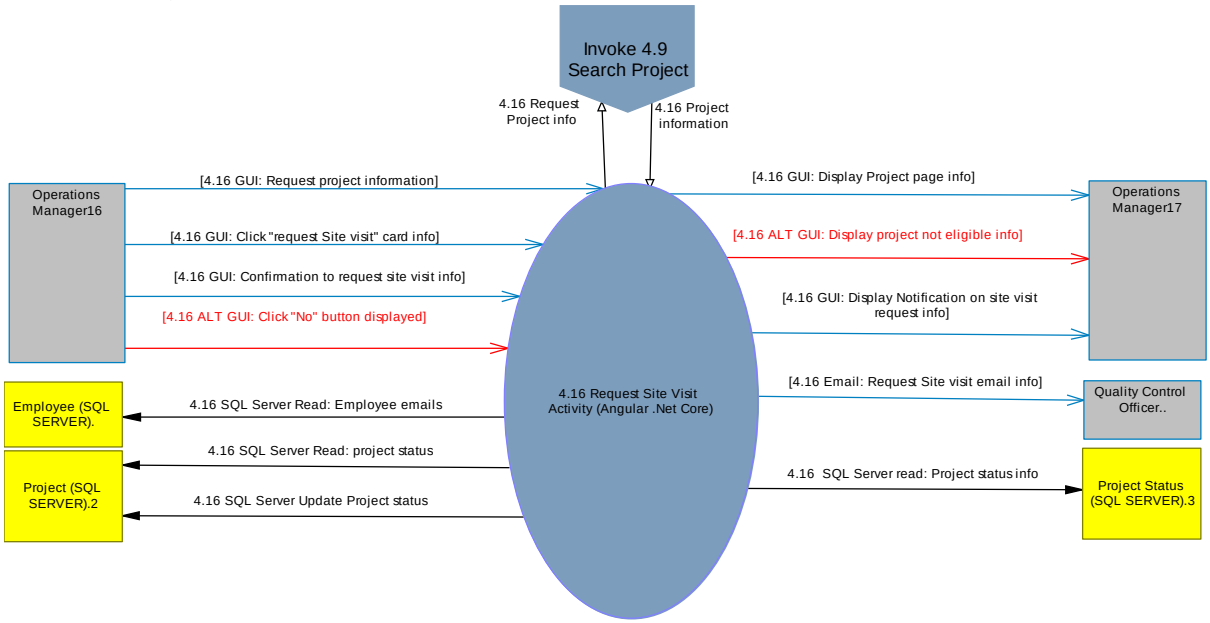


Figure 93: Middle level diagram 4.16 Request Site Visit

2.3.2.4.17 Create Snag

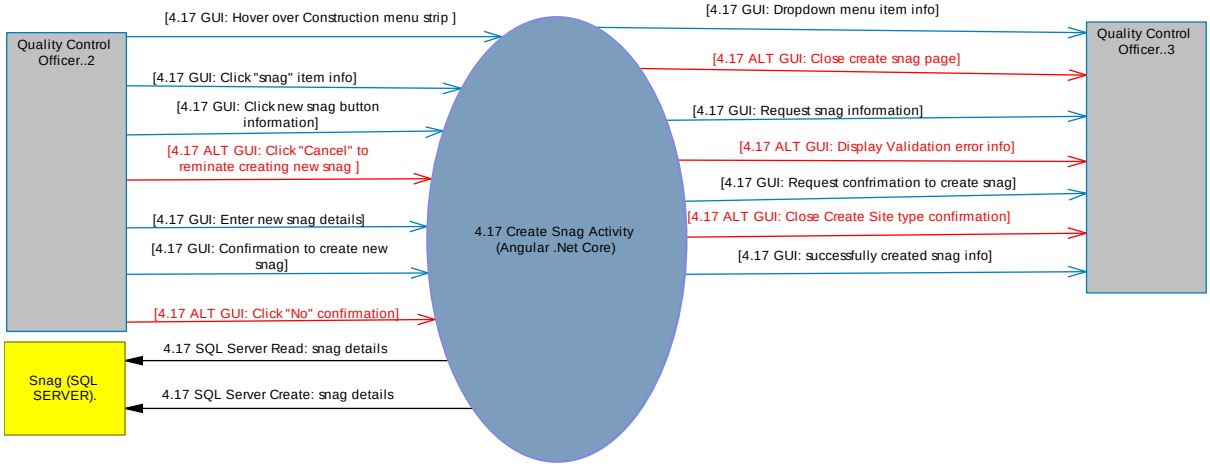


Figure 94: Middle level diagram 4.17 Create Snag

2.3.2.4.18 Search Snag

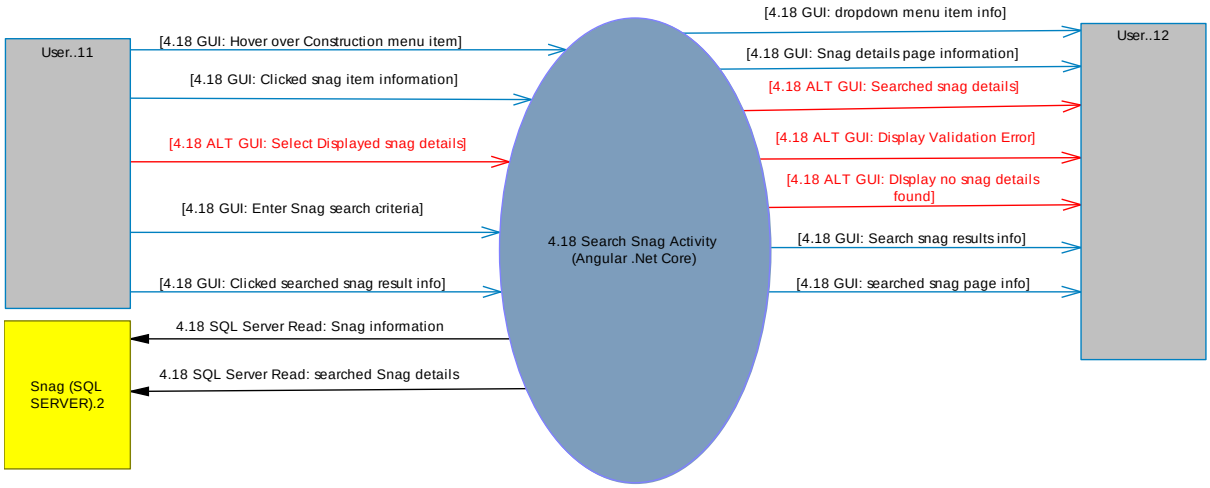


Figure 95: Middle level diagram 4.18 Search Snag

2.3.2.4.19 Edit Snag

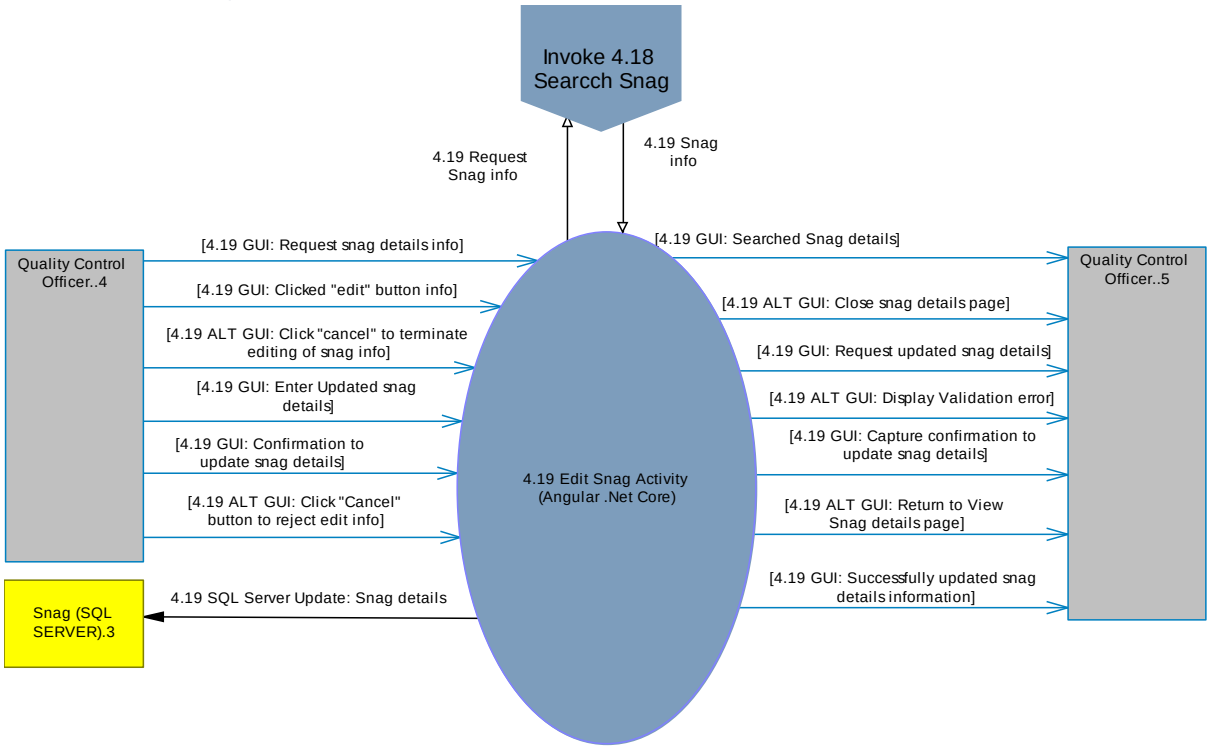


Figure 96: Middle level diagram 4.19 Edit Snag

2.3.2.4.20 Edit Acceptance Document

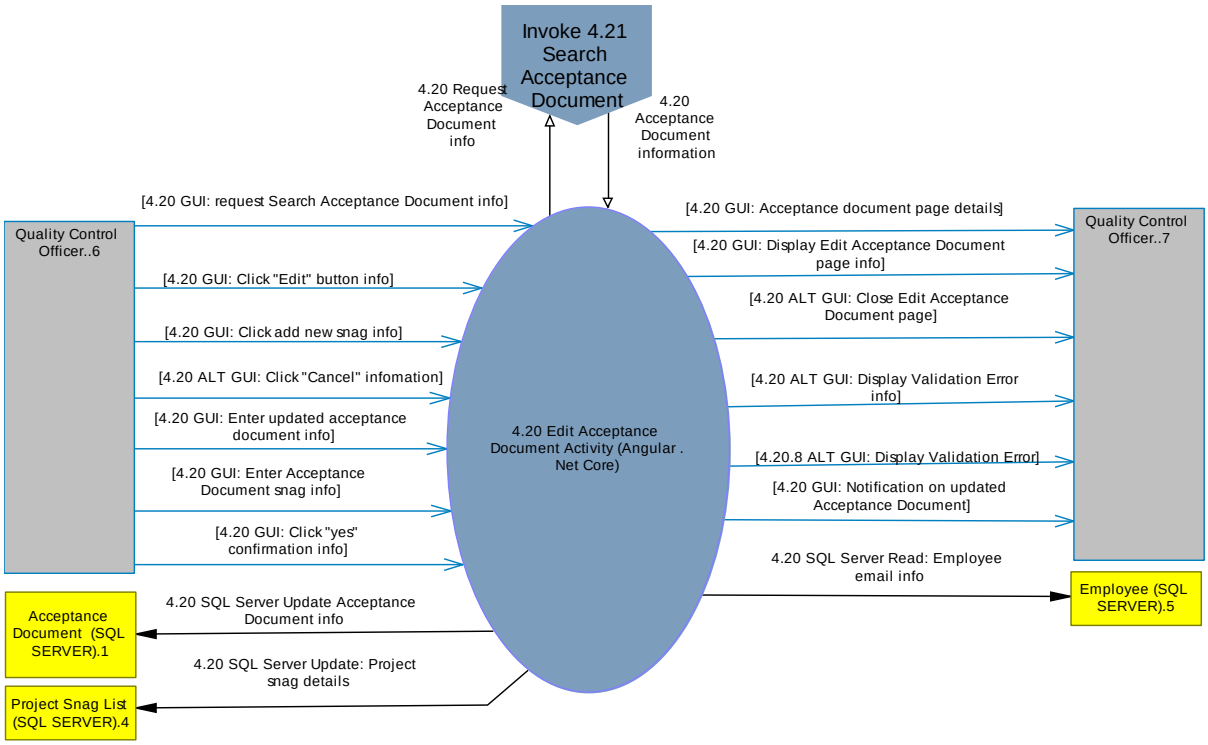


Figure 97: Middle level diagram 4.20 Edit Acceptance Document

2.3.2.4.21 Search Acceptance document

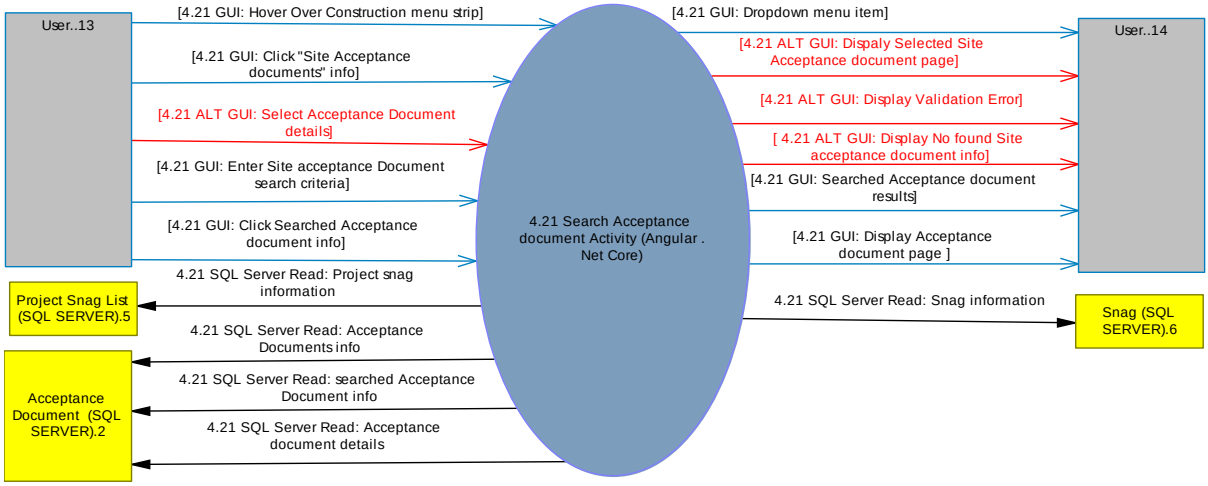


Figure 98: Middle level diagram 4.21 Search Acceptance document

2.3.2.4.22 Confirm acceptance document

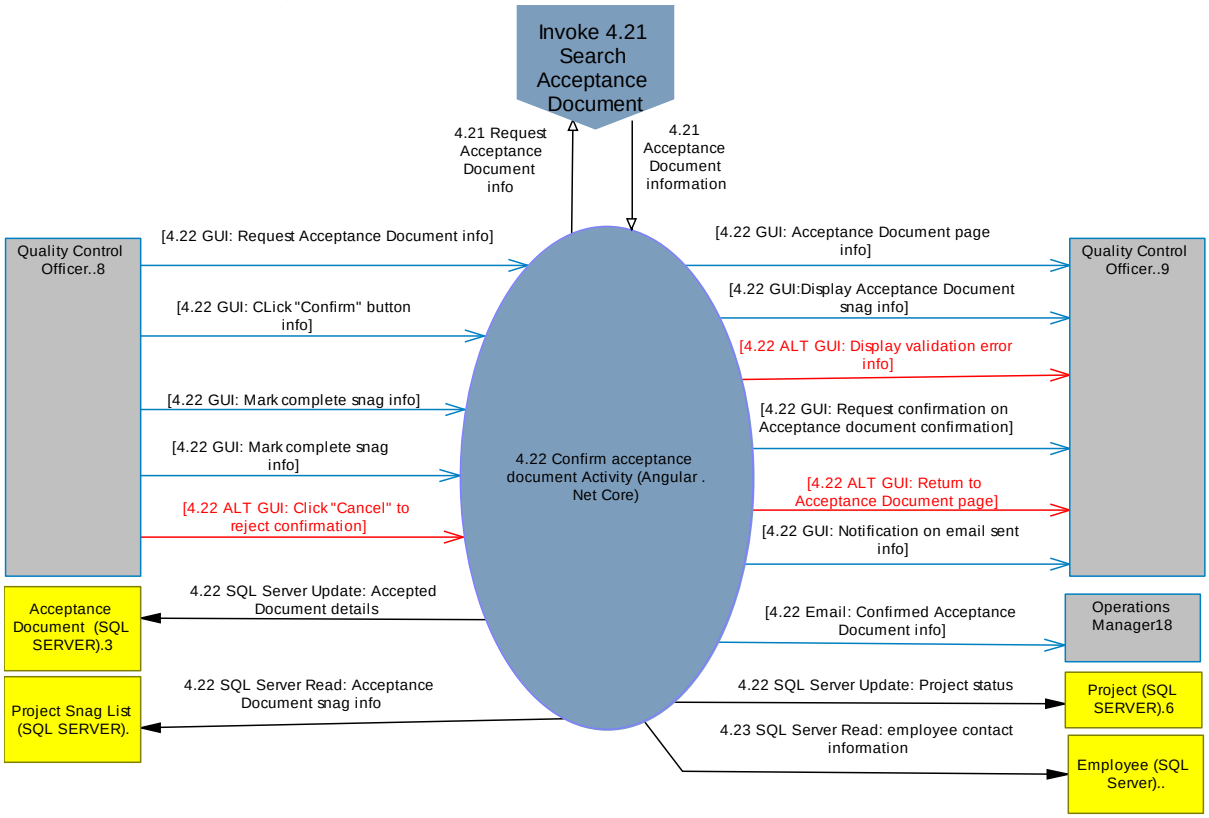


Figure 99: Middle level diagram 4.22 Confirm acceptance document

2.3.2.4.23 Capture Site Sign-off

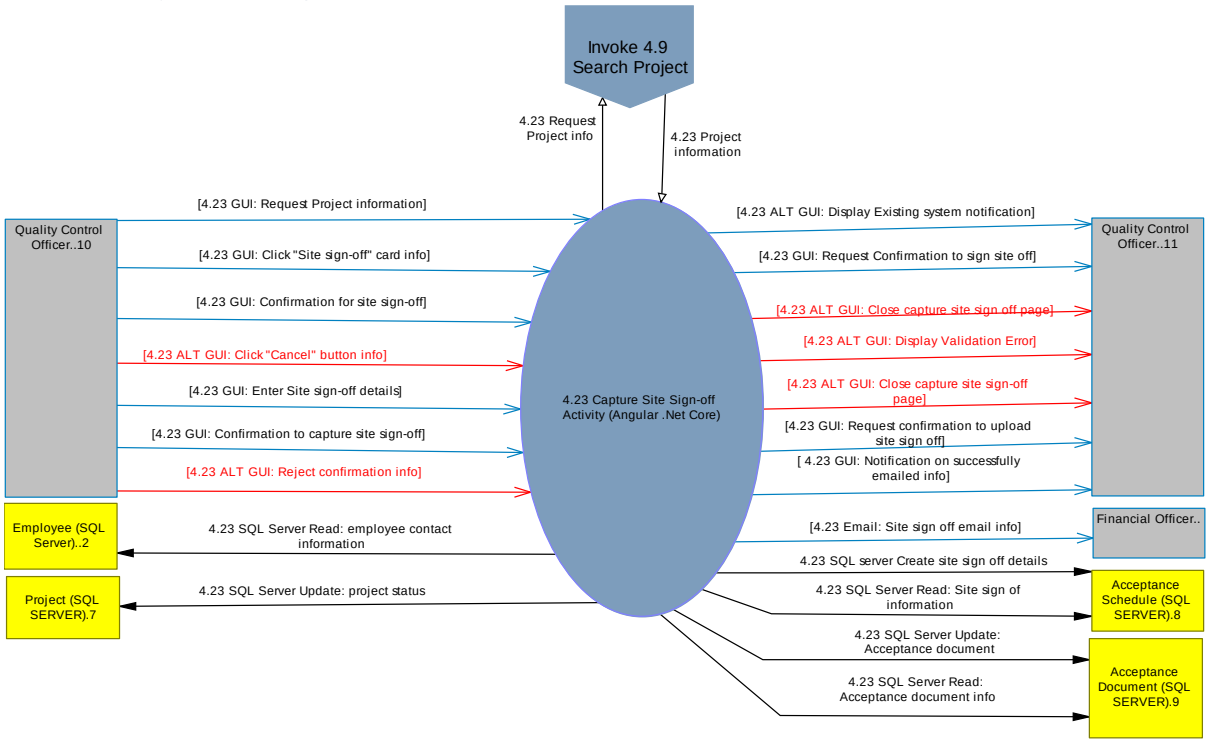


Figure 100: Middle level diagram

2.3.2.4.24 Search Site Sign-off

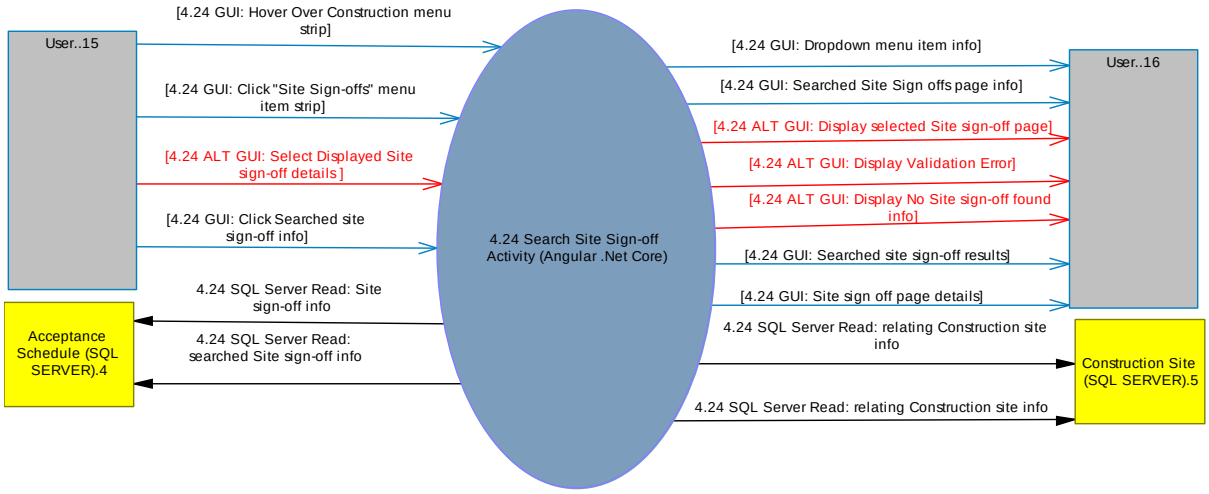


Figure 101: Middle level diagram 23 Capture Site Sign-off

2.3.2.4.25 Capture Commission report

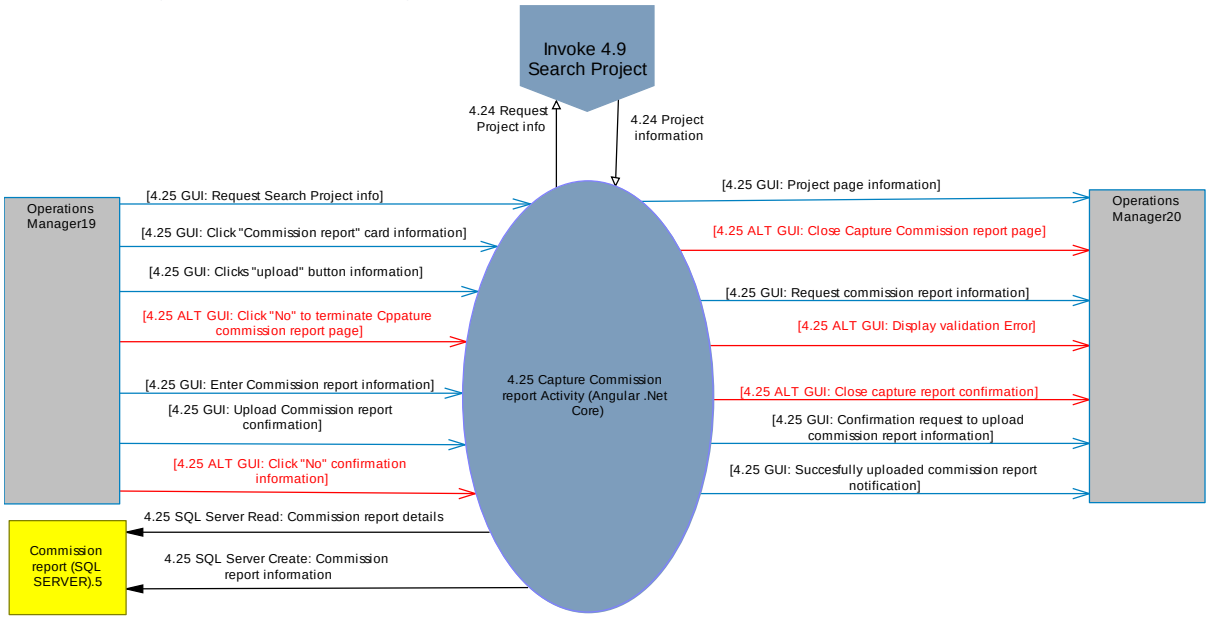


Figure 102: Middle level diagram 4.25 Capture Commission report

2.3.2.4.26 Search Commission report

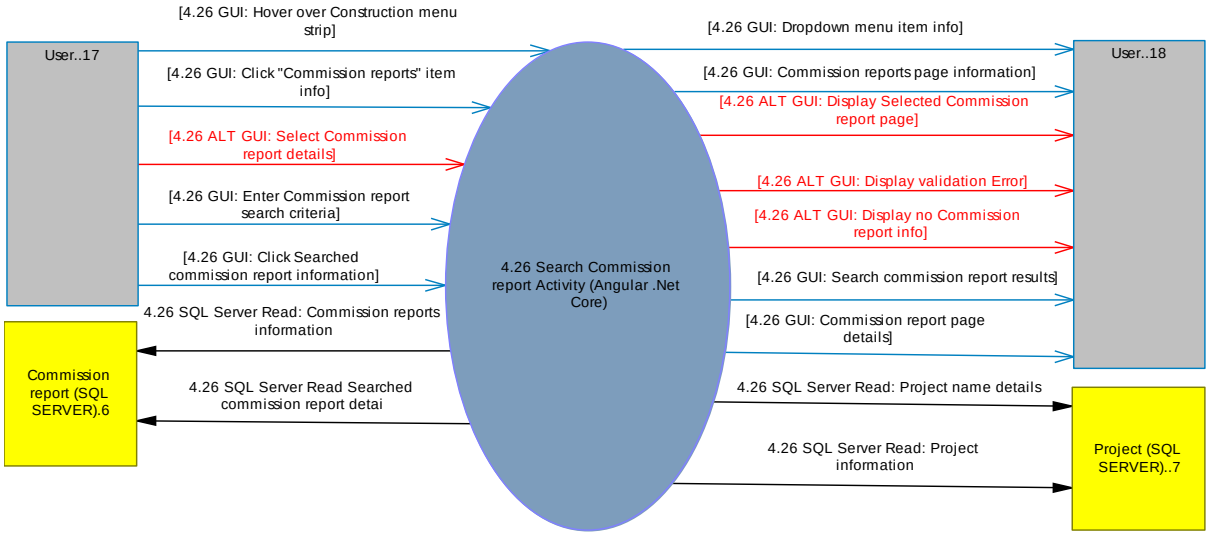


Figure 103: Middle level diagram 4.26 Search Commission report

2.3.2.4.27 Create Site Type

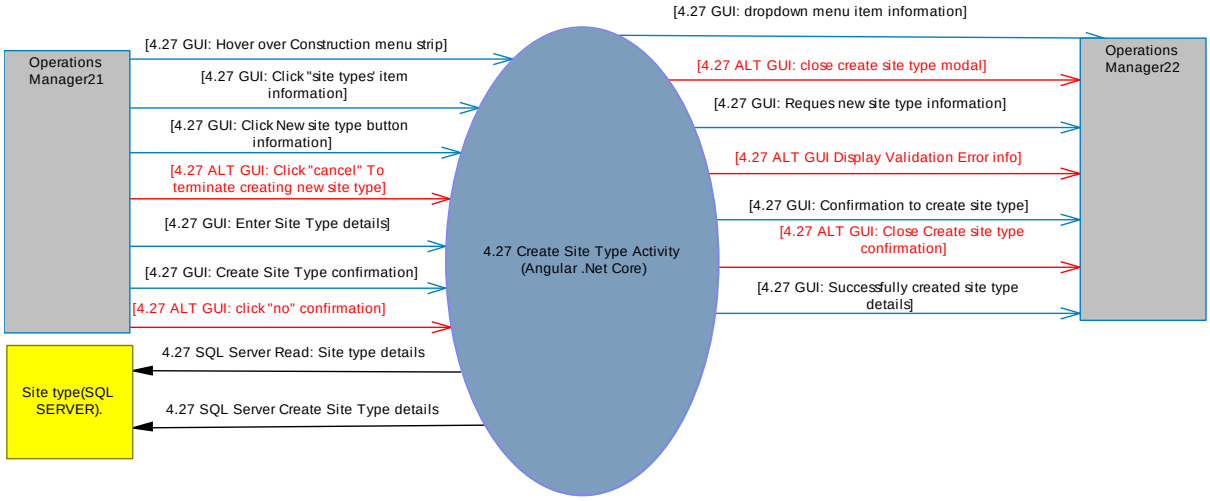


Figure 104: Middle level diagram 4.27 Create Site Type

2.3.2.4.28 Search Site type

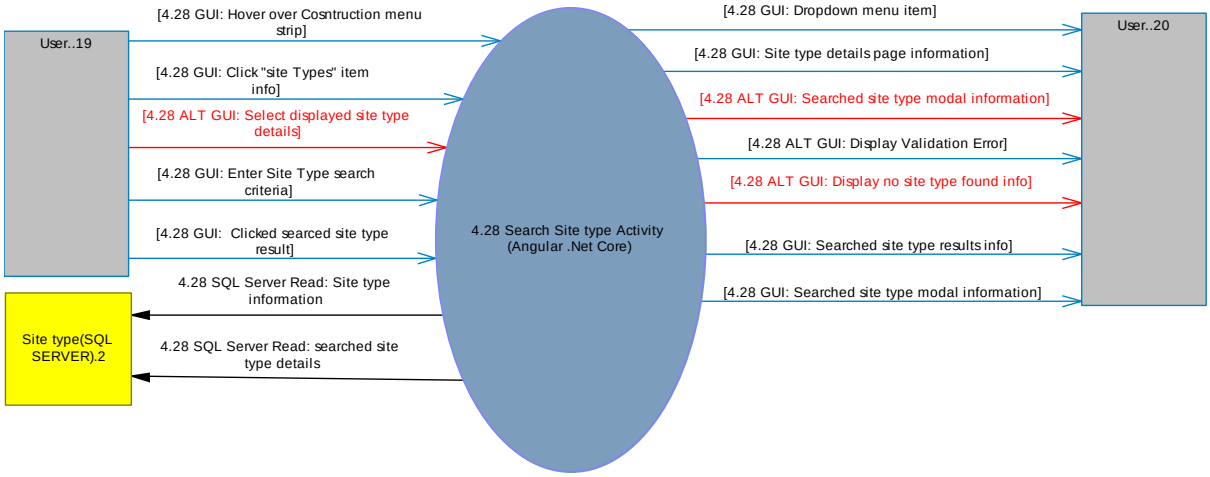


Figure 105: Middle level diagram 4.28 Search Site type

2.3.2.4.29 Edit Site Type

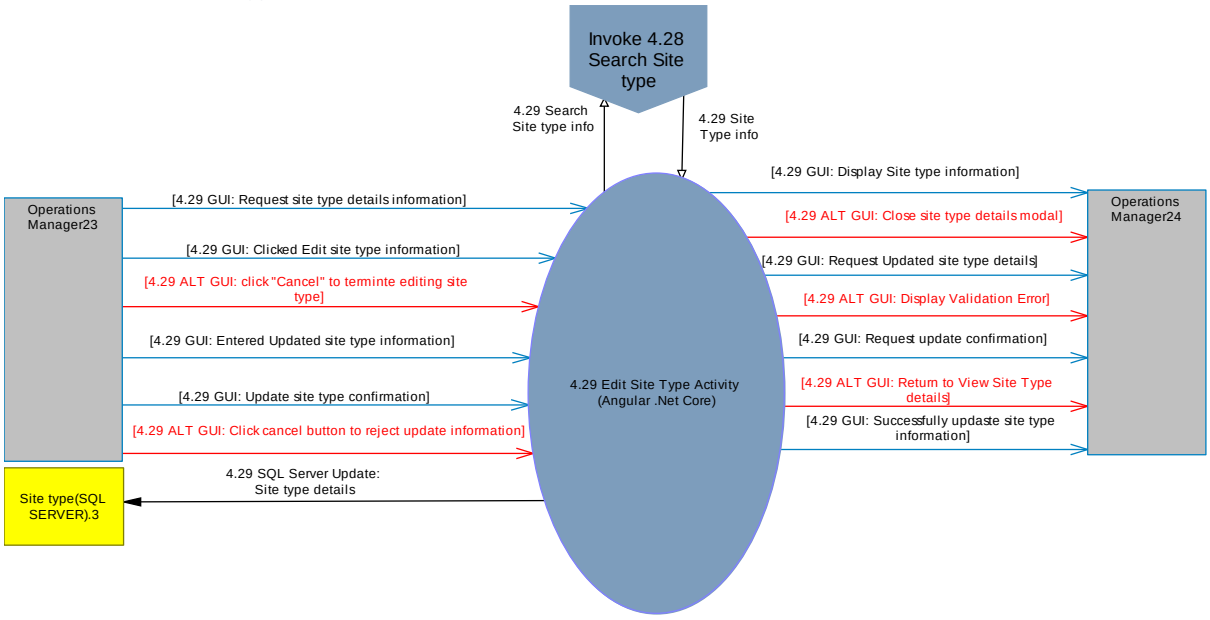


Figure 106: Middle level diagram 4.29 Edit Site Type

2.3.2.5. Invoicing Subsystem

2.3.2.5.1 Search Invoice

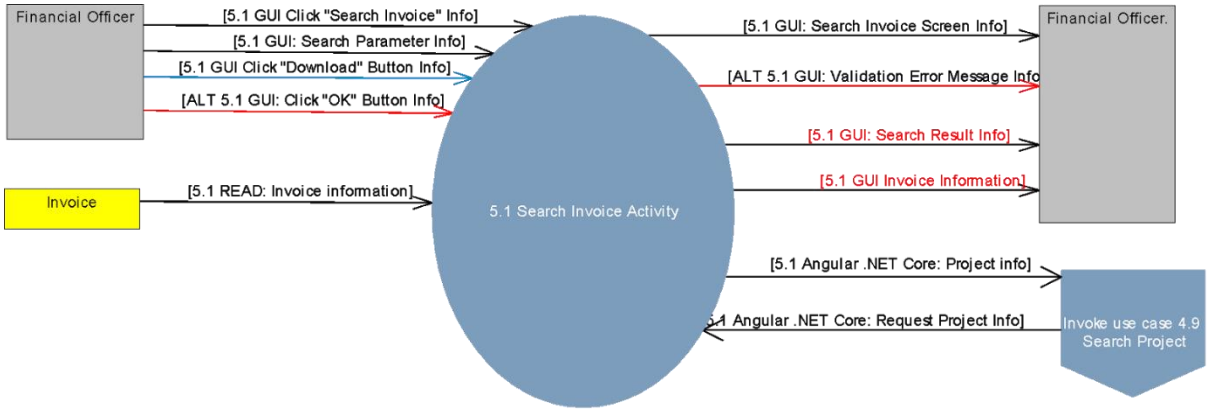


Figure 107: Middle level diagram 5.1 Search invoice

2.3.2.5.2 Send Invoice

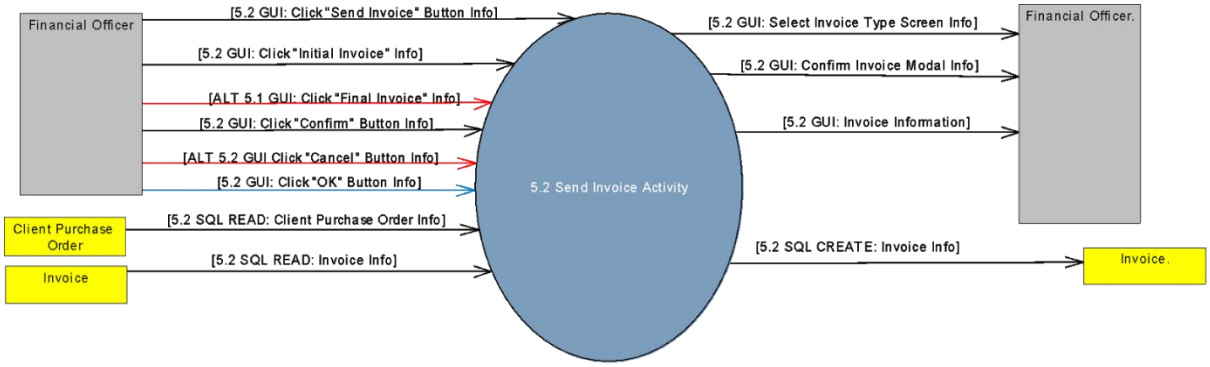


Figure 108: Middle level diagram 5.2 Send invoice

2.3.2.5.3 Capture Proof of Payment

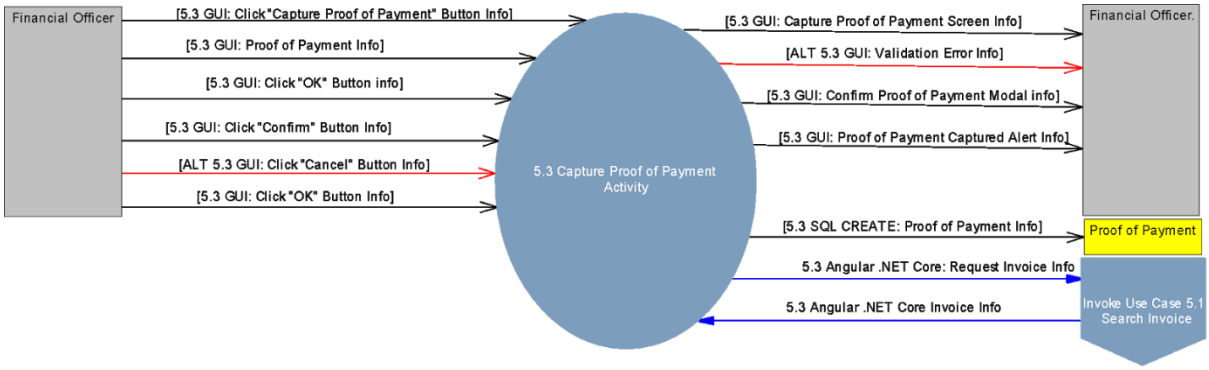


Figure 109: Middle level diagram 5.3 Capture proof of payment

2.3.2.5.4 Capture Claim

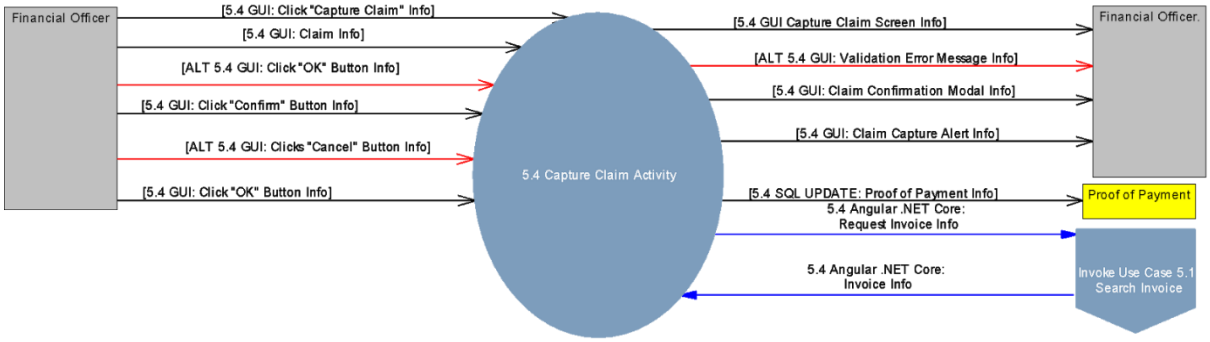


Figure 110: Middle level diagram 5.4 Capture claim

2.3.2.5.5 Search Proof of Payment

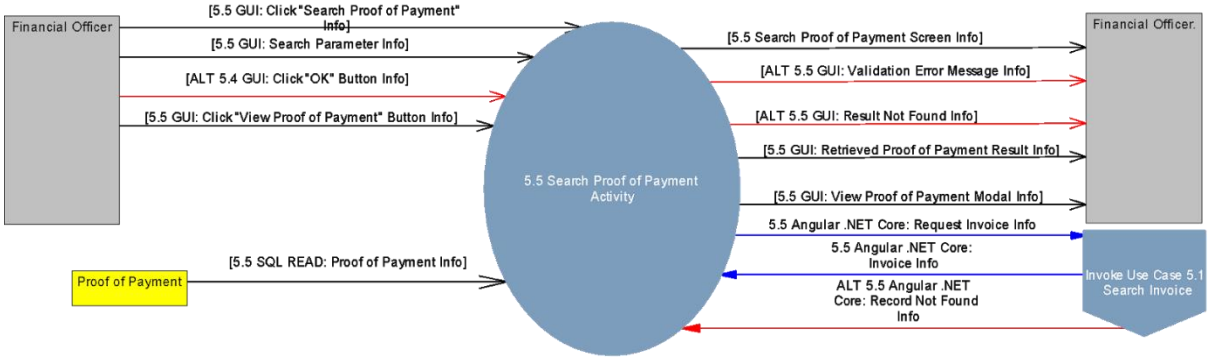


Figure 111: Middle level diagram 5.5 Search proof of payment

2.3.2.5.6 Search Invoice Type

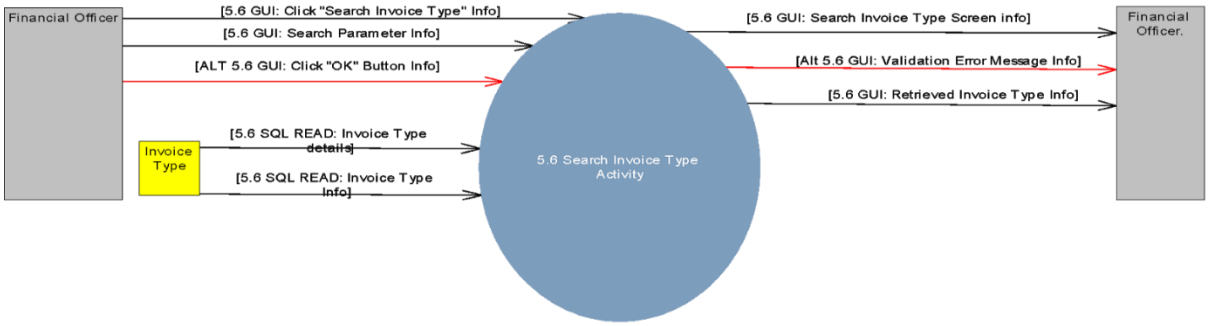


Figure 112: Middle level diagram 5.6 Search invoice type

2.3.2.5.7 Create Invoice Type

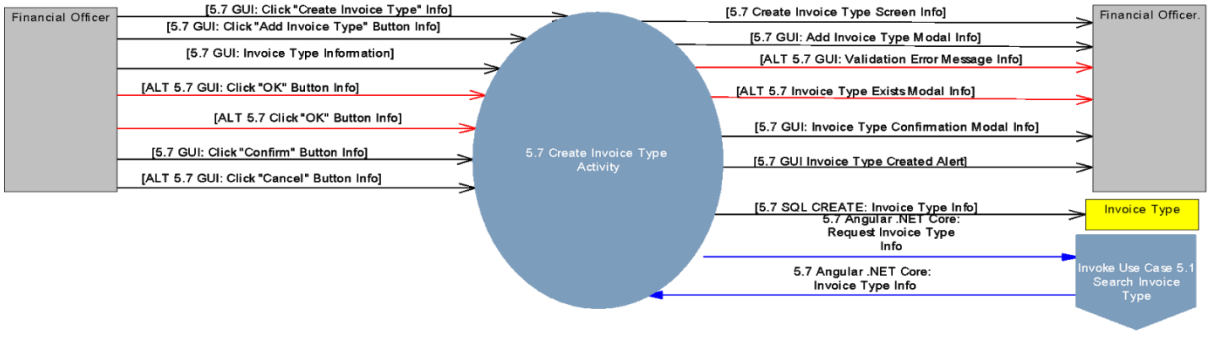


Figure 113: Middle level diagram 5.7 Create invoice type

2.3.2.6. Human Resource Subsystem

2.3.2.6.1 Add employee

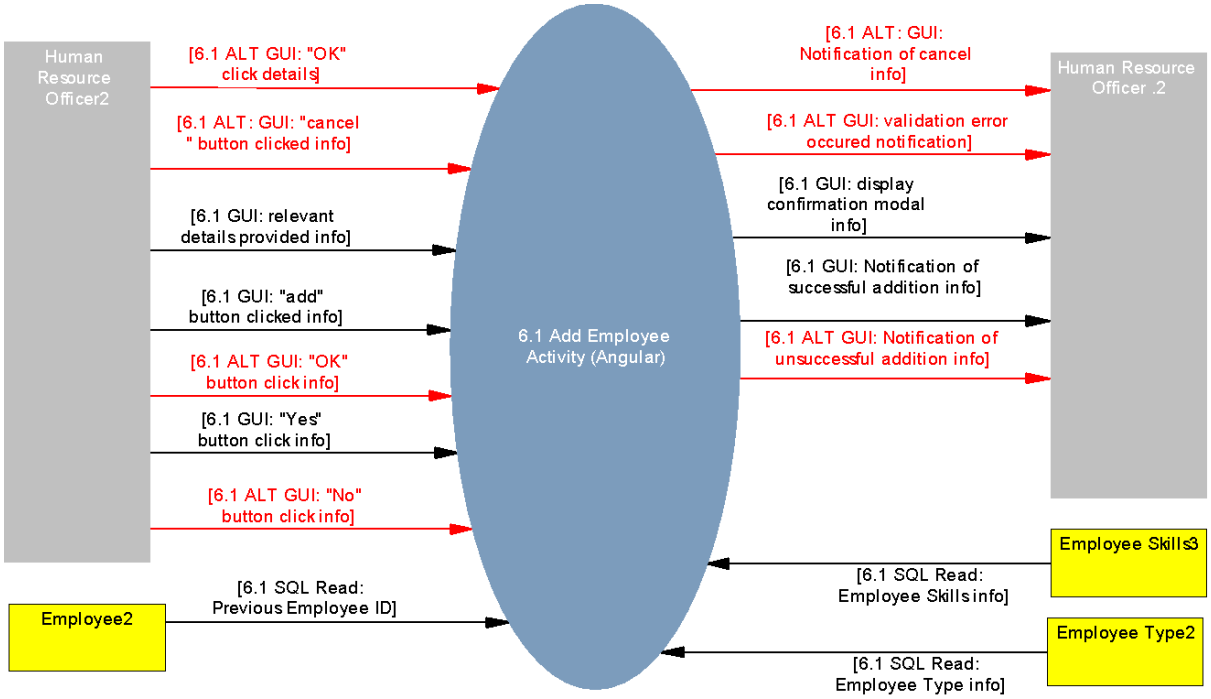


Figure 114: Middle level diagram 6.1 Add employee

2.3.2.6.2 Edit employee

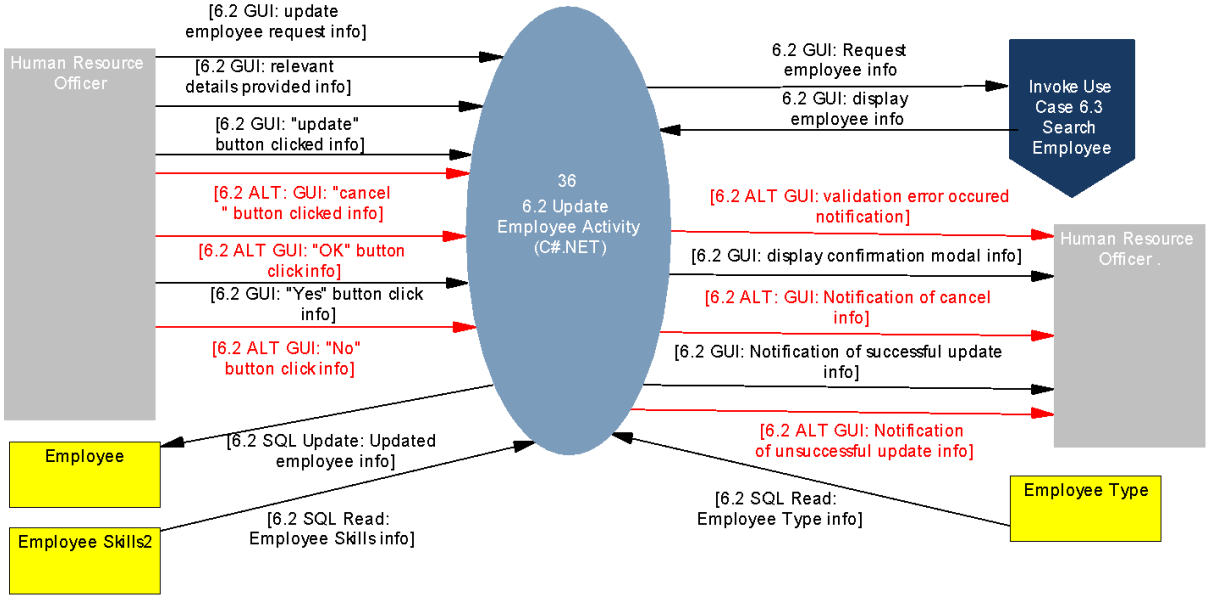


Figure 115: Middle level diagram 6.2 Edit employee

2.3.2.6.3 Search employee

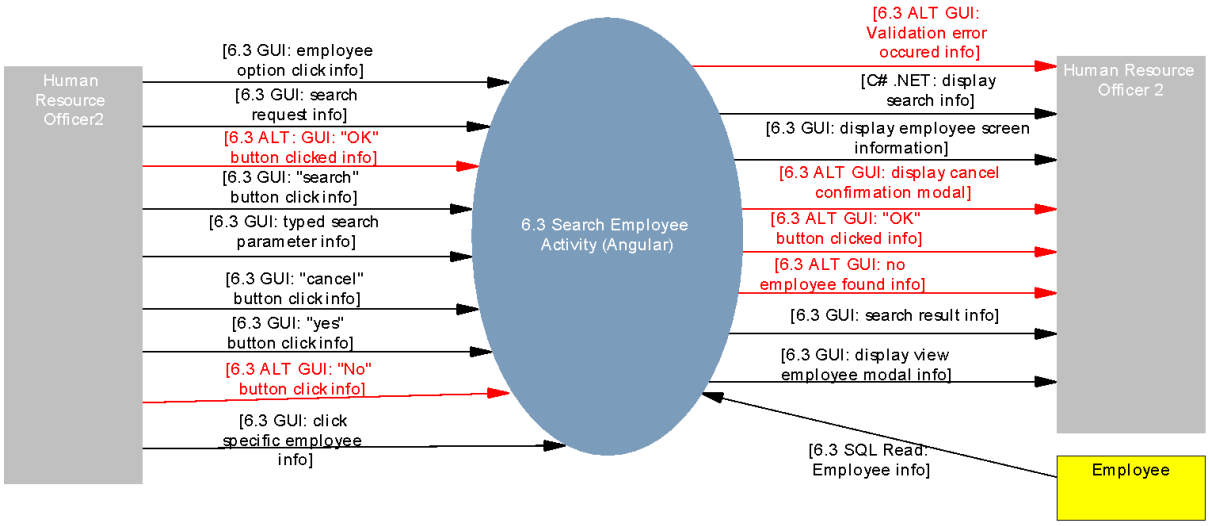


Figure 116: Middle level diagram 6.3 Search employee

2.3.2.6.4 Upload Employee Document

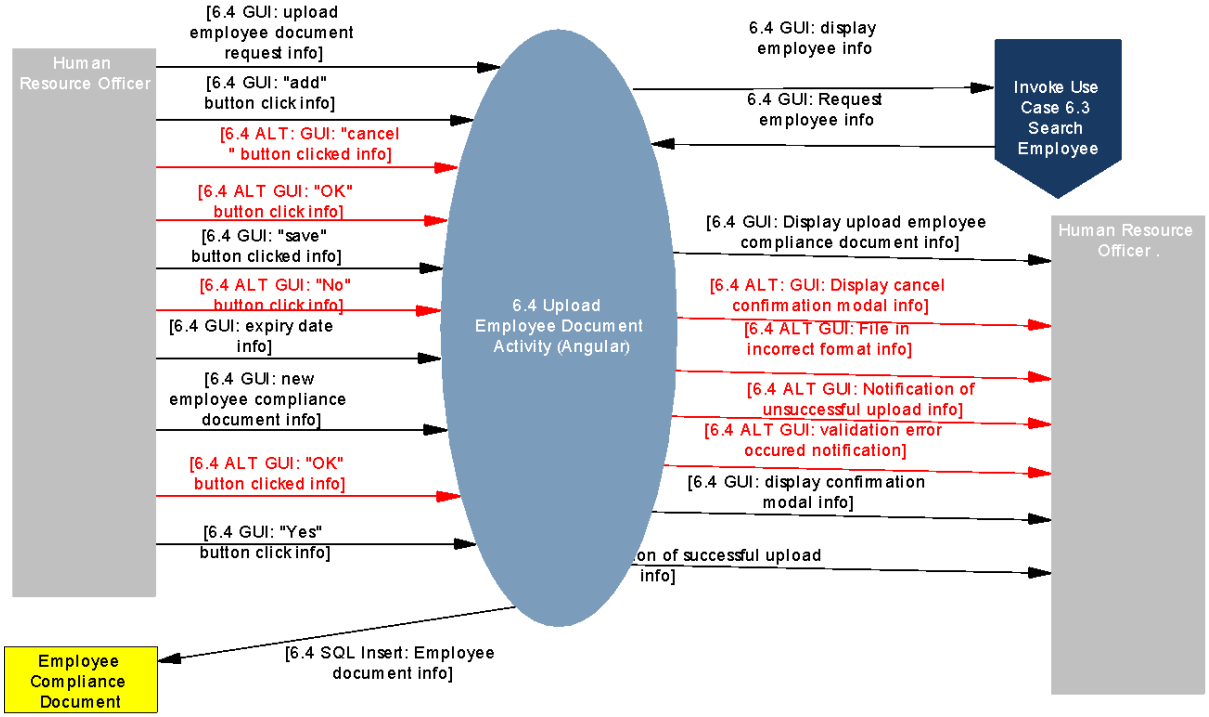


Figure 117: Middle level diagram 6.4 Upload Employee document

2.3.2.6.5 Edit Employee Document

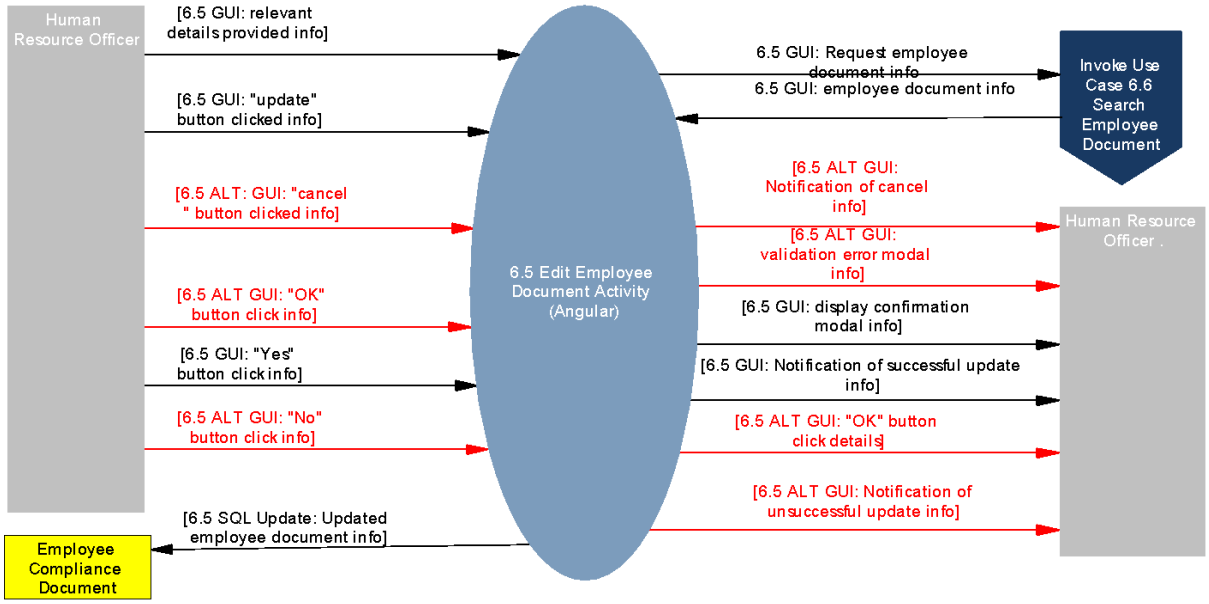


Figure 118: Middle level diagram 6.5 Edit employee documents

2.3.2.6.6 Search Employee document

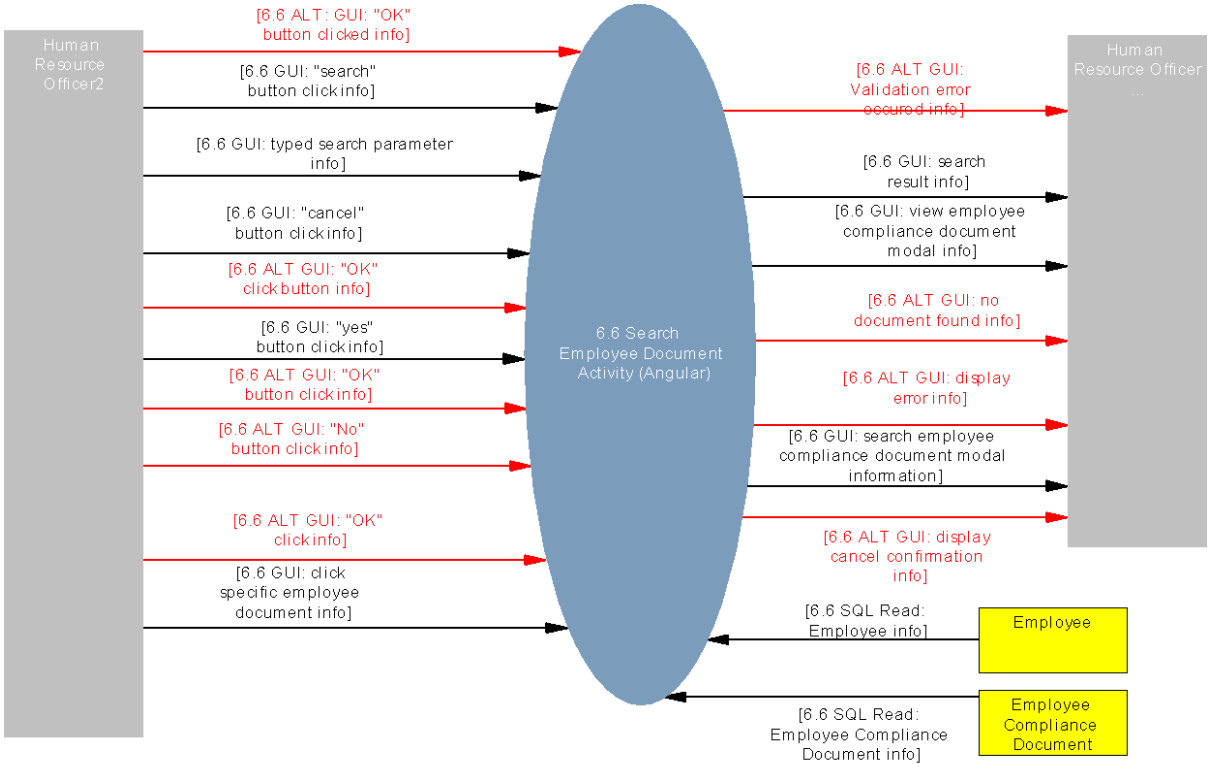


Figure 119: Middle level diagram 6.6 Search employee documents

2.3.2.6.7 Create Account

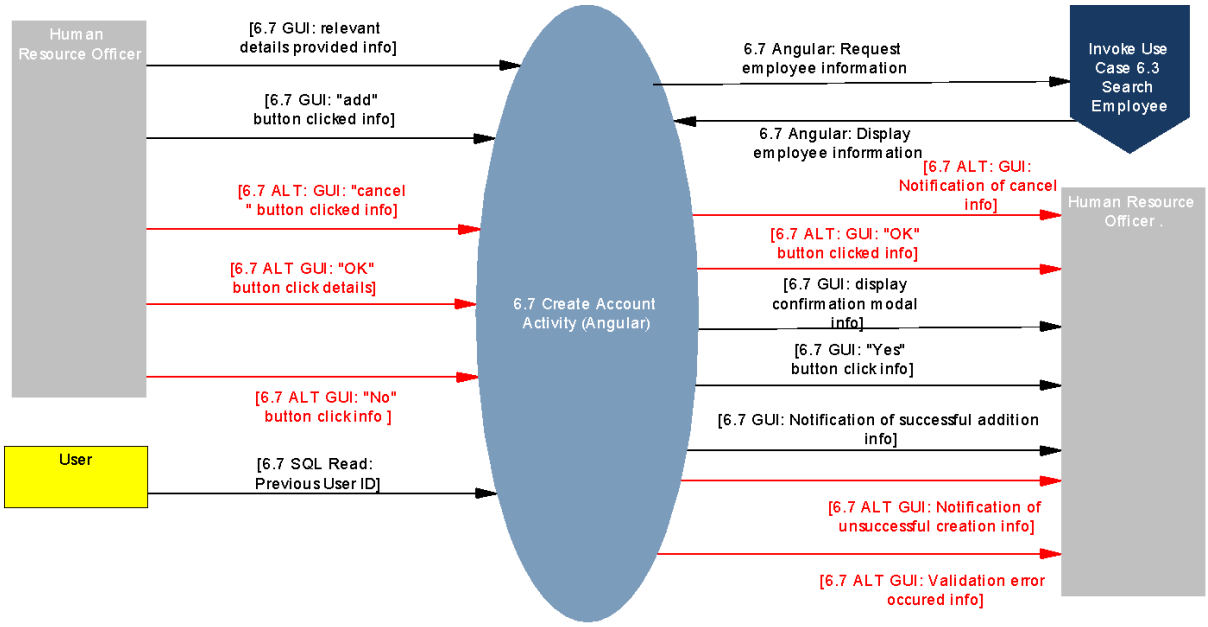


Figure 120: Middle level diagram 6.7 Create account

2.3.2.6.8 Edit Account

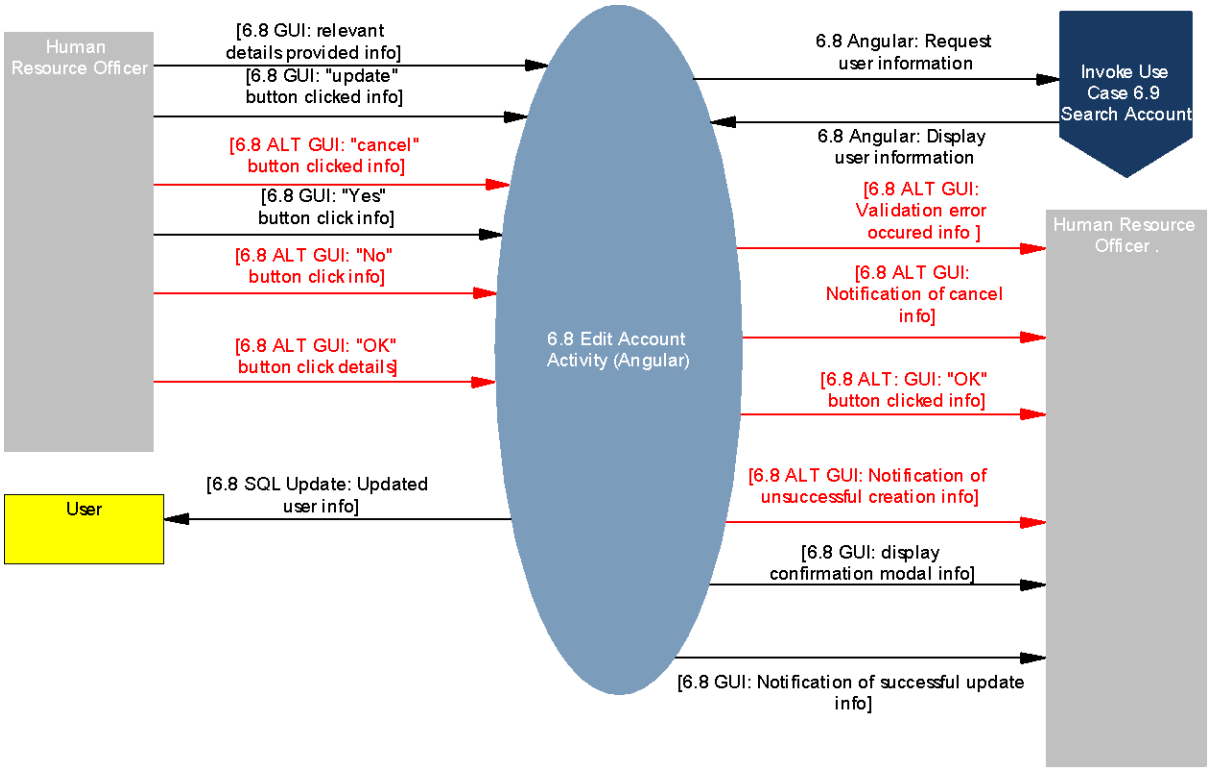


Figure 121: Middle level diagram 6.8 Edit account

2.3.2.6.9 Search Account

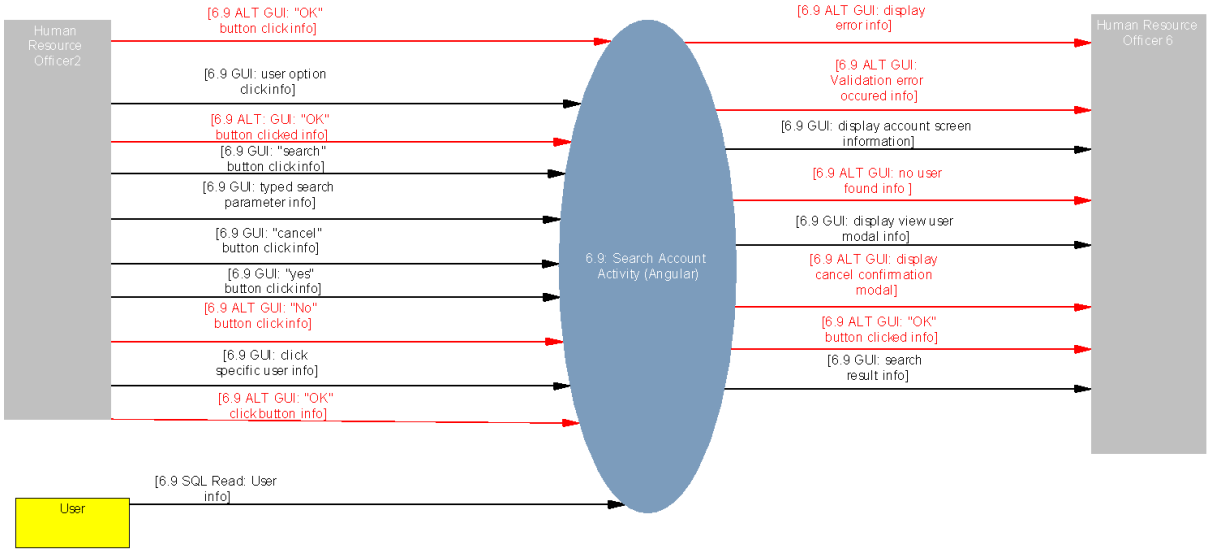


Figure 122: Middle level diagram 6.9 Search account

2.3.2.6.10 Create Employee Type

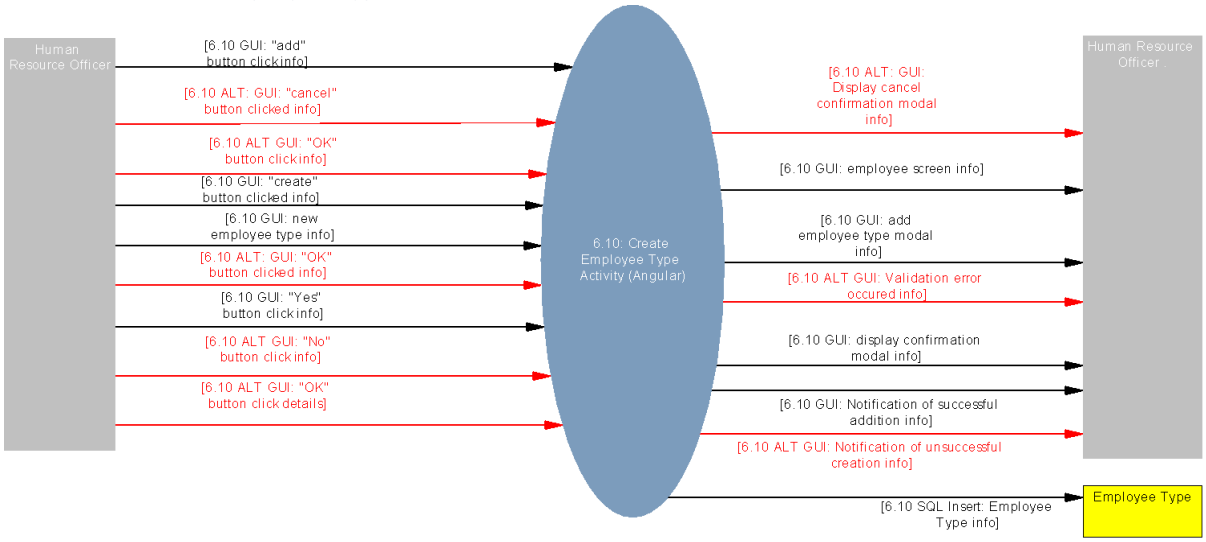


Figure 123: Middle level diagram 6.10 Create employee type

2.3.2.6.11 Search Employee Type

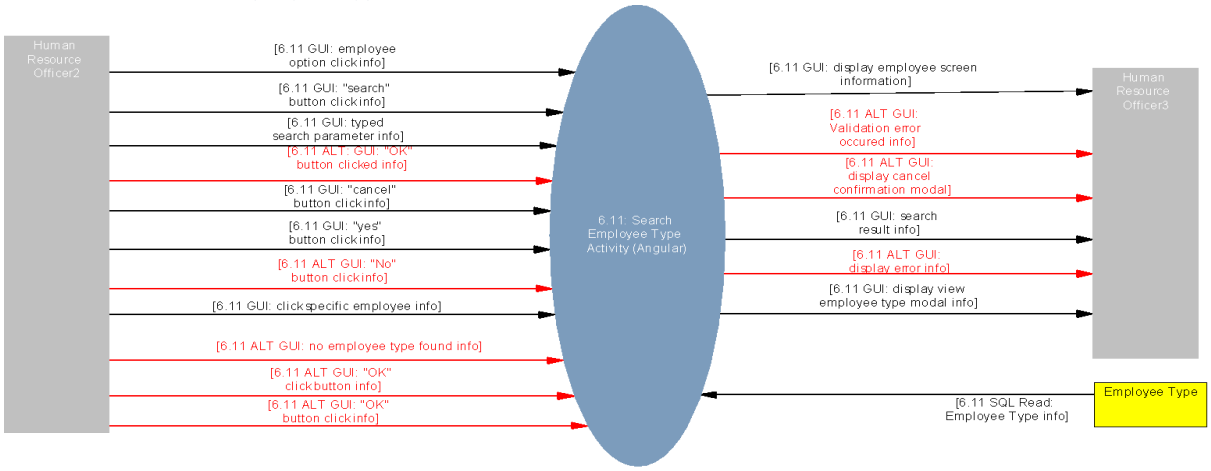


Figure 124: Middle level diagram 6.11 Search employee type

2.3.2.6.12 Create Employee Skill

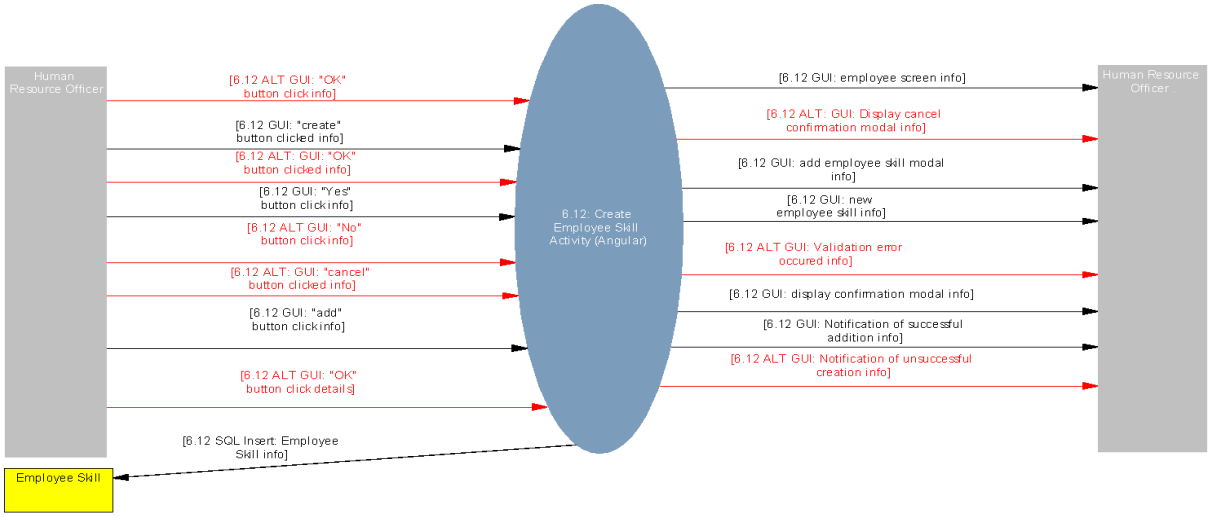


Figure 125: Middle level diagram 6.12 Create employee skills

2.3.2.6.13 Search Employee Skill

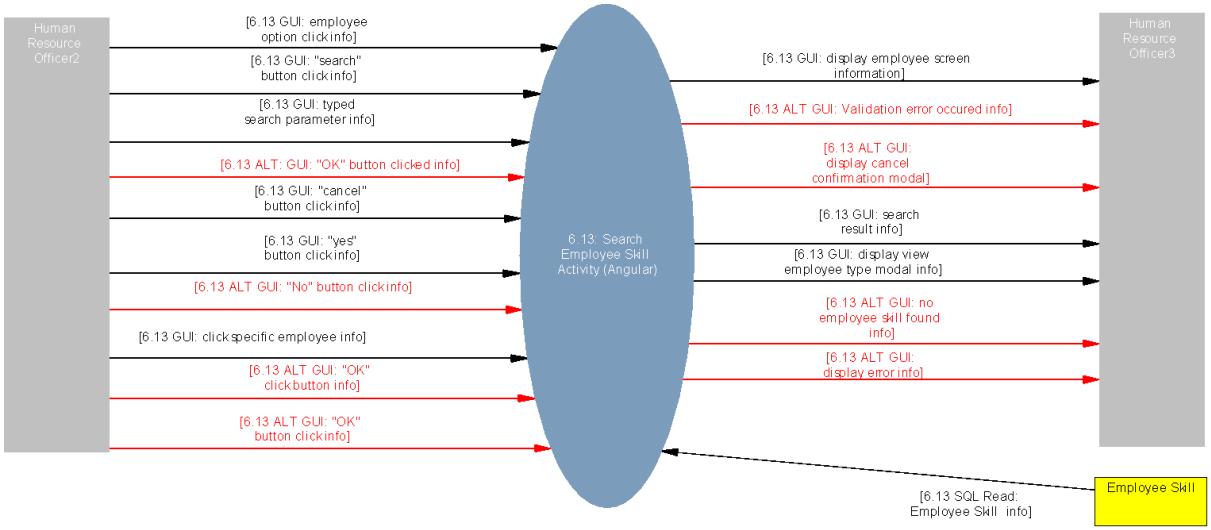


Figure 126:Middle level diagram 6.13Search employee skill

2.3.2.7. Team Subsystem

2.3.2.7.1 Create Team

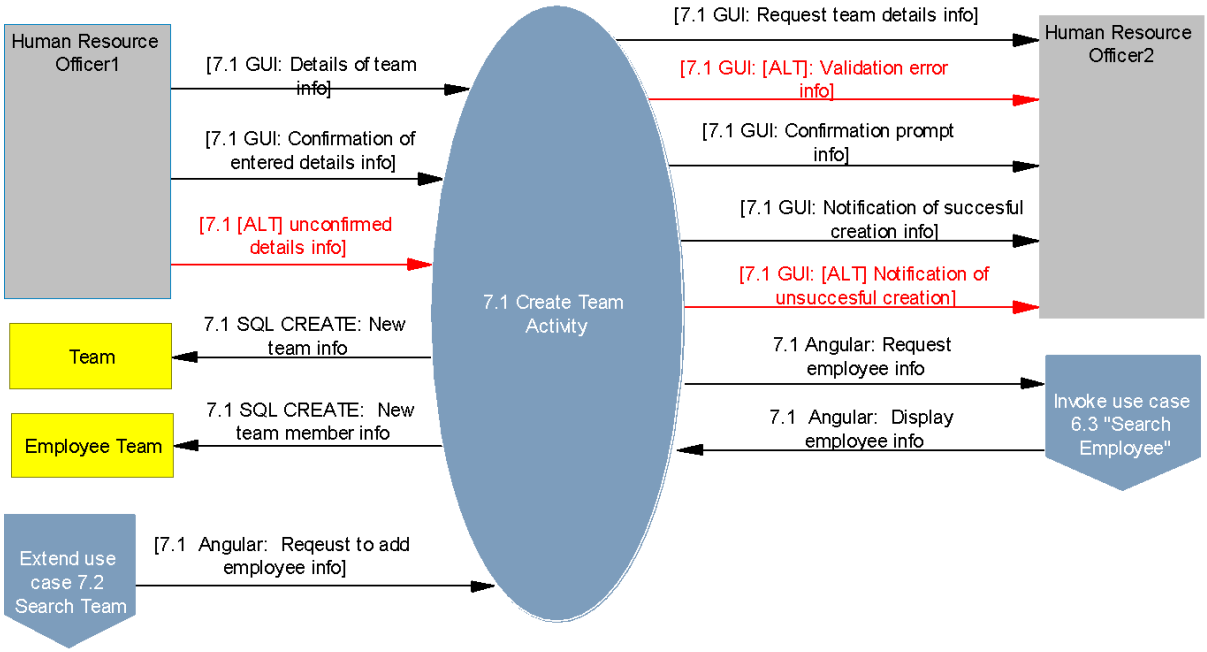


Figure 127: Middle level diagram 7.1 Create Team

2.3.2.7.2 Search Team

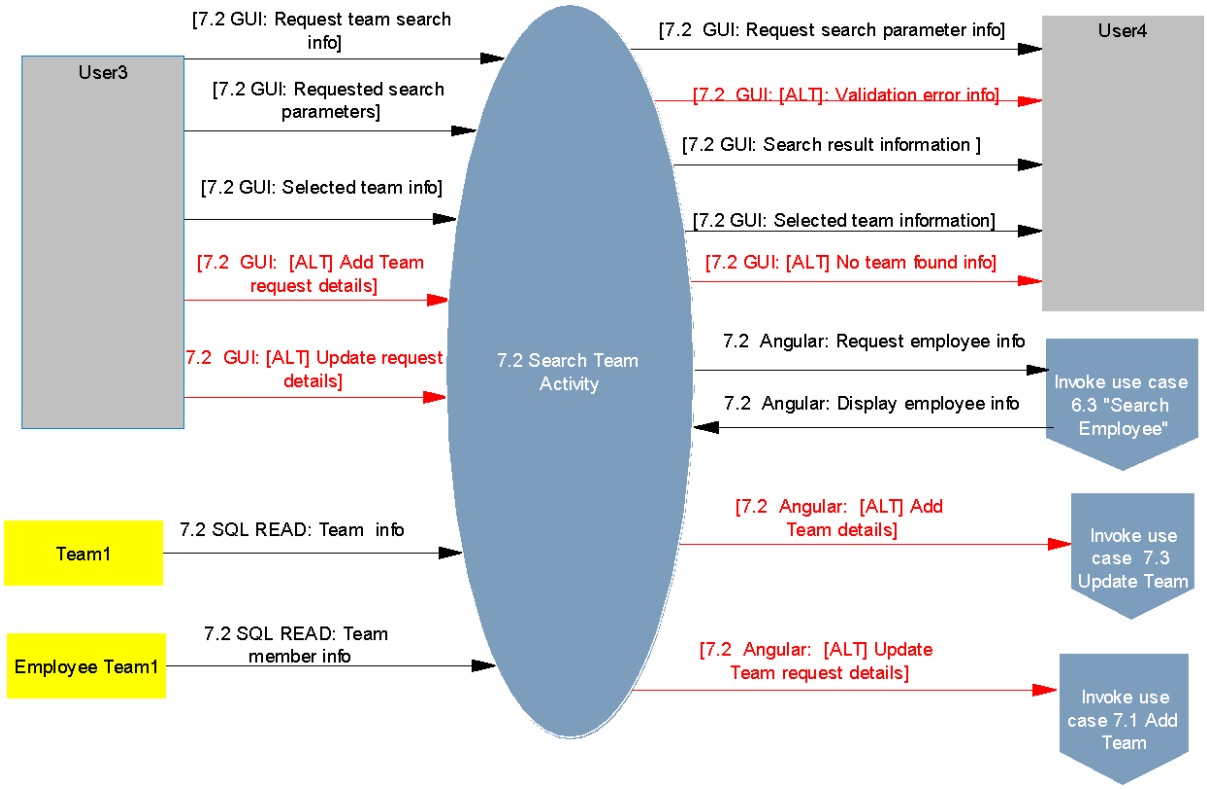


Figure 128: Middle level diagram 7.2 Search Team

2.3.2.7.3 Edit Team

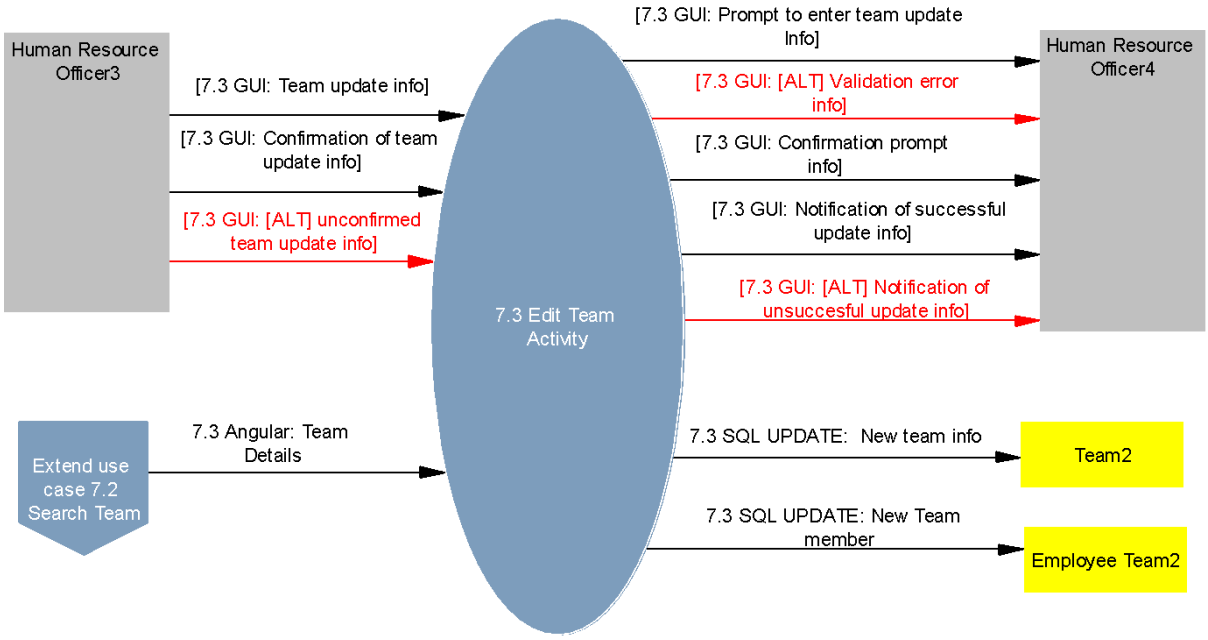


Figure 129: Middle level diagram 7.3 Edit team

2.3.2.8. Supplier Order Subsystem

2.3.2.8.1 Search Project Line-Item List

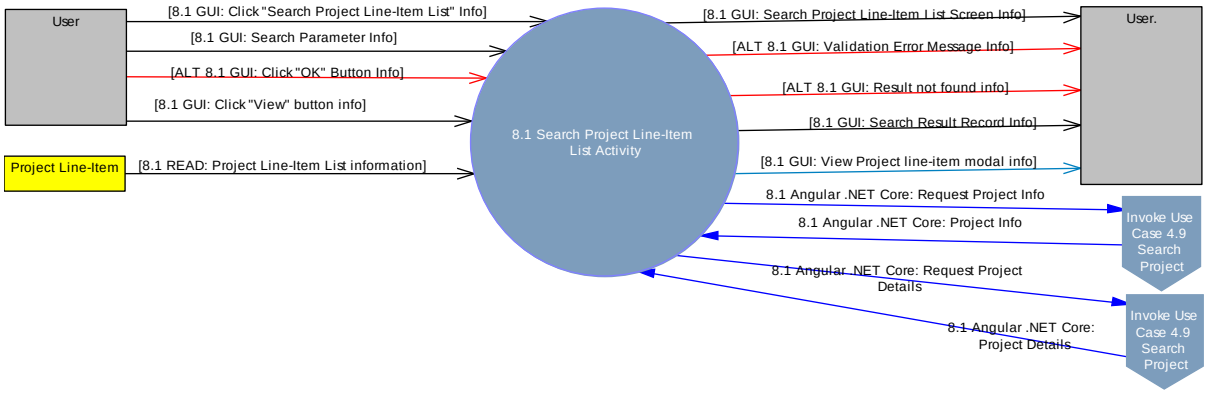


Figure 130: Middle Level diagram- 8.1 Search Project Line-Item List Activity

2.3.2.8.2 Edit Project Line-Item List

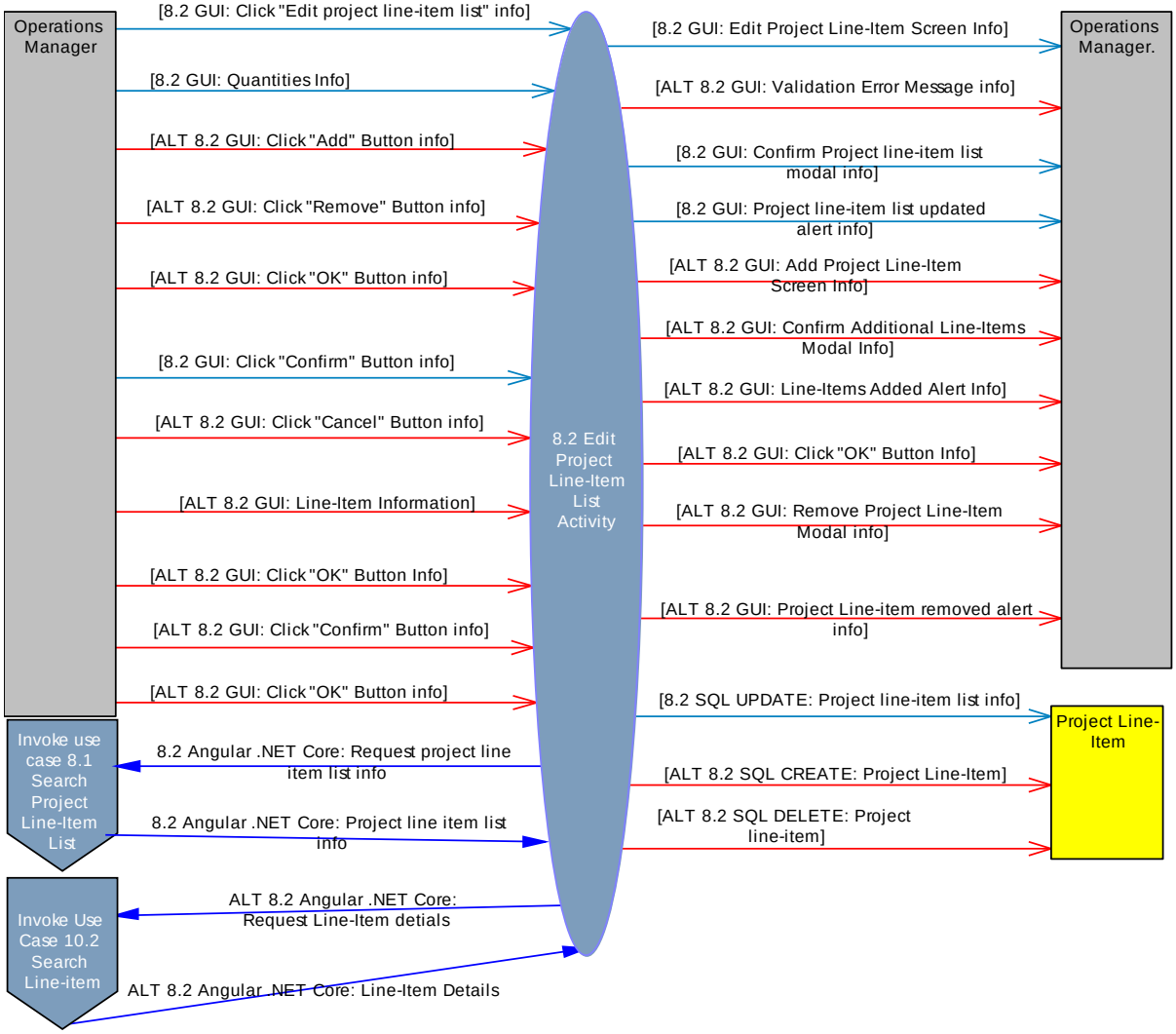


Figure 131: Middle Level diagram- 8.2 Edit Project Line-Item List Activity

2.3.2.8.3 Create Supplier Purchase Order

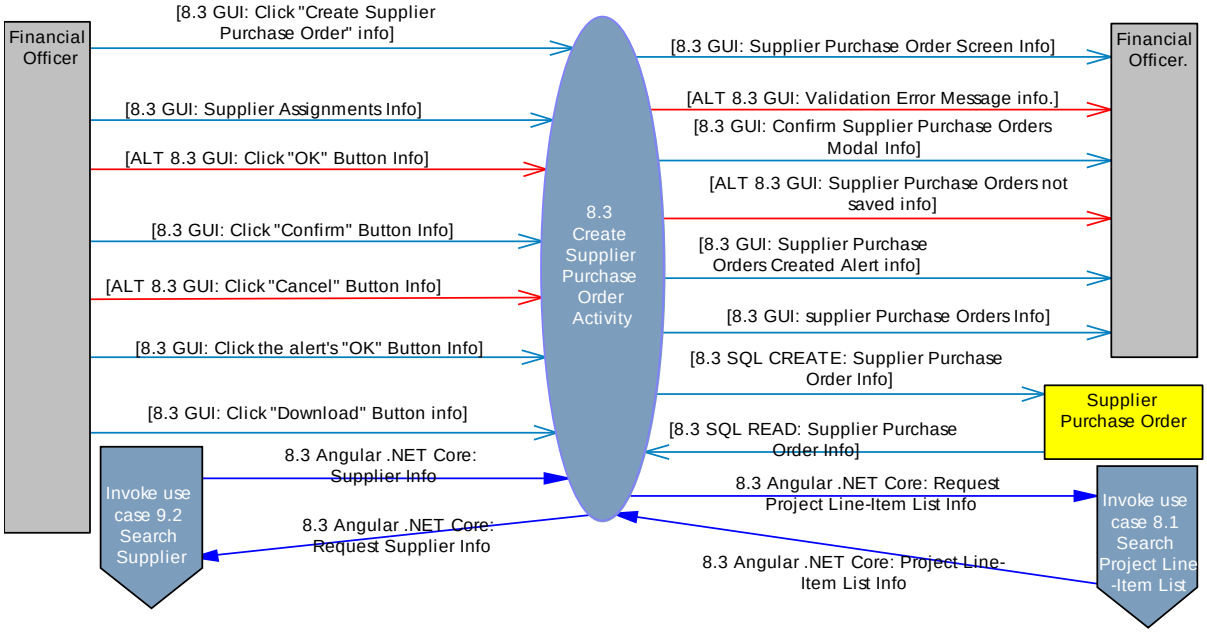


Figure 132: Middle Level diagram- 8.3 Create Supplier Purchase Order Activity

2.3.2.8.4 Capture Supplier Quote

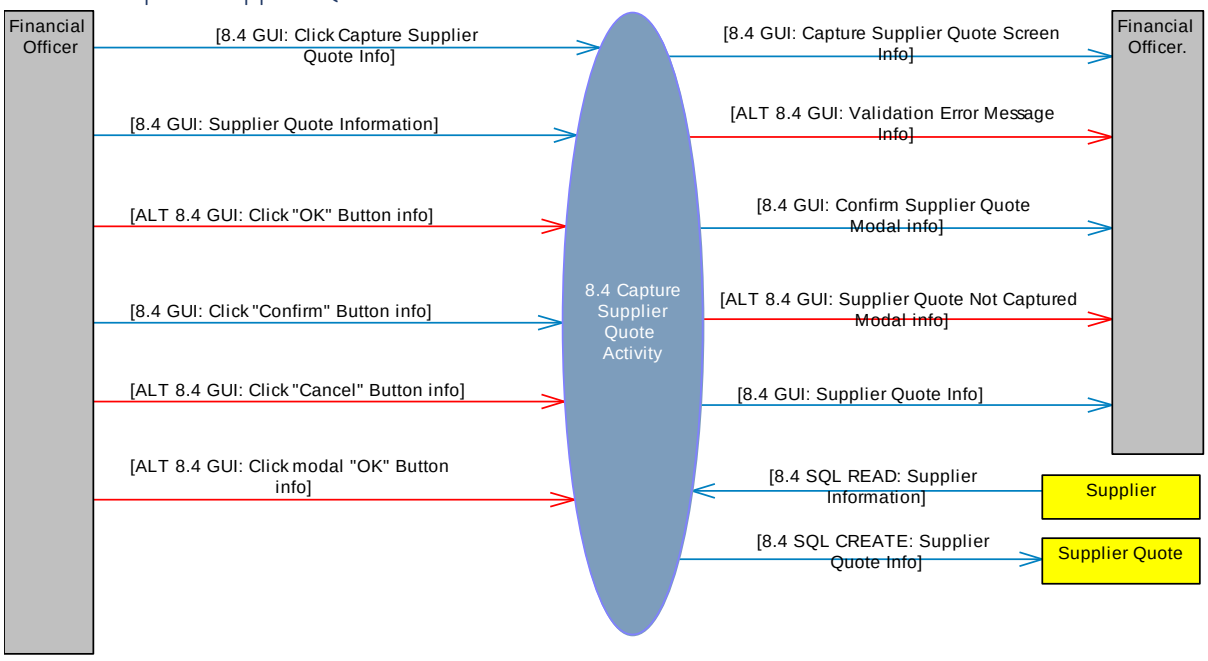


Figure 133: Middle Level diagram- 8.4 Capture Supplier Quote Activity

2.3.2.8.5 Search Supplier Quote

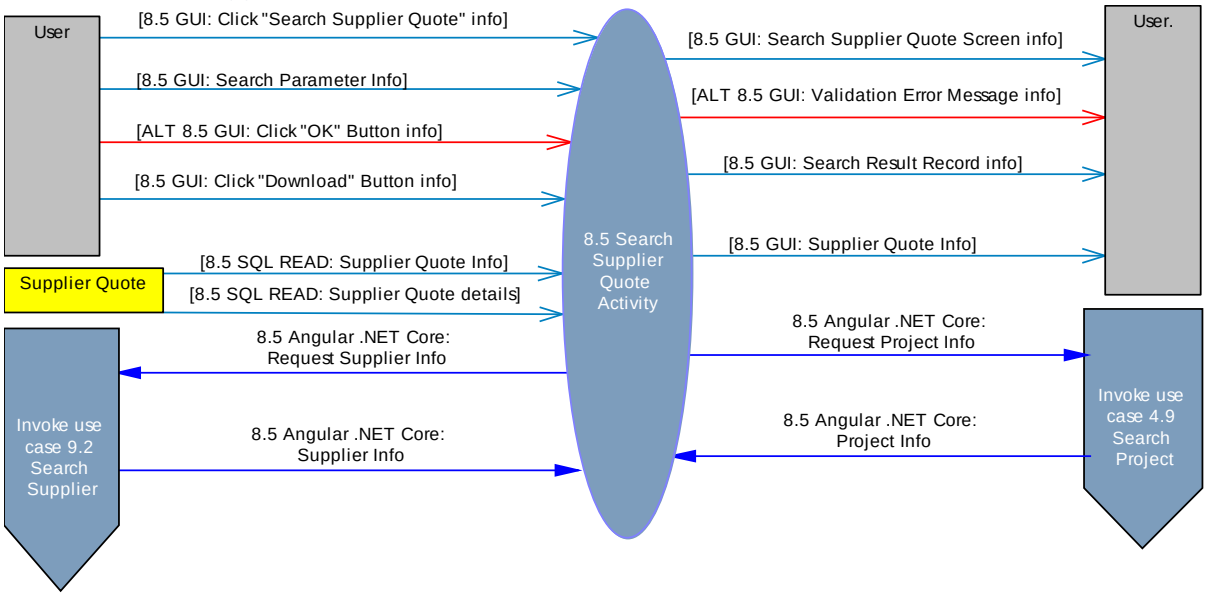


Figure 134: Middle Level diagram- 8.5 Search Supplier Quote Activity

2.3.2.8.6 Search Supplier Purchase Order



Figure 135: Middle Level diagram- 8.6 Search Supplier Purchase Order Activity

2.3.2.9. Supplier Subsystem

2.3.2.9.1 Add Supplier

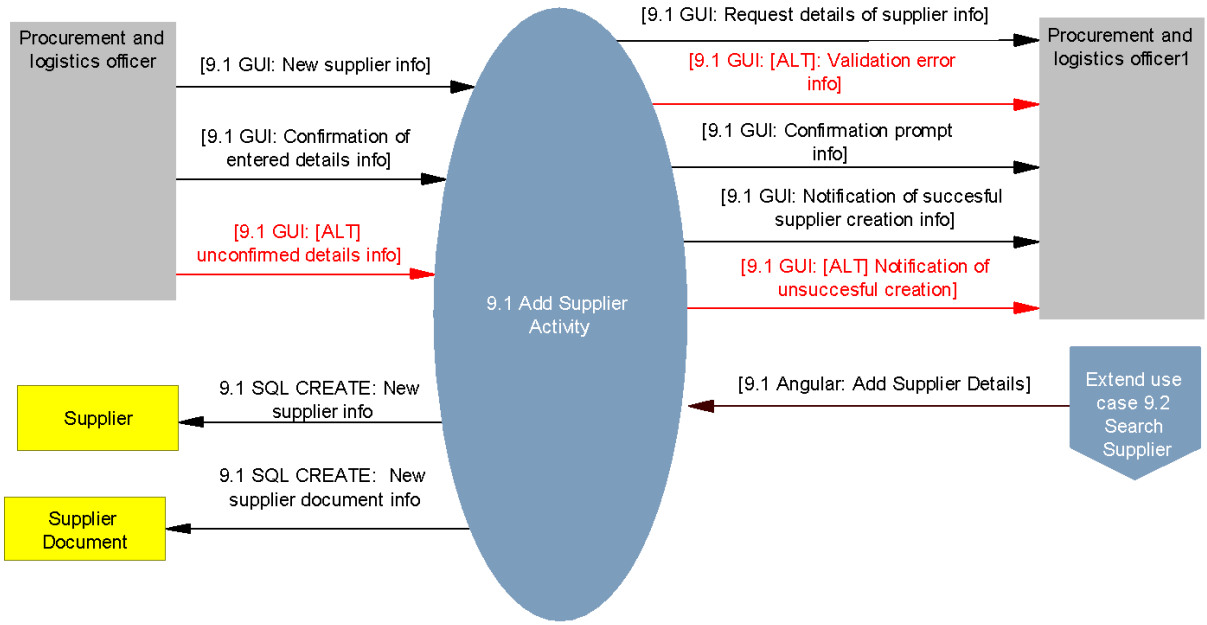


Figure 136: Middle level diagram 9.1 Add supplier

2.3.2.9.2 Search Supplier

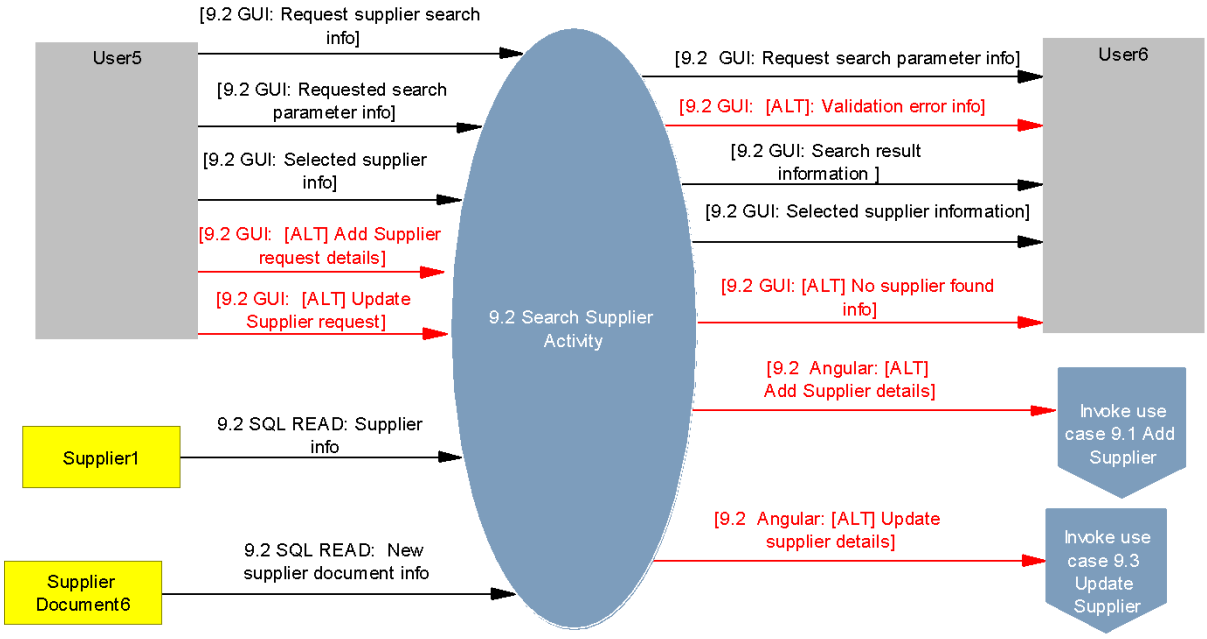


Figure 137: Middle level diagram 9.2 Search supplier

2.3.2.9.3 Edit Supplier

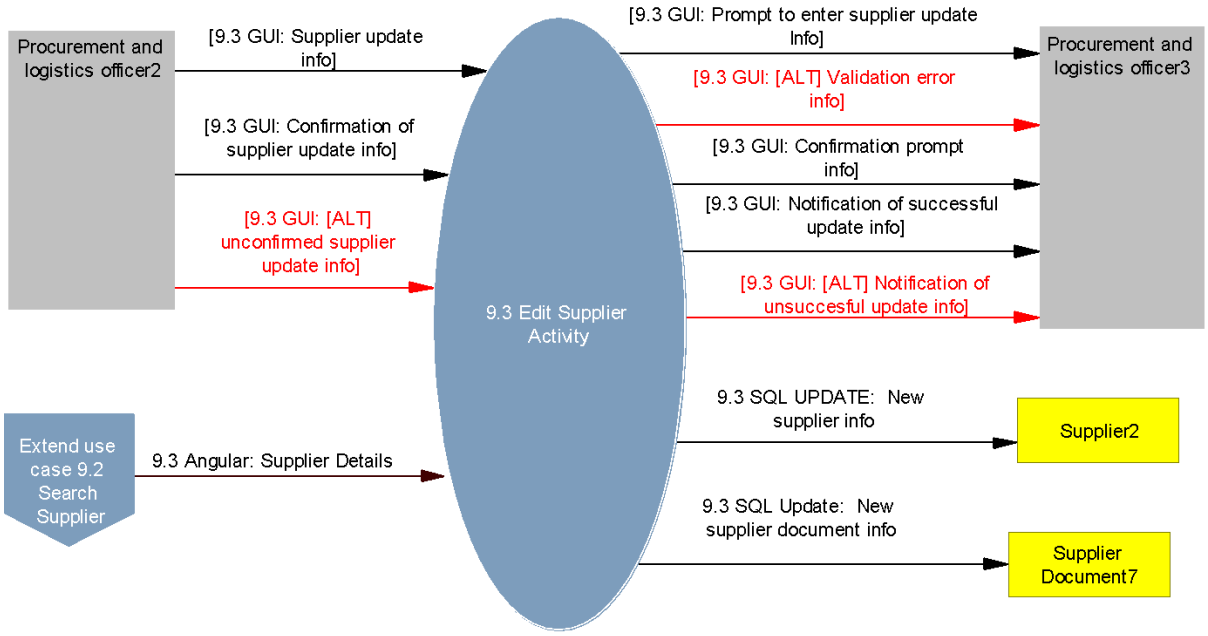


Figure 138: Middle level diagram 9.3 Edit Supplier

2.3.2.10. Inventory Management Subsystem

2.3.2.10.1 Add-Line Item

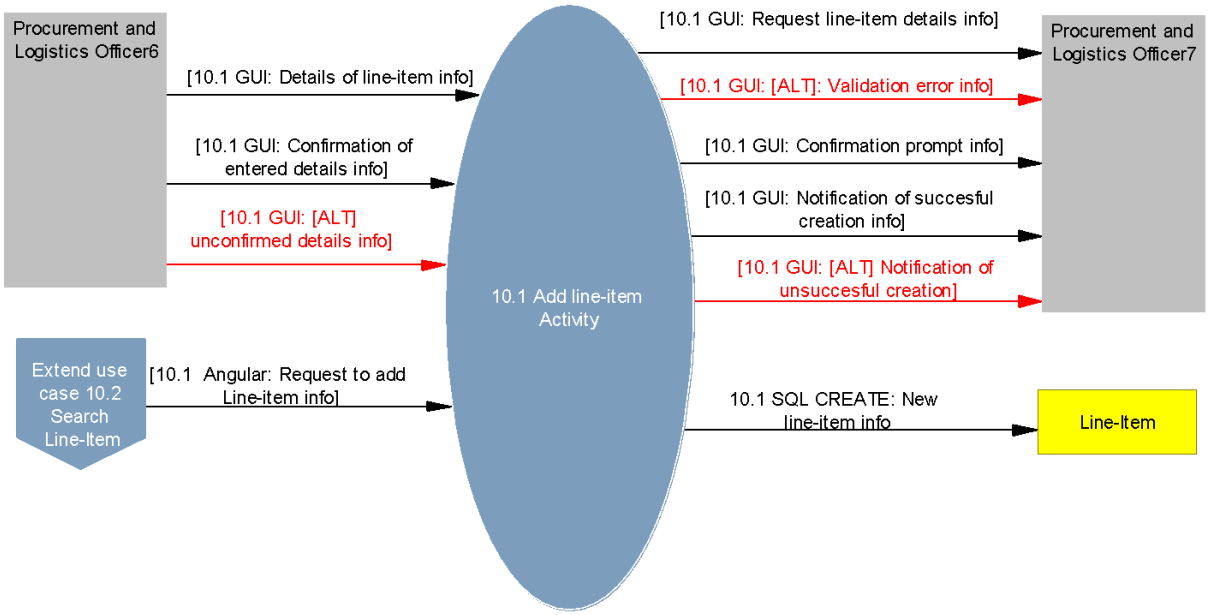


Figure 139: Middle level diagram 10.1 Add-line item

2.3.2.10.2 Search-Line Item

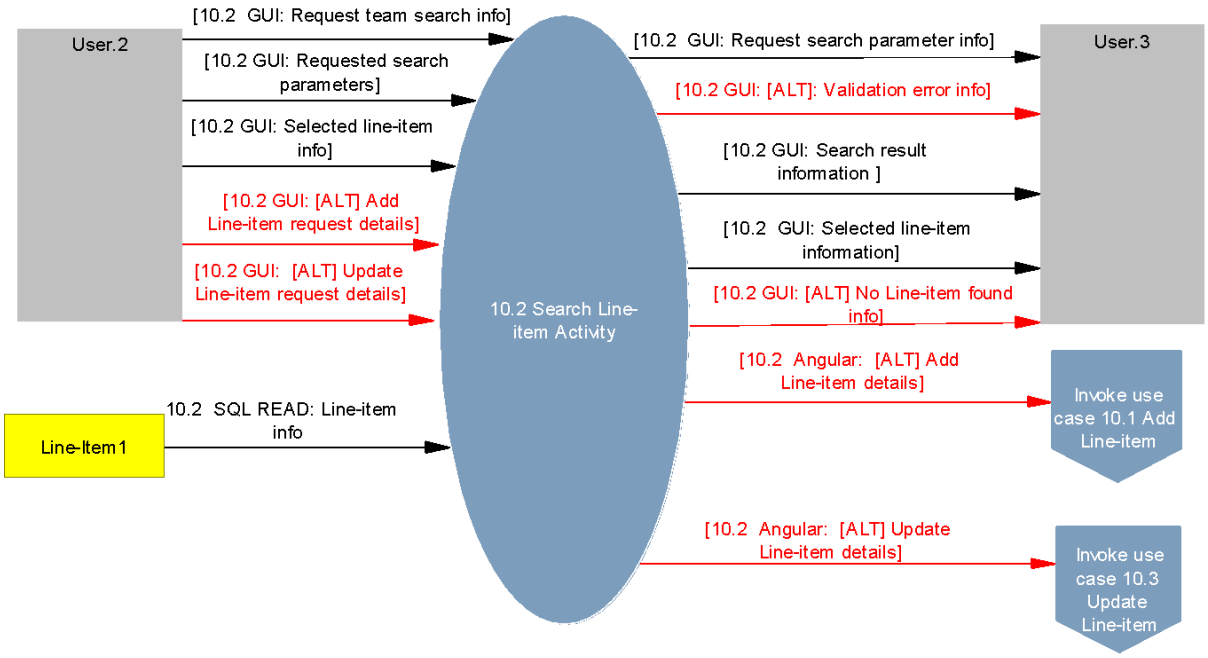


Figure 140: Middle level diagram 10.2 Search-line item

2.3.2.10.3 Edit line-item

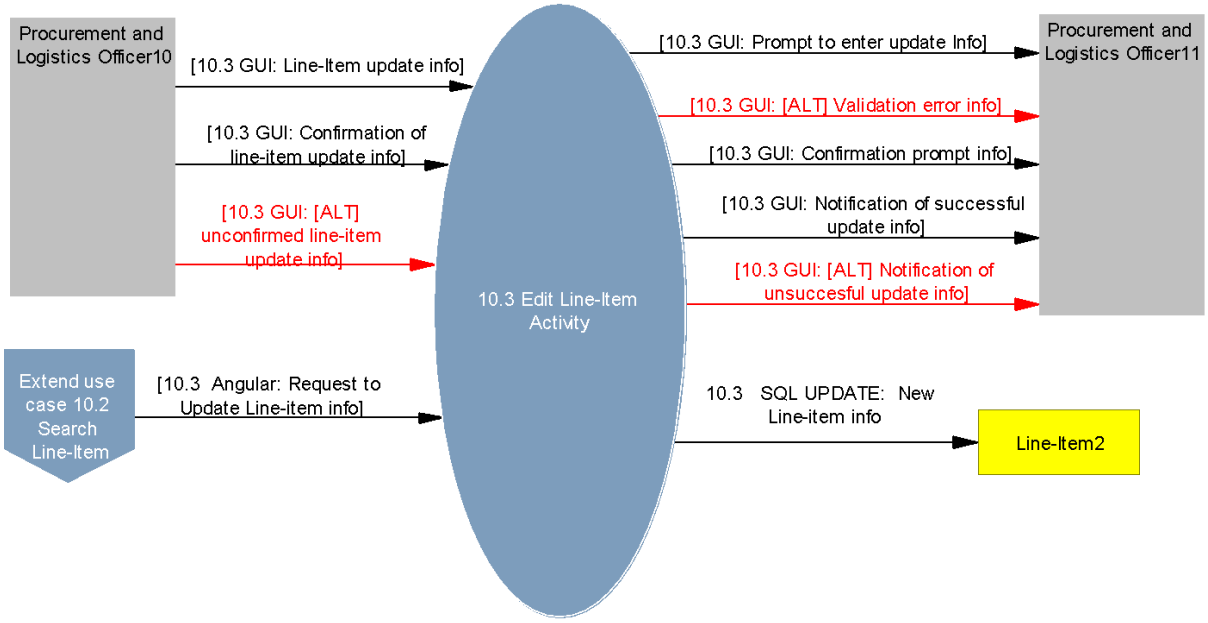


Figure 141: Middle level diagram 10.3 Edit-line -item

2.3.2.11. Reporting Subsystem

2.3.2.11.1 Generate Site Progress Report

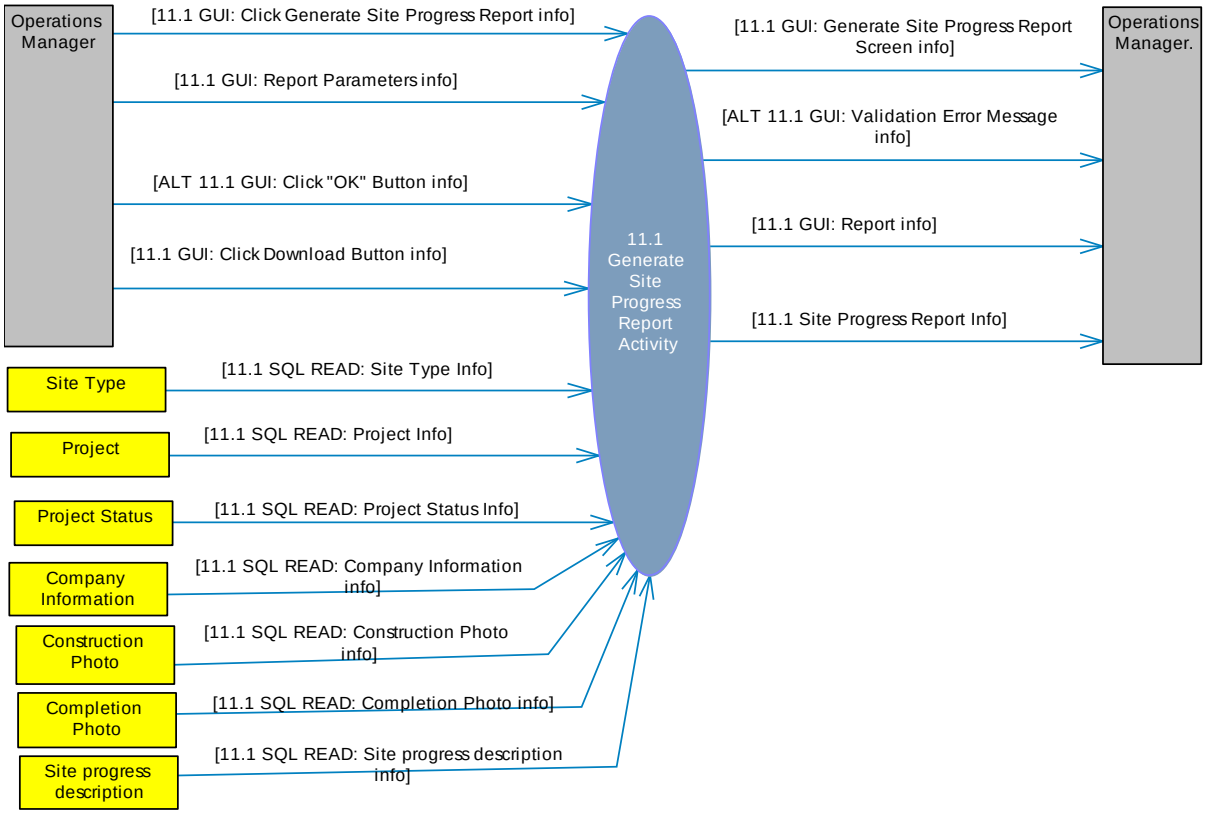


Figure 142: Middle Level diagram- 11.1 Generate Site Progress Report Activity

2.3.2.11.2 Generate Site Type Report

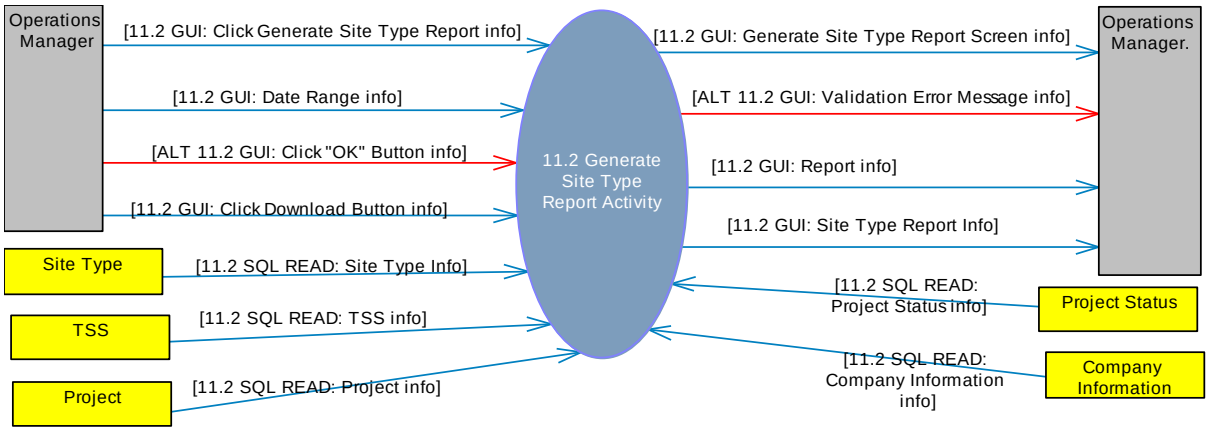


Figure 143: Middle Level diagram- 11.2 Generate Site Type Report Activity

2.3.2.11.3 Generate Team Site Completed Report

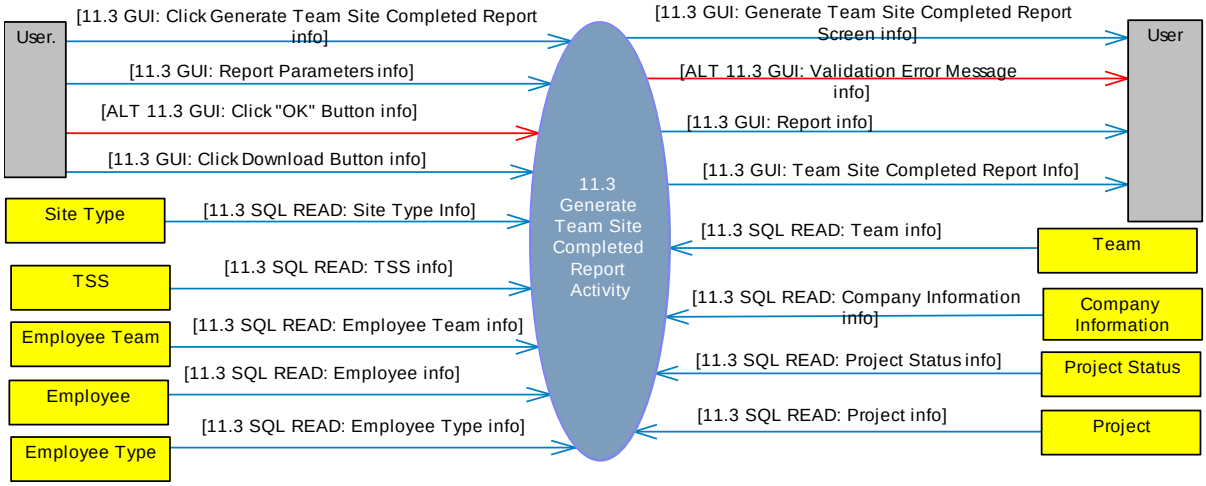


Figure 144: Middle Level diagram- 11.3 Generate Team Site Completed Report Activity

2.3.2.11.4 Generate Employee Document Expiry Report

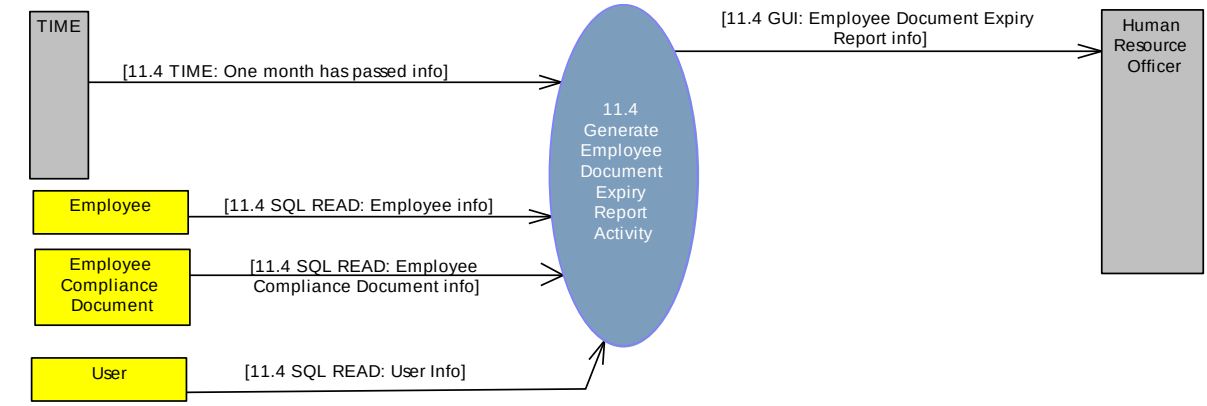


Figure 145: Middle Level diagram- 11.4 Generate Employee Document Expiry Report Activity

2.3.2.11.5 Generate Site Acceptance Document

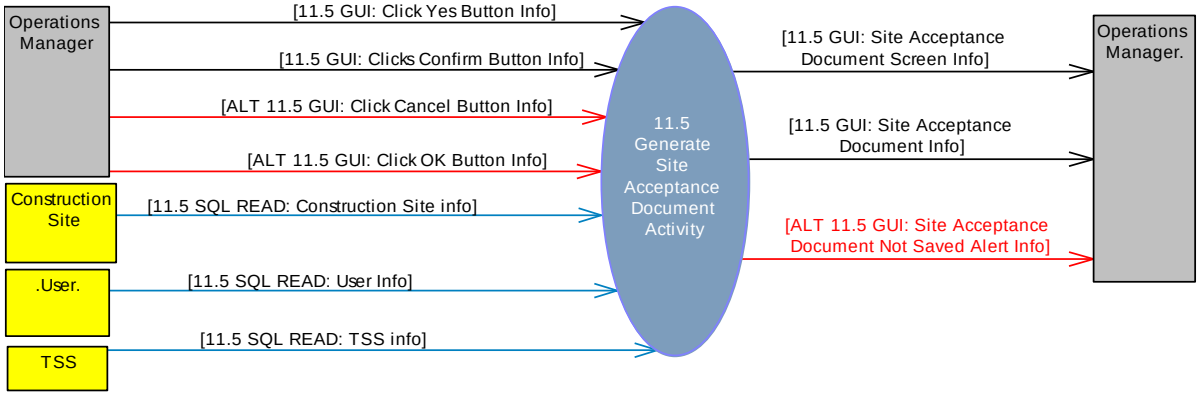


Figure 146: Middle Level diagram- 11.5 Generate Site Acceptance Document Activity

2.3.2.12. User Management Subsystem

2.3.2.12.1 Login

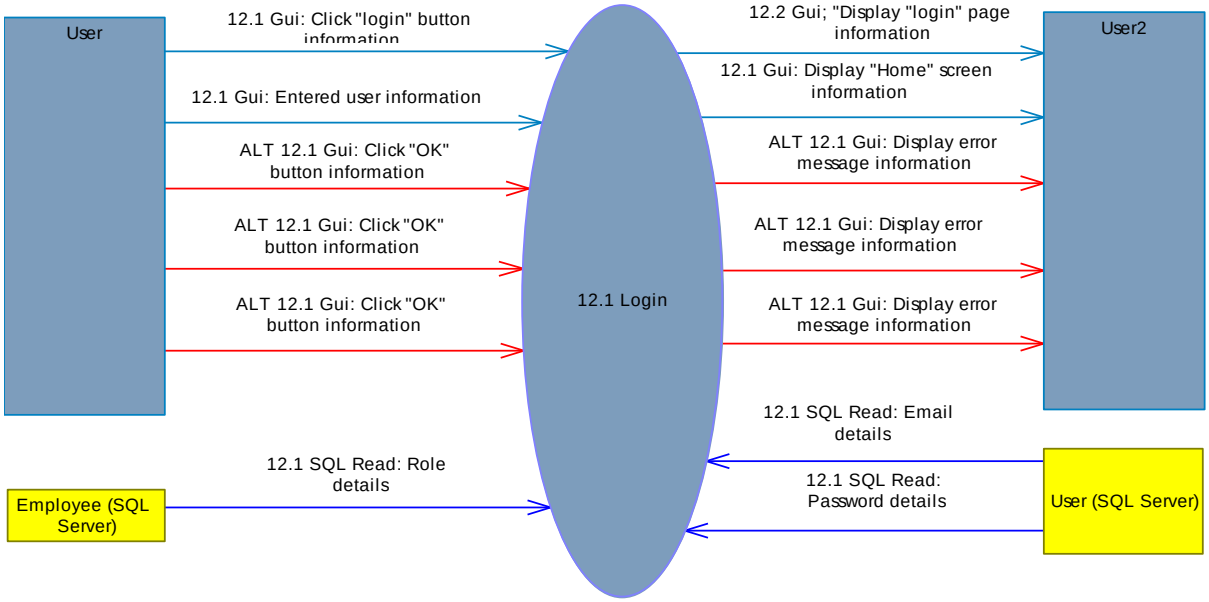


Figure 147: Middle level diagram 12.1 Login

2.3.2.12.2 Logout

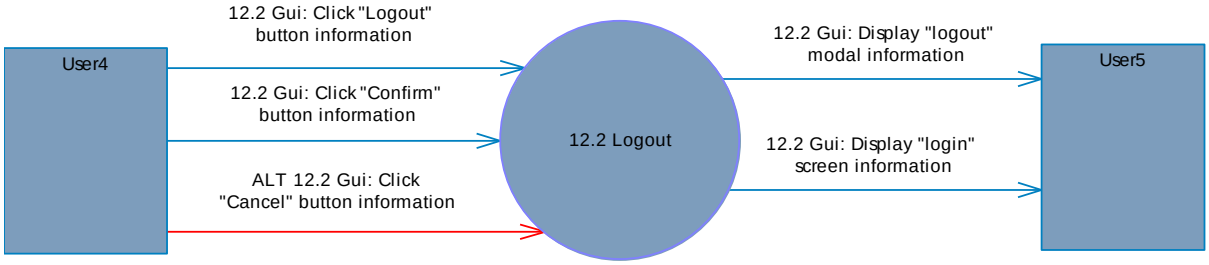


Figure 148: Middle level diagram 12.2 Logout

2.3.2.12.3 Reset Password

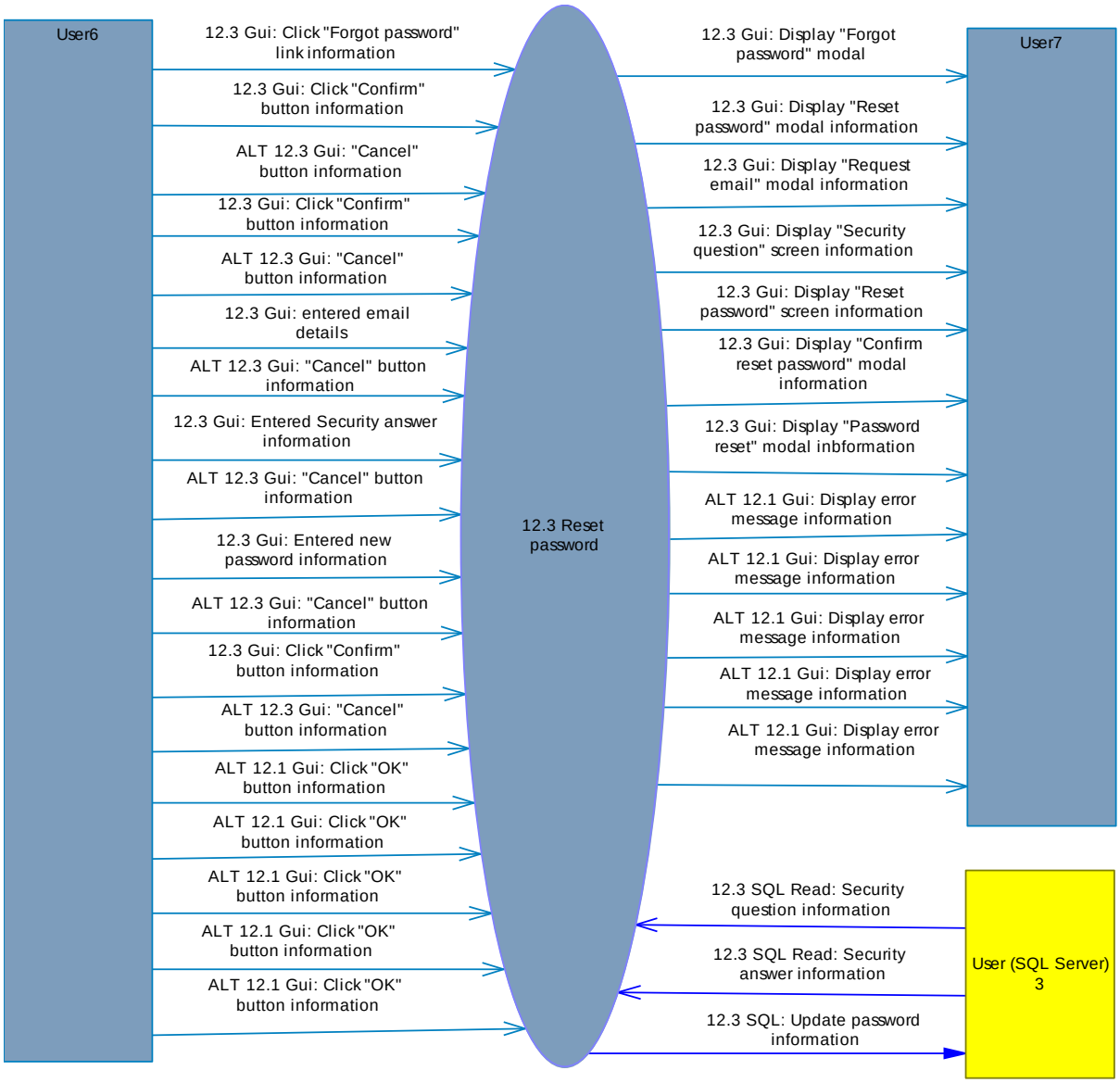


Figure 149: Middle level diagram 12.3 Reset password

2.3.2.13. Client Subsystem

2.3.2.13.1 Add Client

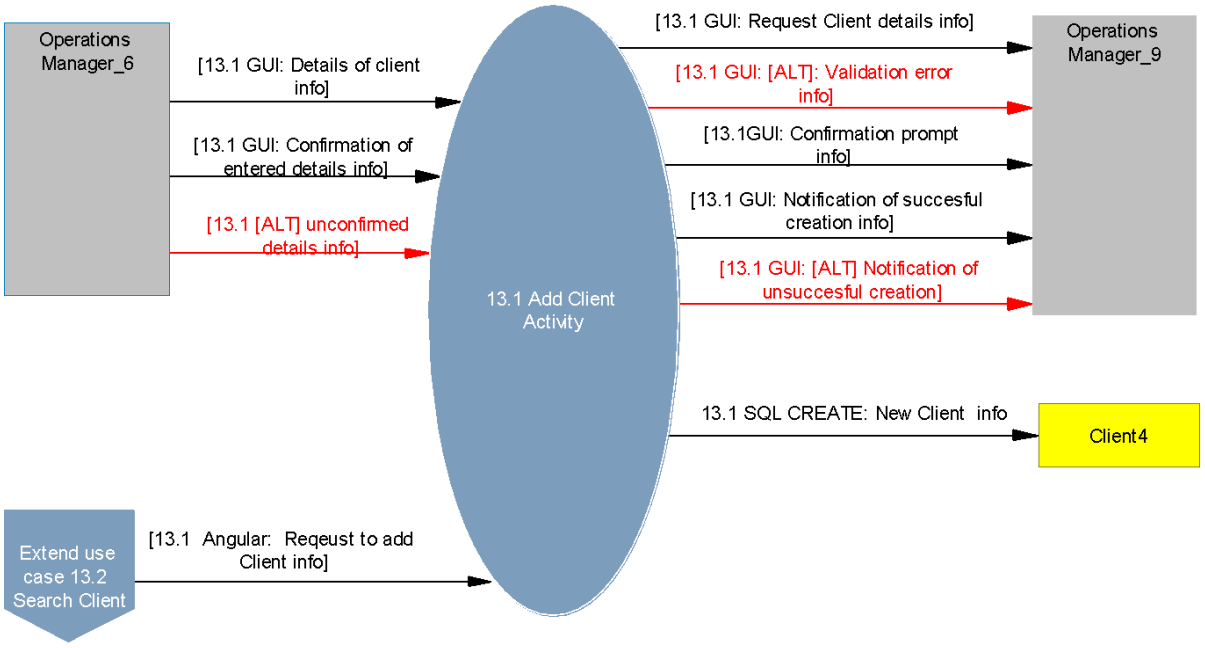


Figure 150: Middle level diagram 13.1 Add Client

2.3.2.13.2 Search Client

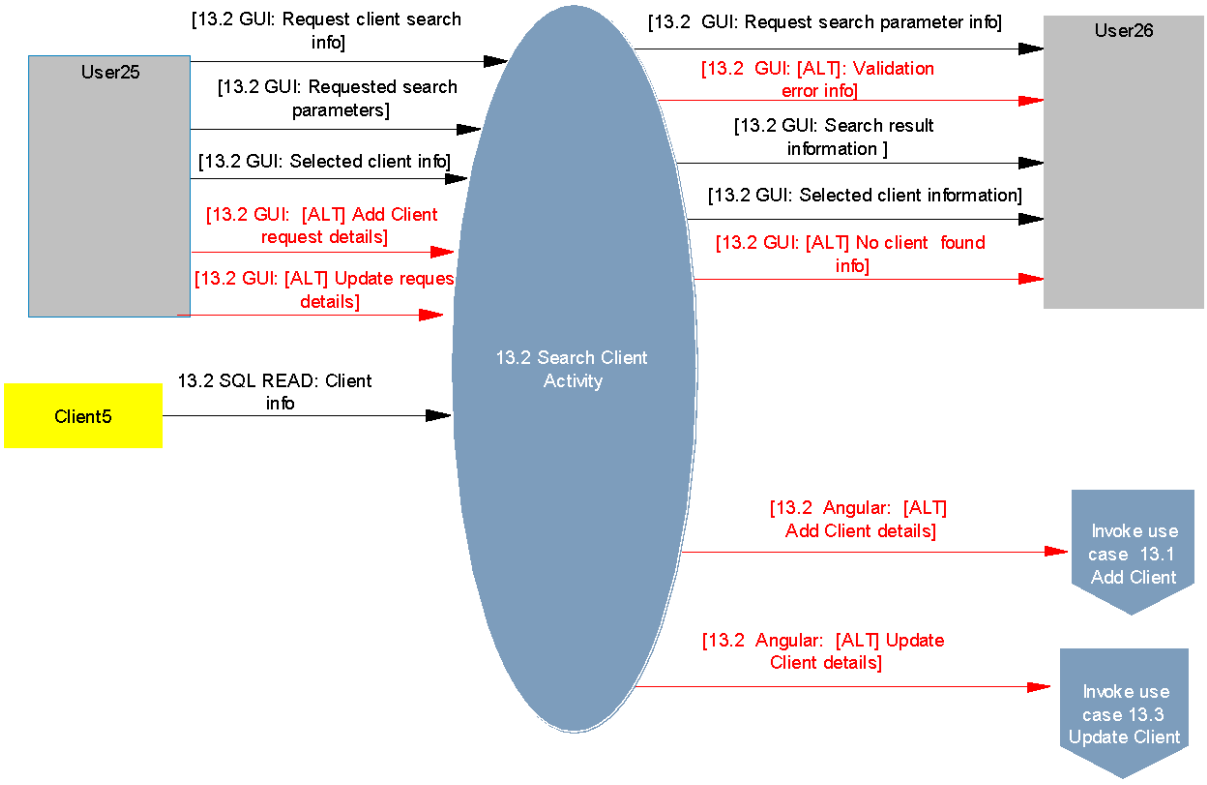


Figure 151: Middle level diagram 13.2 Search Client

2.3.2.13.3 Edit Client

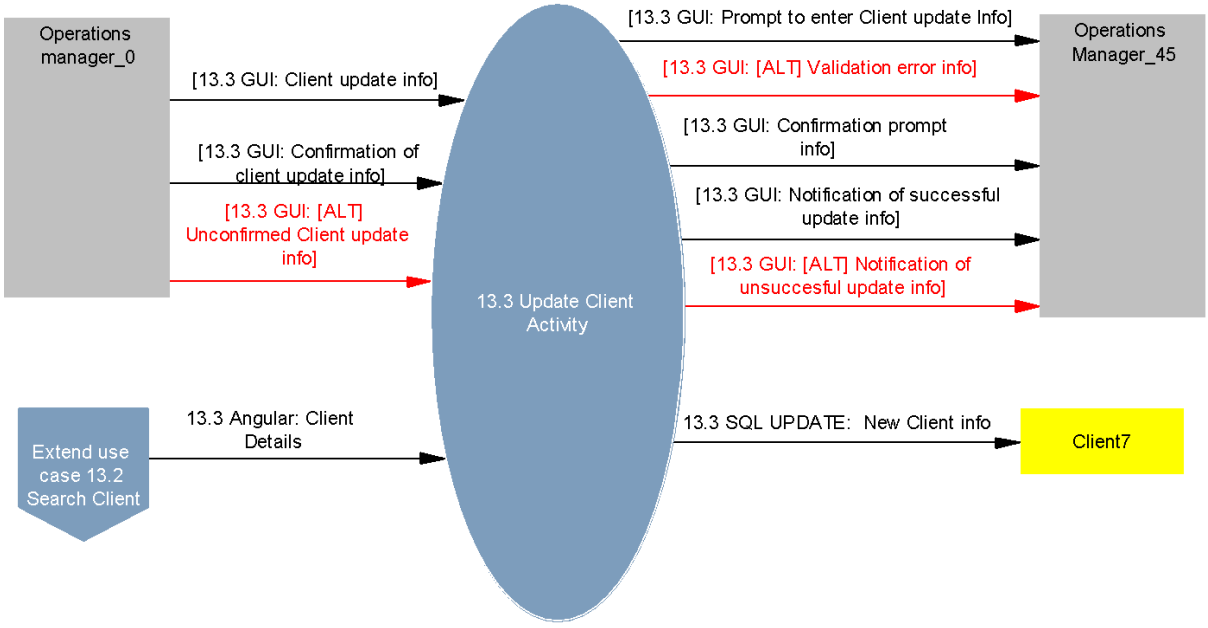


Figure 152: Middle level diagram 13.3 Edit Client

2.3.3 Primitive Level Date flow diagrams

2.3.3.1. Establishment Subsystem

2.3.3.1.1 Create TSS

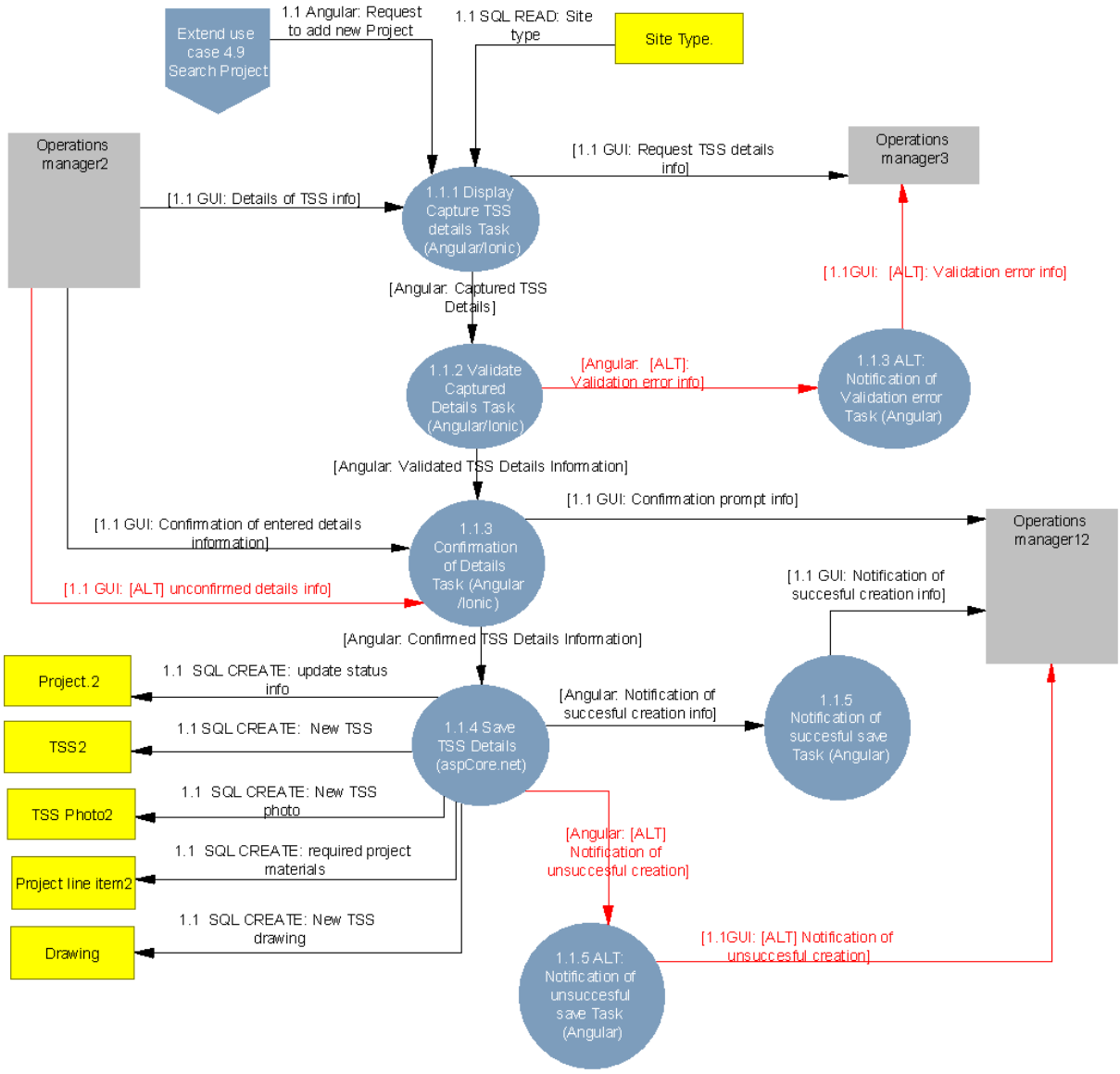


Figure 153: Primitive level data diagram1.1 Create TSS

2.3.3.1.2 Edit TSS

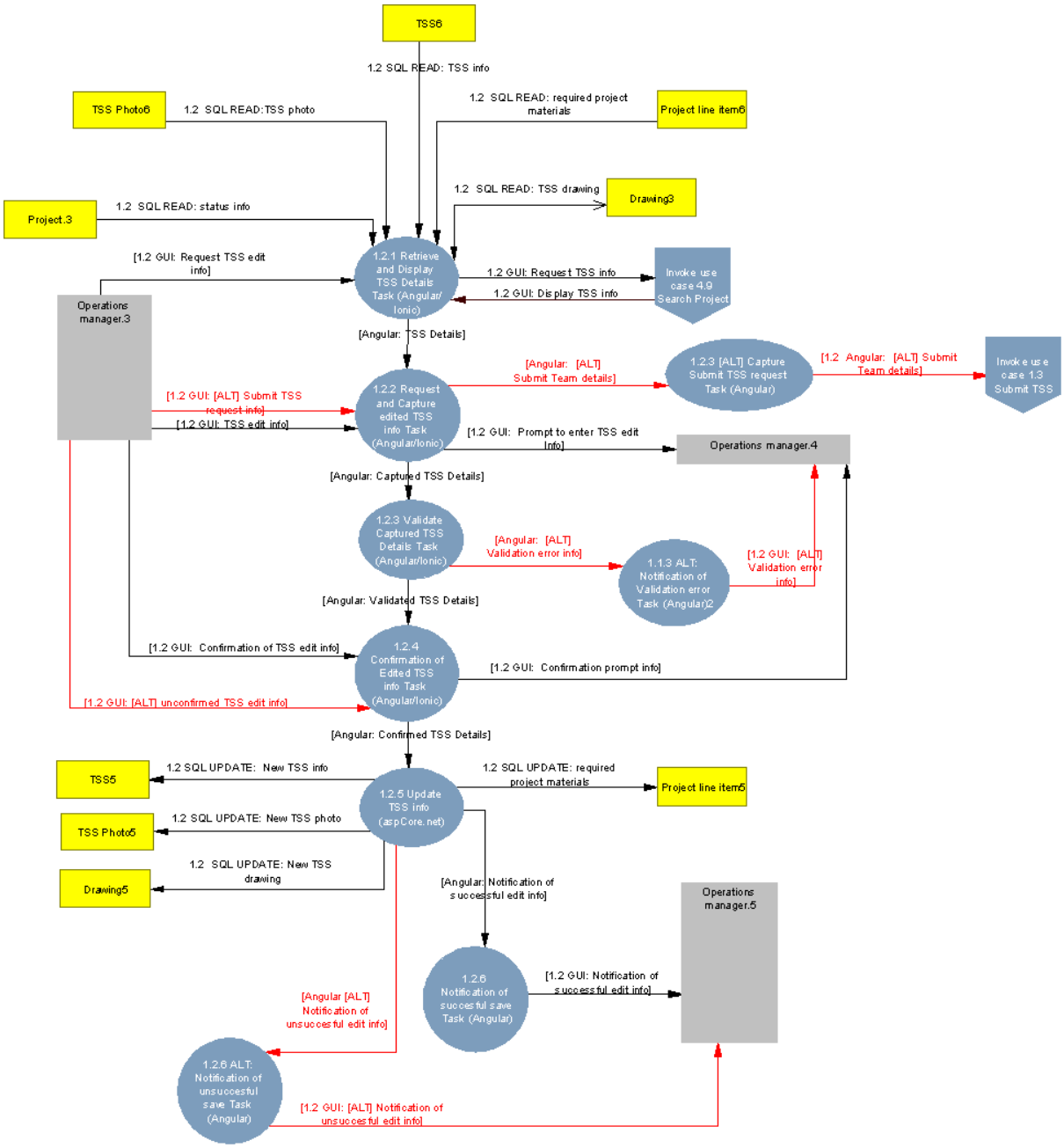


Figure 154: Primitive level data diagram1.2 Edit TSS

2.3.3.1.3 Submit TSS Report

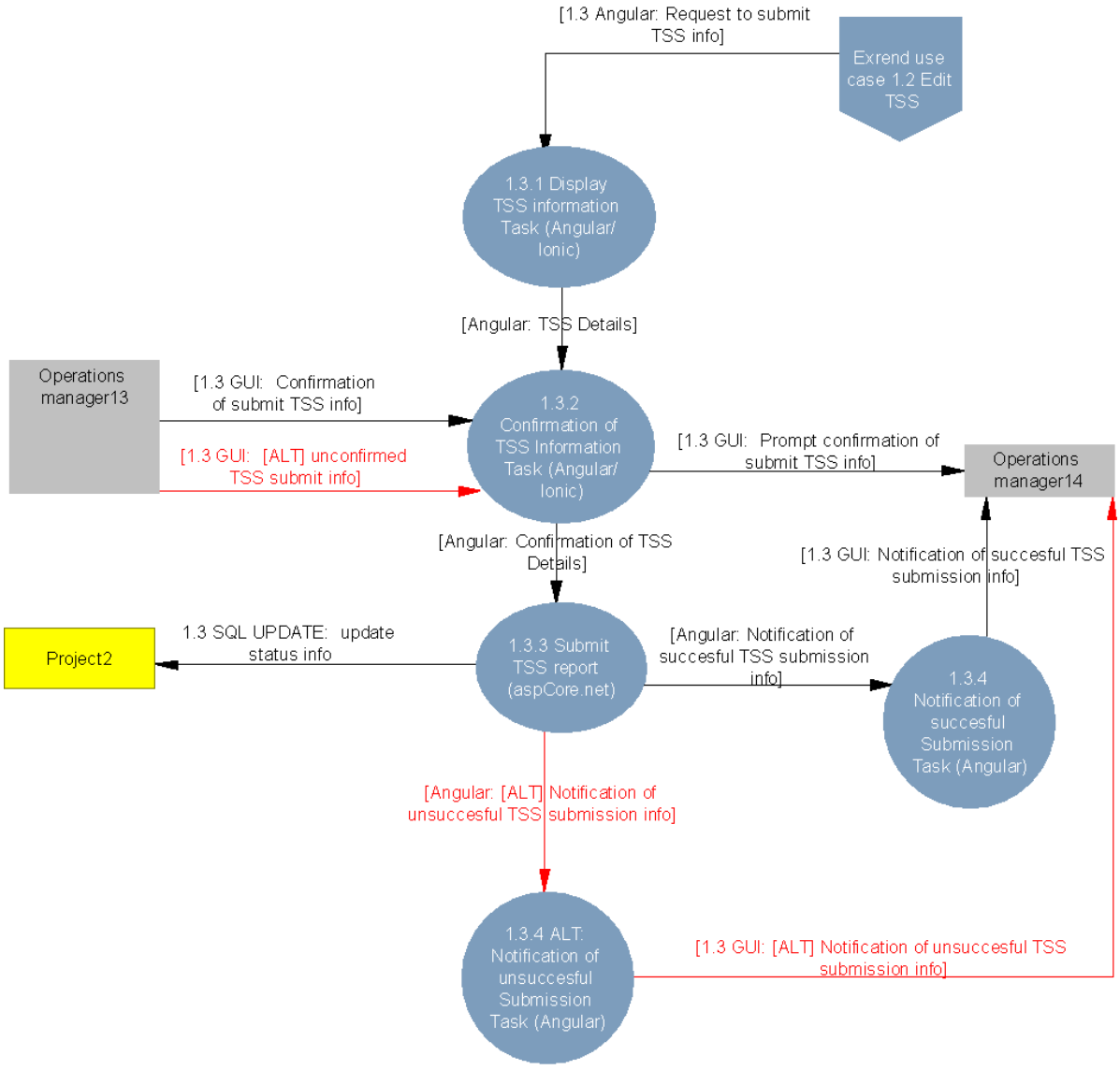


Figure 155: Primitive level data diagram1.3 Submit TSS Report

2.3.3.2. Quotation Subsystem

2.3.3.2.1 Create Quote

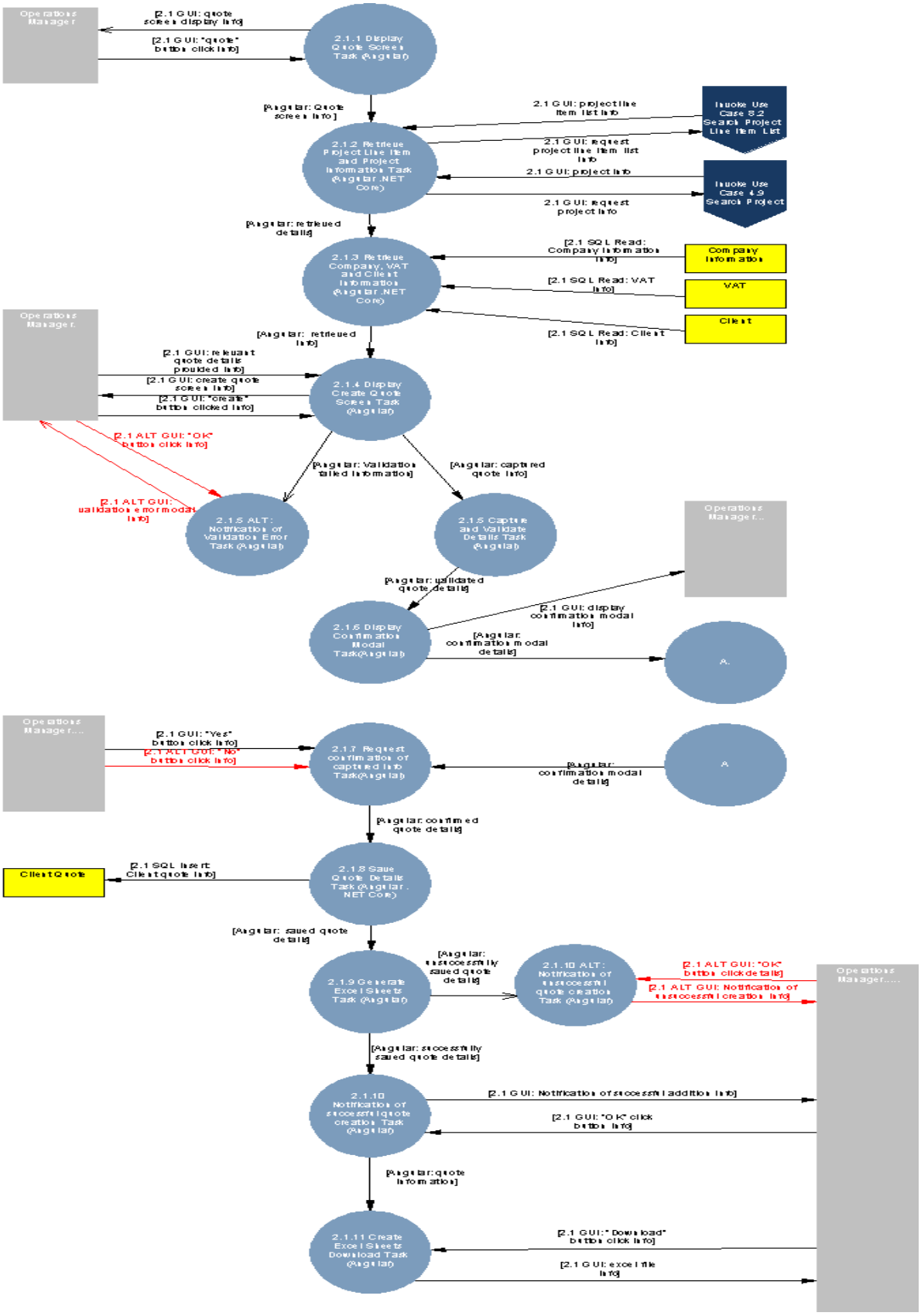


Figure 156: Primitive level data diagram 2.1 Create Quote

2.3.3.2.2 Search Quote

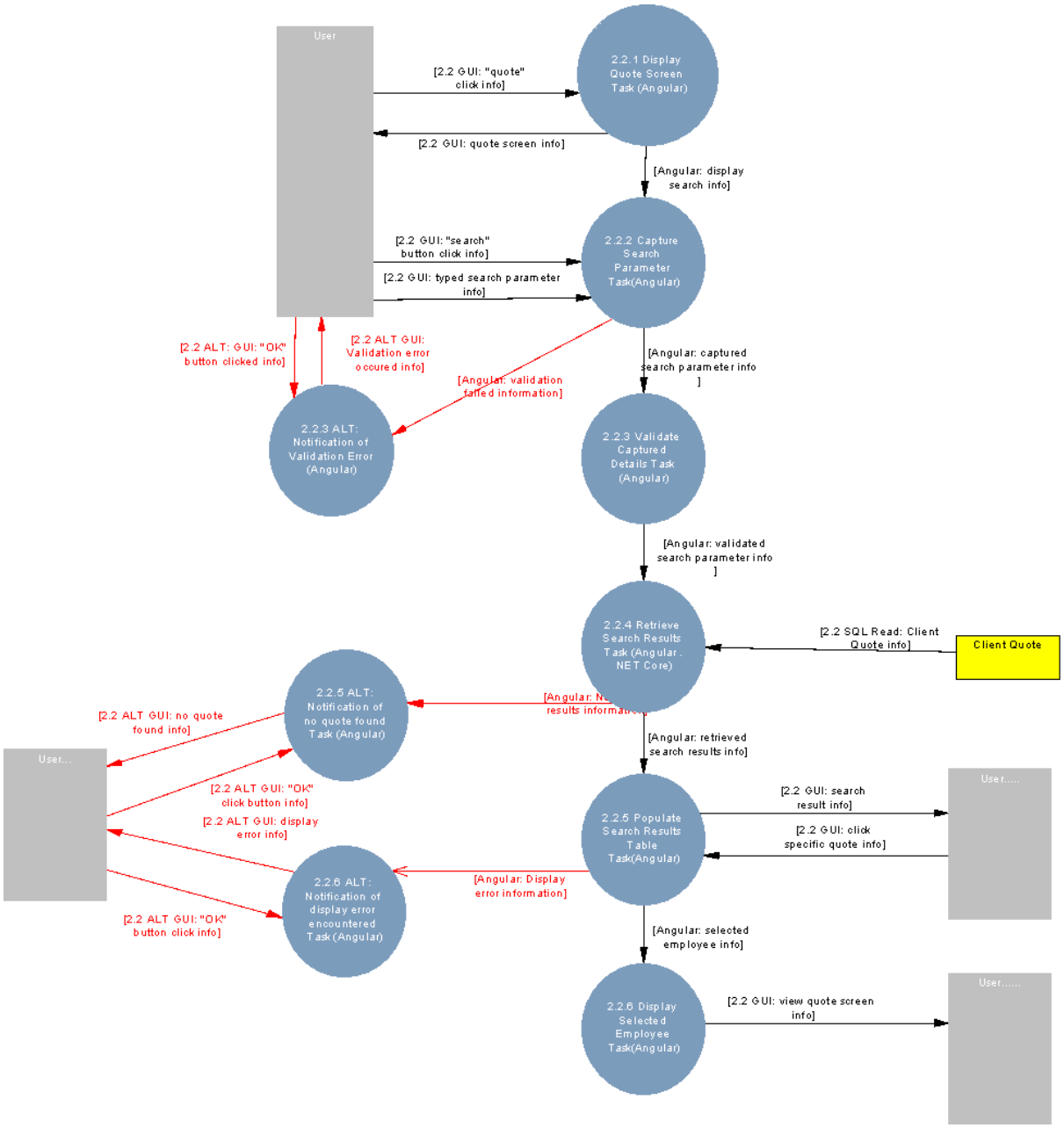


Figure 157: Primitive level data diagram2.2 Search Quote

2.3.3.2.3 Capture Purchase Order

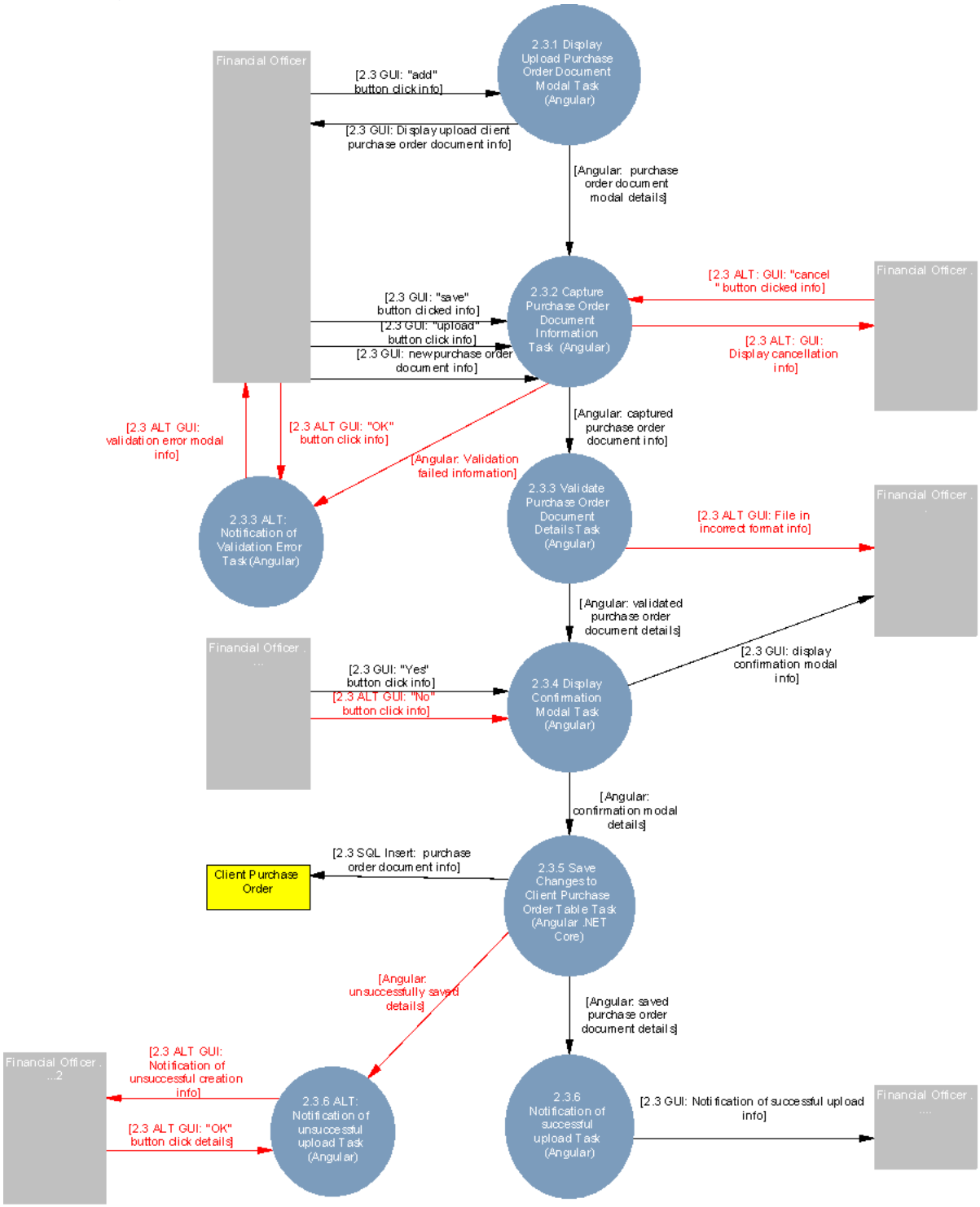


Figure 158: Primitive level data diagram2.3 Capture Purchase Order

2.3.3.3. Administration Subsystem

2.3.3.3.1 Create Health and Safety file

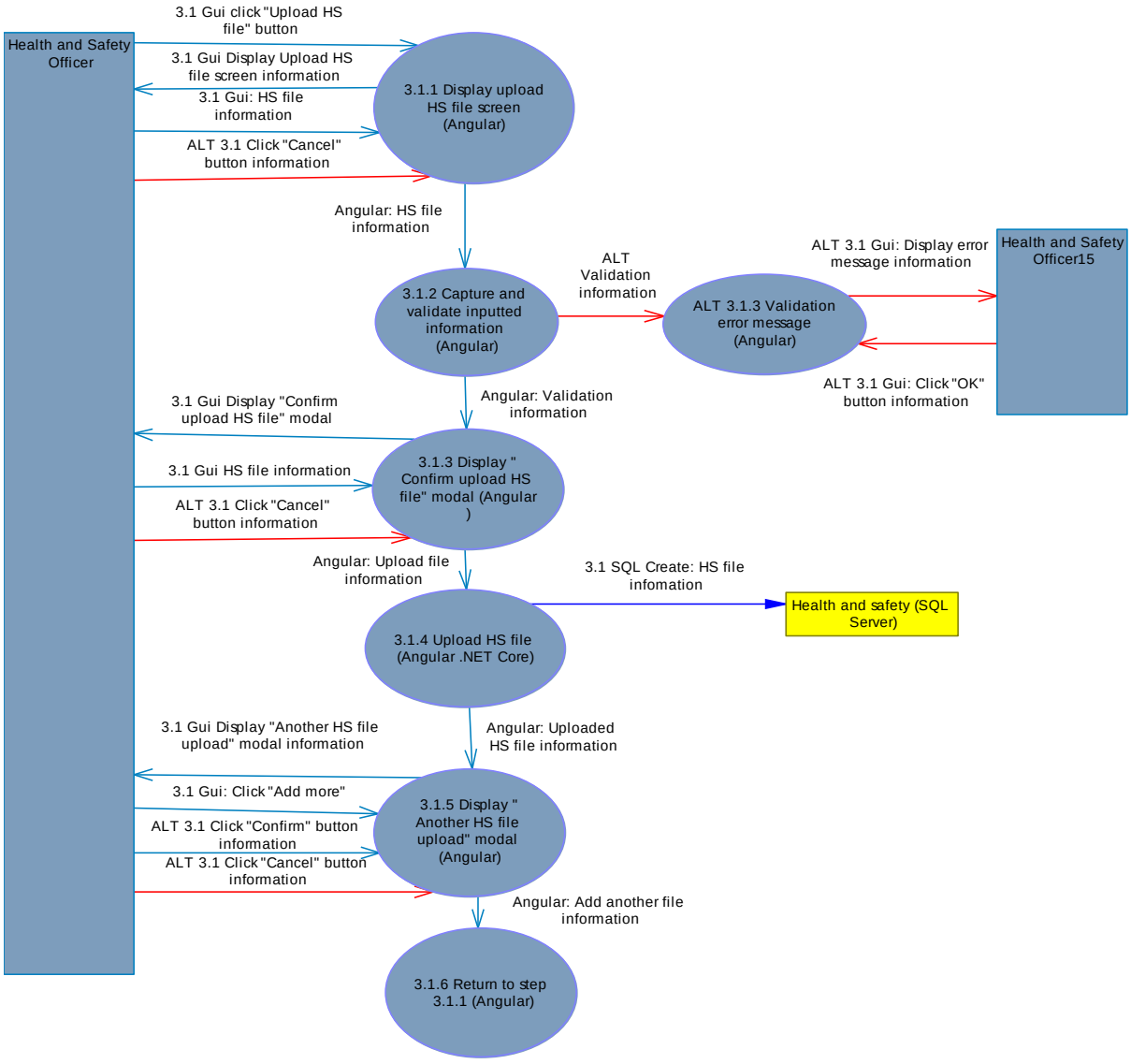


Figure 159: Primitive level data diagram3.1 Create Health and Safety file

2.3.3.3.2 Search Health and Safety file

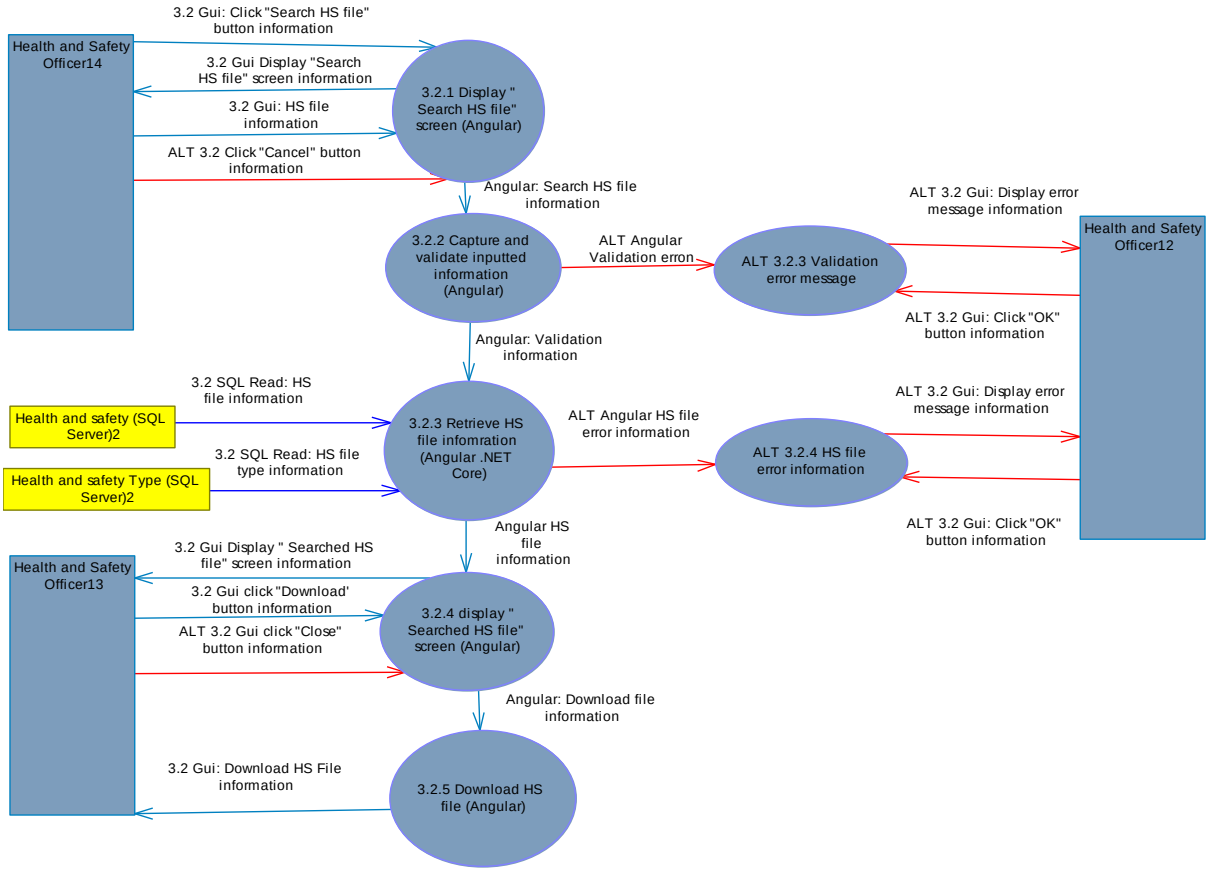


Figure 160: Primitive level data diagram3.2 Search Health and Safety file

2.3.3.3.3 Create Health and Safety Type

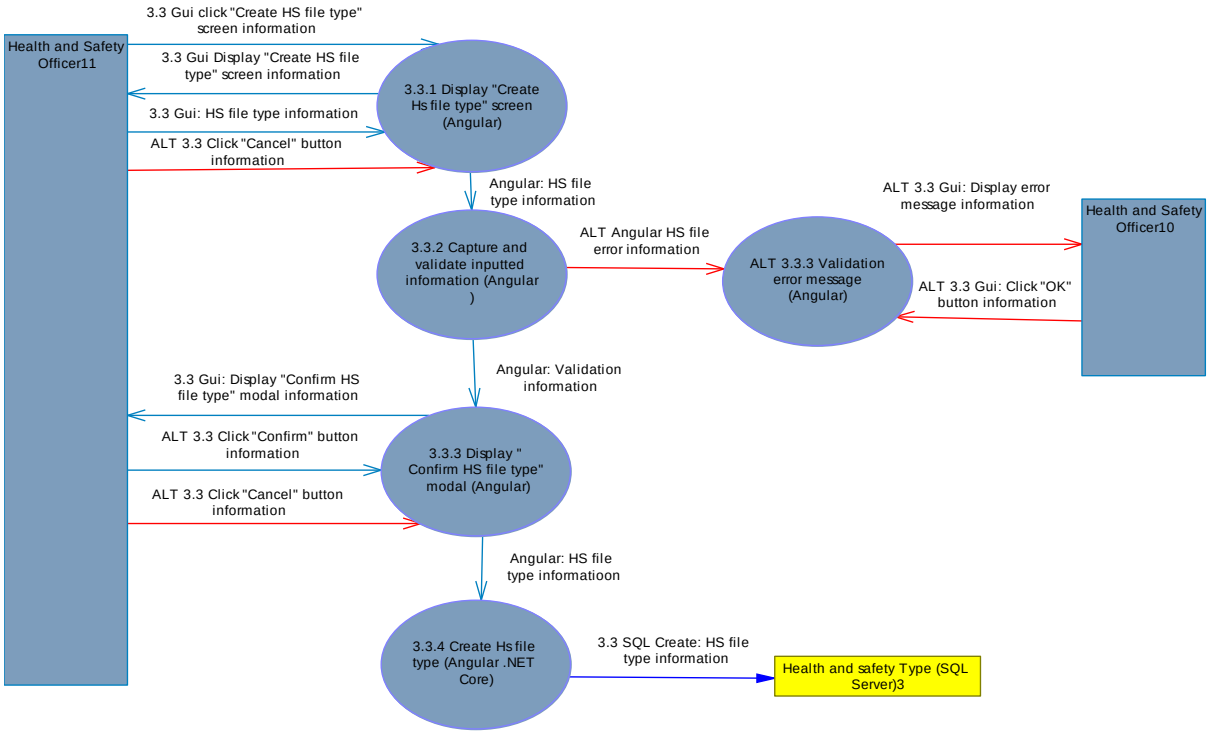


Figure 161: Primitive level data diagram3.3 Create Health and Safety Type

2.3.3.3.4 Search Health and Safety Type

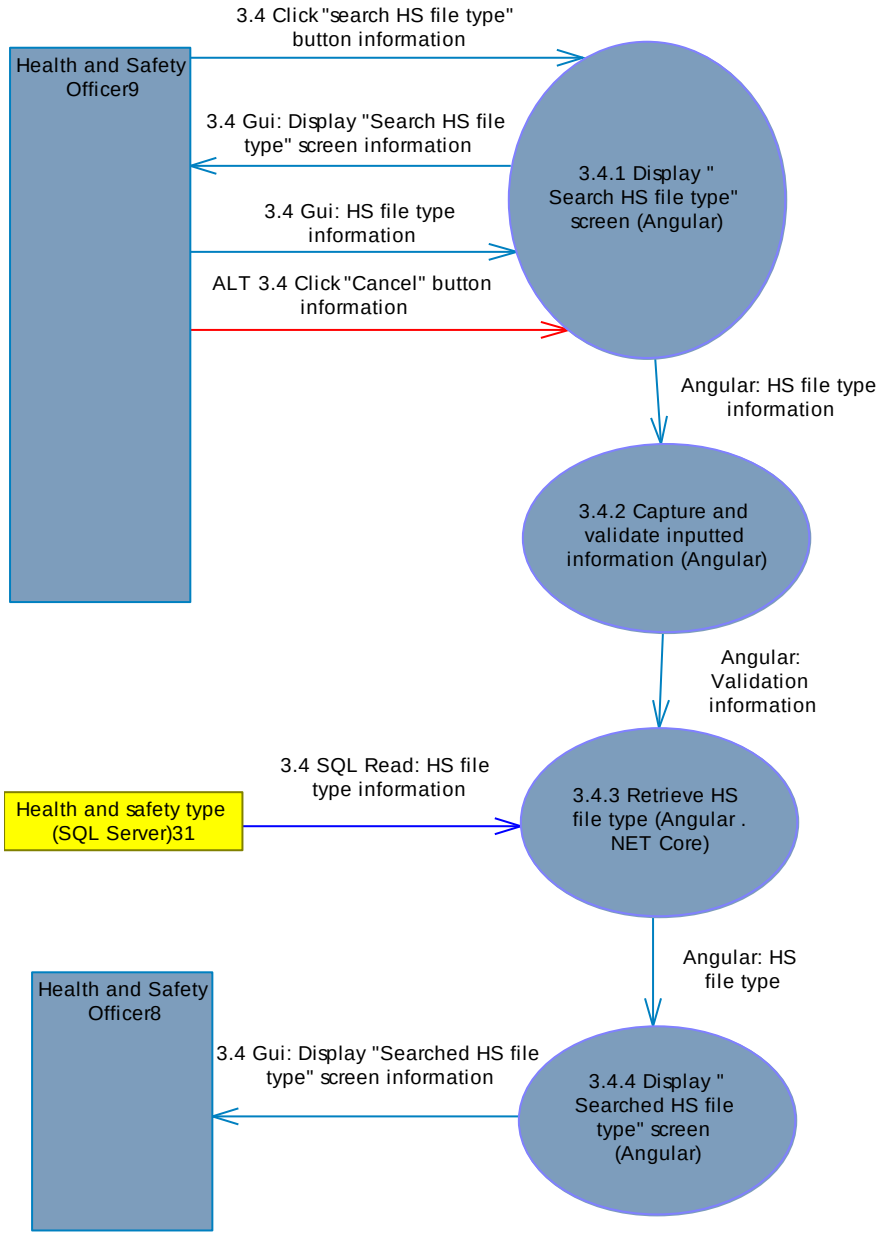


Figure 162: Primitive level data diagram 3.4 Search Health and Safety Type

2.3.3.3.5 Create Drawing Type

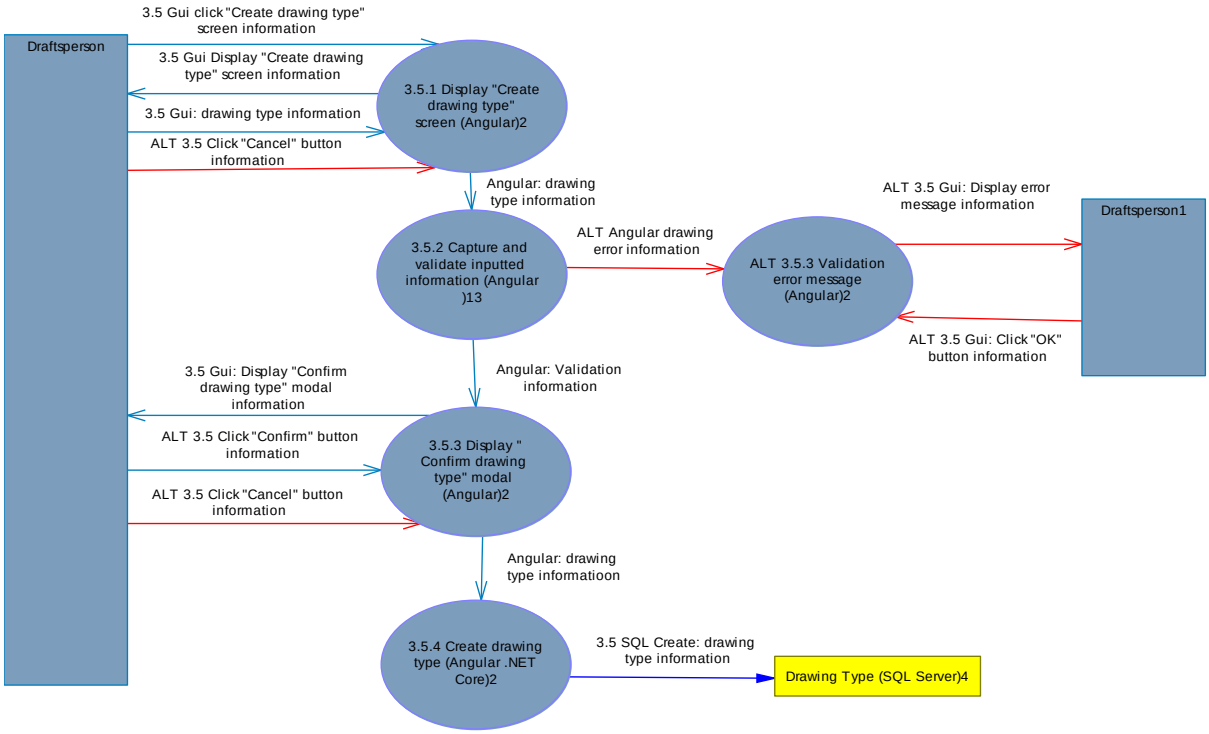


Figure 163: Primitive level data diagram3.5 Create Drawing Type

2.3.3.3.6 Search Drawing Type

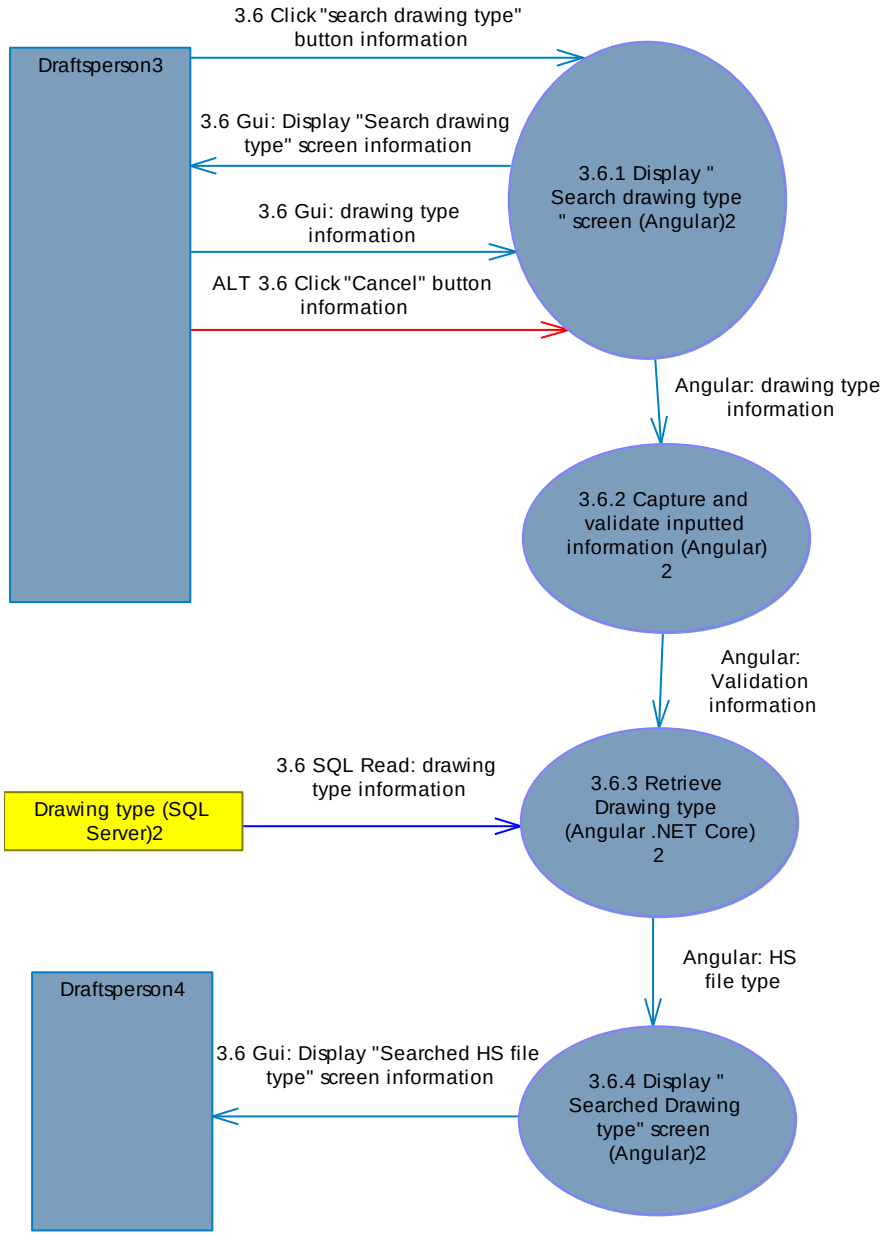


Figure 164: Primitive level data diagram3.6 Search Drawing Type

2.3.3.3.7 Upload Formal Drawing

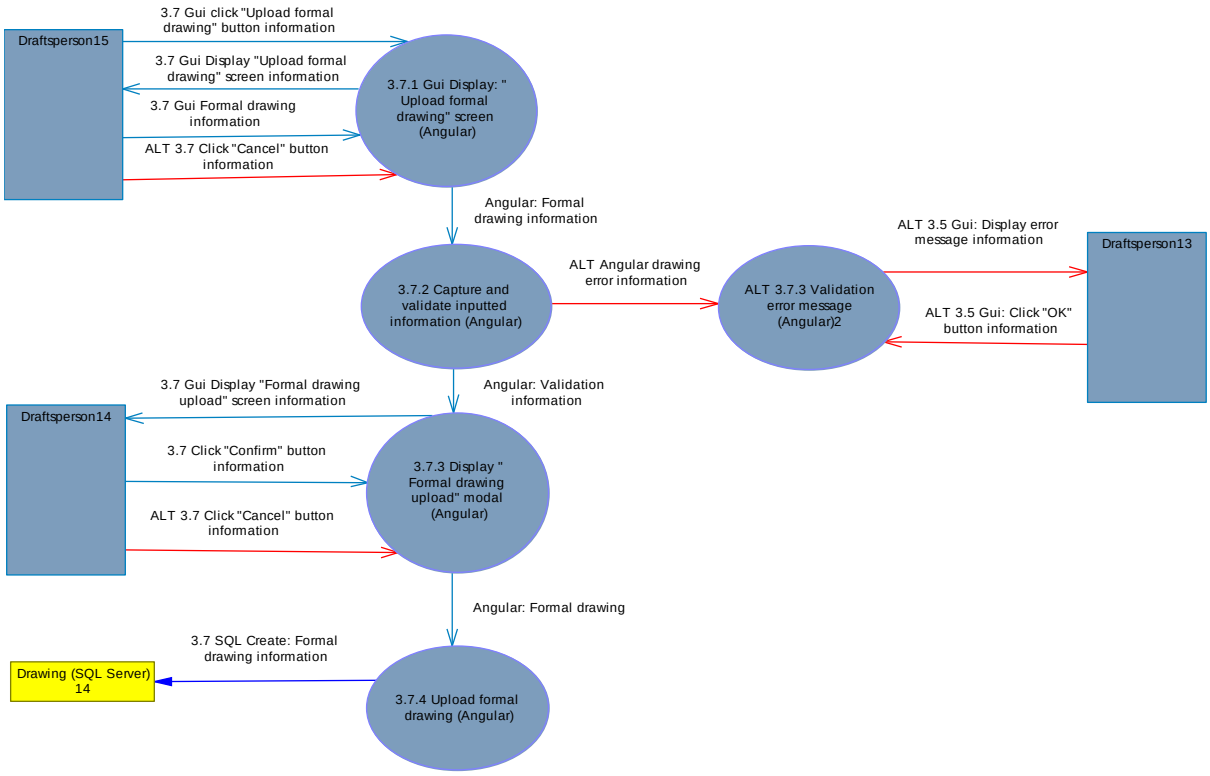


Figure 165: Primitive level data diagram3.7 Upload Formal Drawing

2.3.3.3.8 Upload Second cut Drawing

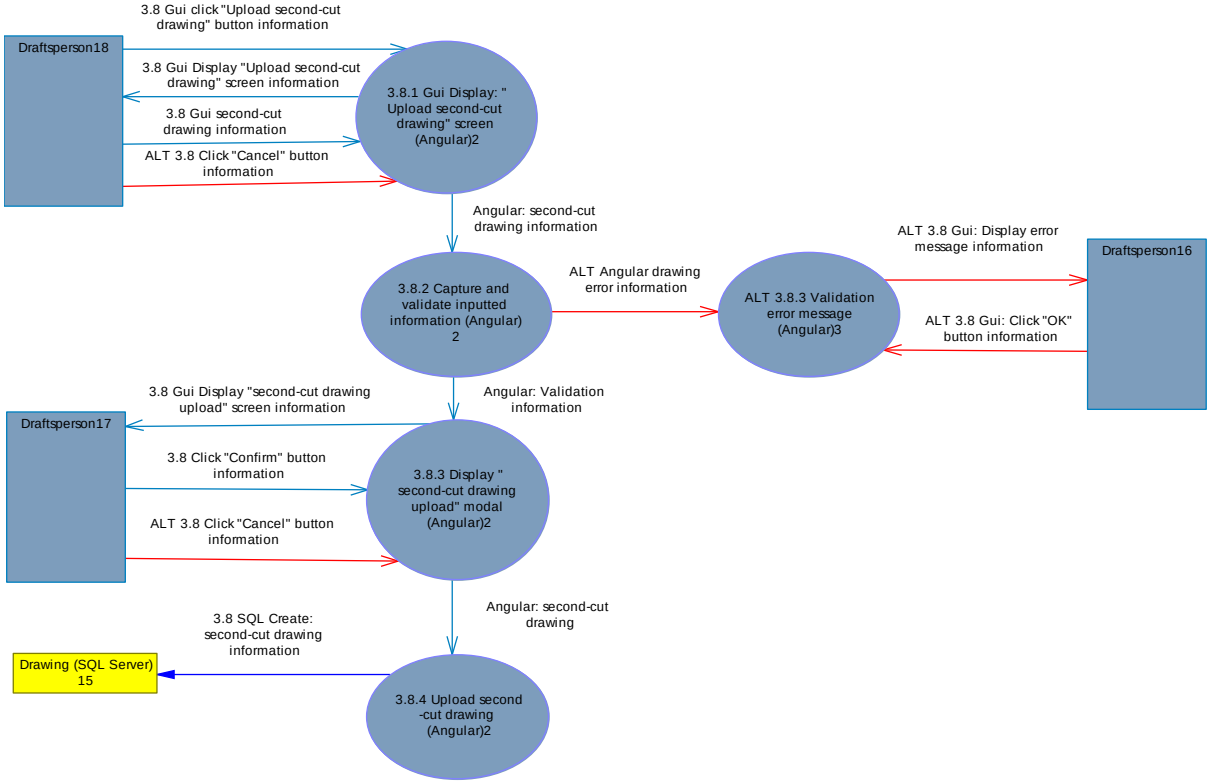


Figure 166: Primitive level data diagram3.8 Upload Second cut Drawing

2.3.3.3.9 Search As-built document

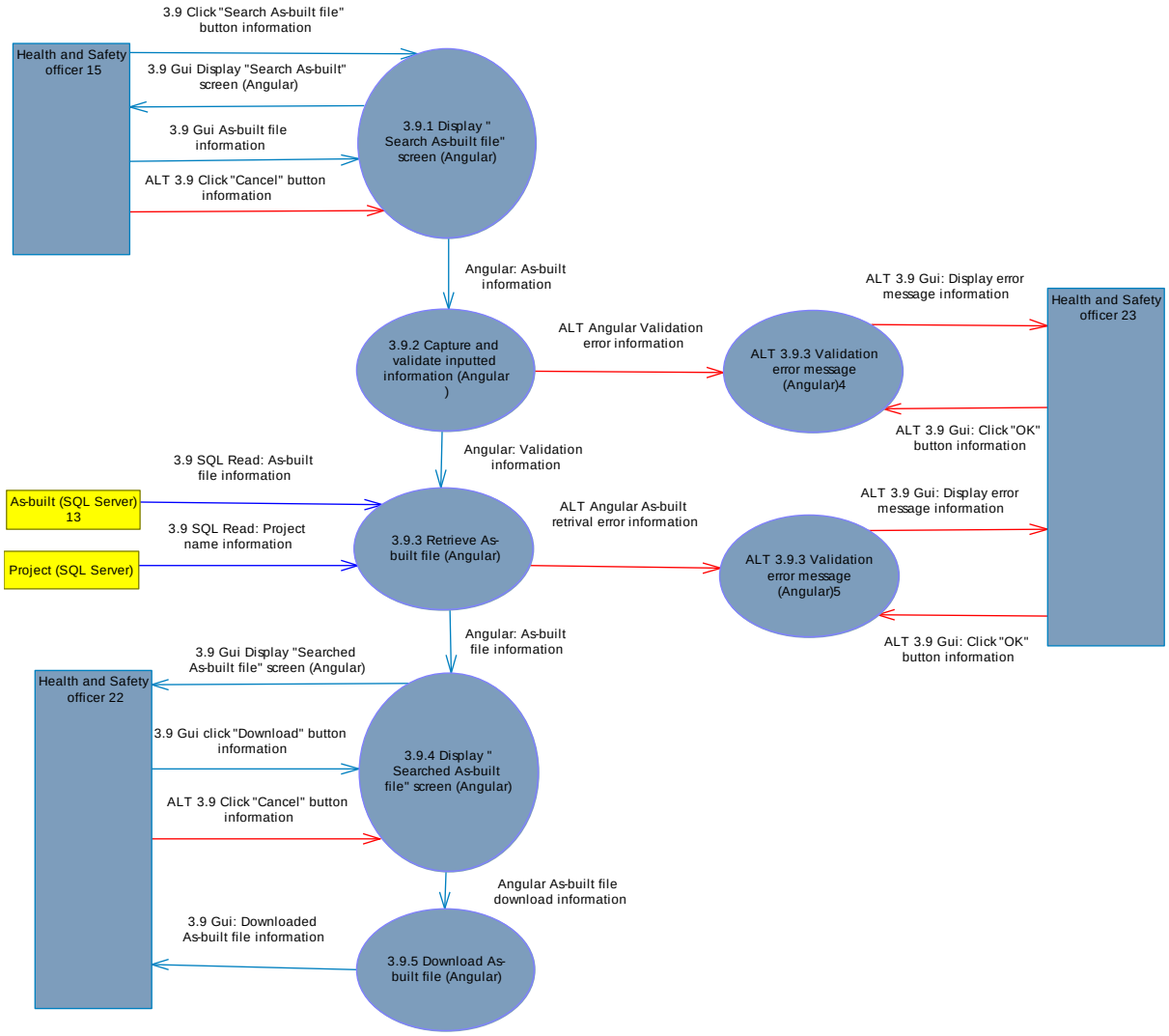


Figure 167: Primitive level data diagram3.9 Search As-built document

2.3.3.3.10 Generate As-Built Document

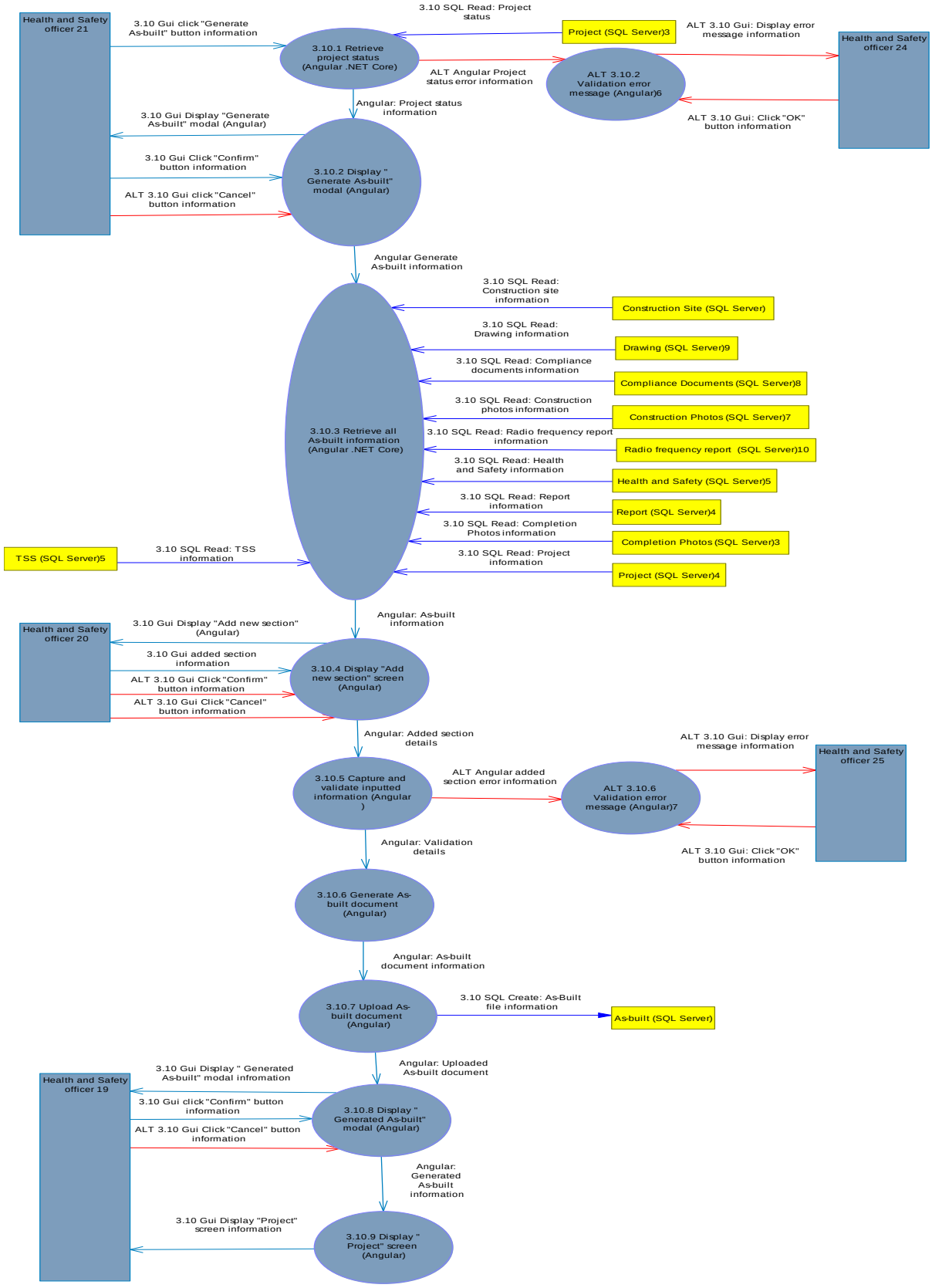


Figure 168: Primitive level data diagram3.10 Generate As-Built Document

2.3.3.4. Construction Subsystem

2.3.3.4.1 Capture Site-Go-Ahead

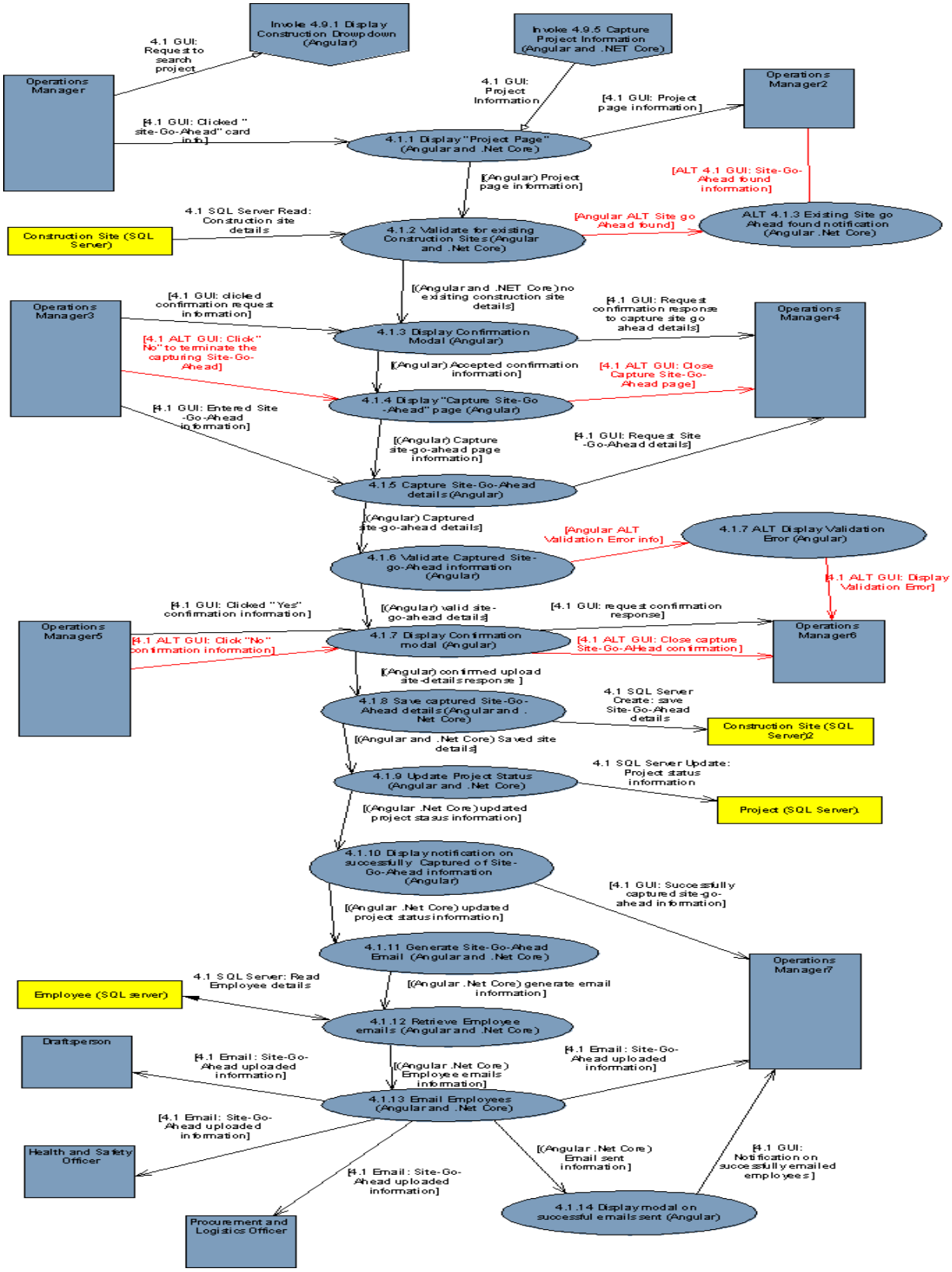


Figure 169: Primitive level data diagram4.1 Capture Site-Go-Ahead

2.3.3.4.2 Search construction

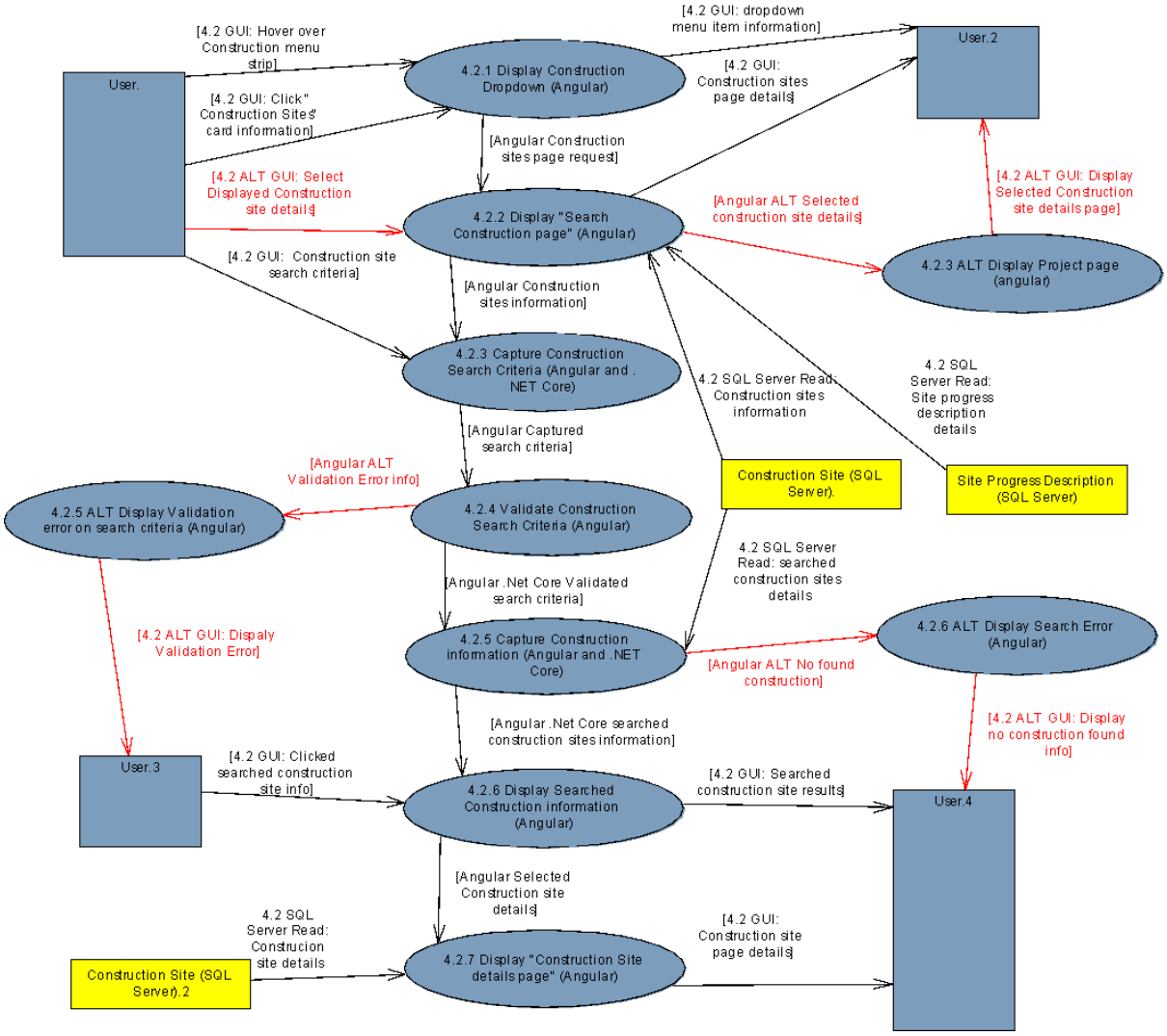


Figure 170: Primitive level data diagram4.2 Search construction

2.3.3.4.3 Edit construction

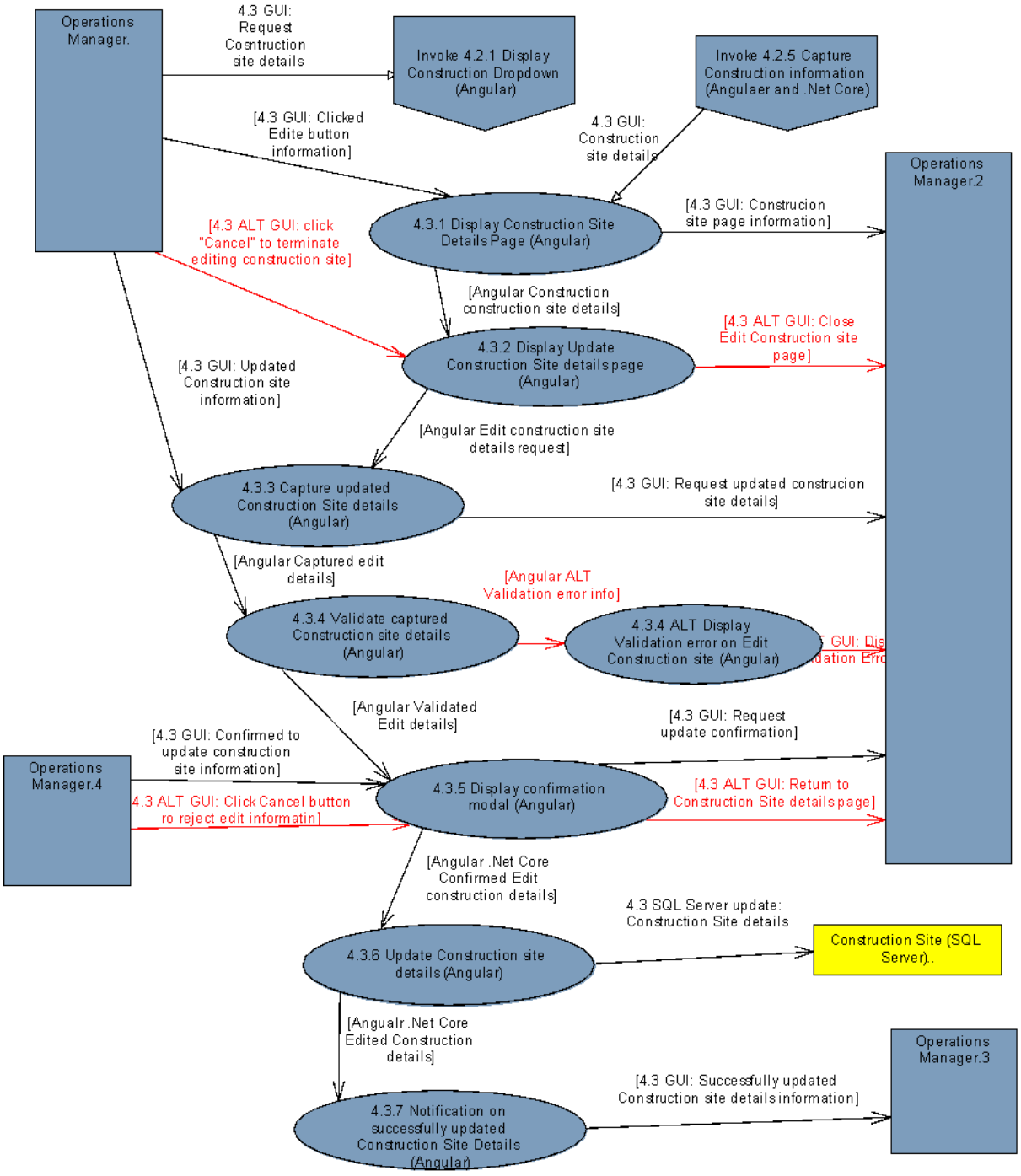


Figure 171: Primitive level data diagram4.3 Edit construction

2.3.3.4.4 Allocate Team Site

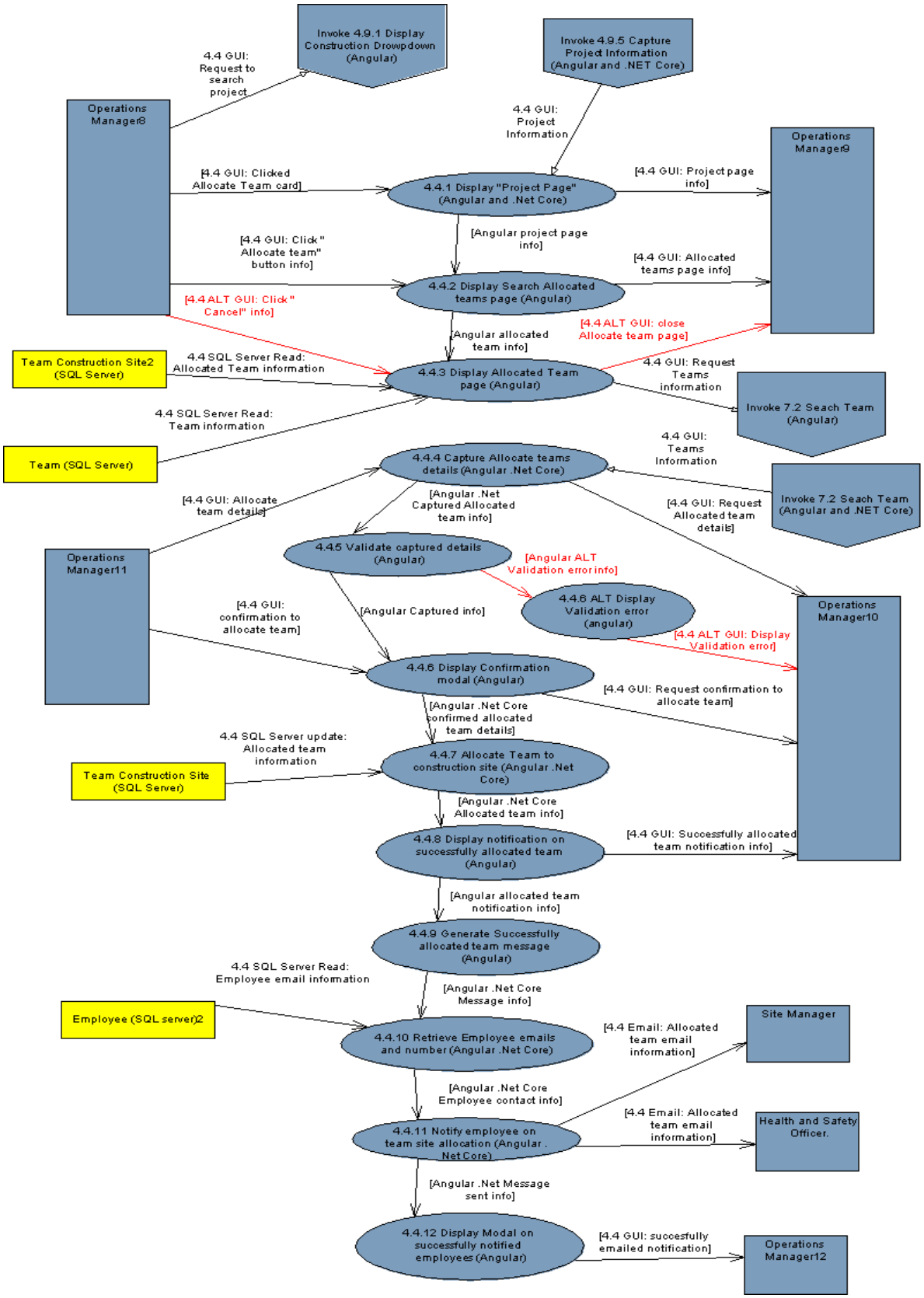


Figure 172: Primitive level data diagram4.4 Allocate Team Site

2.3.3.4.5 Search Allocated Team

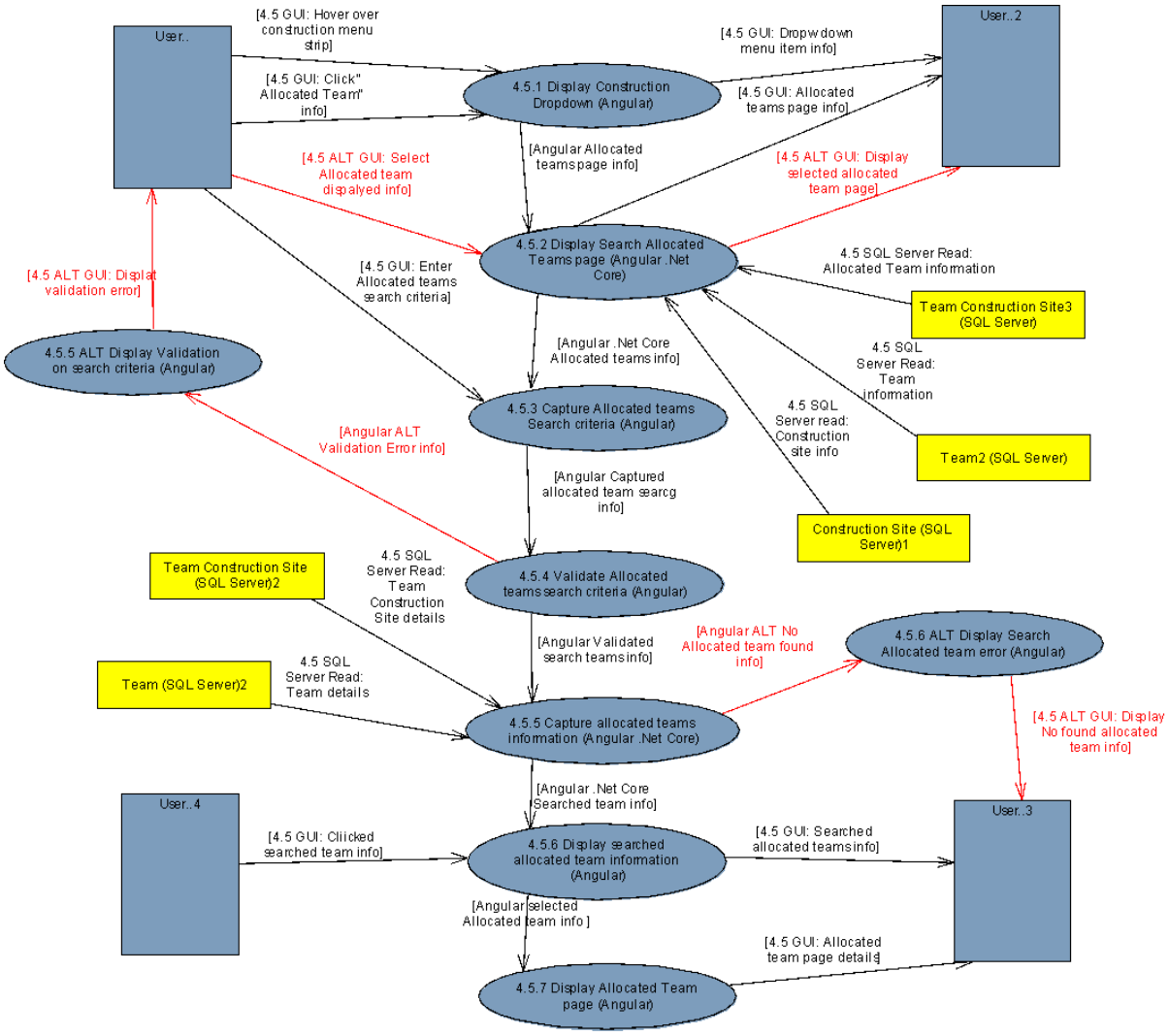


Figure 173: Primitive level data diagram4.5 Search Allocated Team

2.3.3.4.6 Update Allocated Team

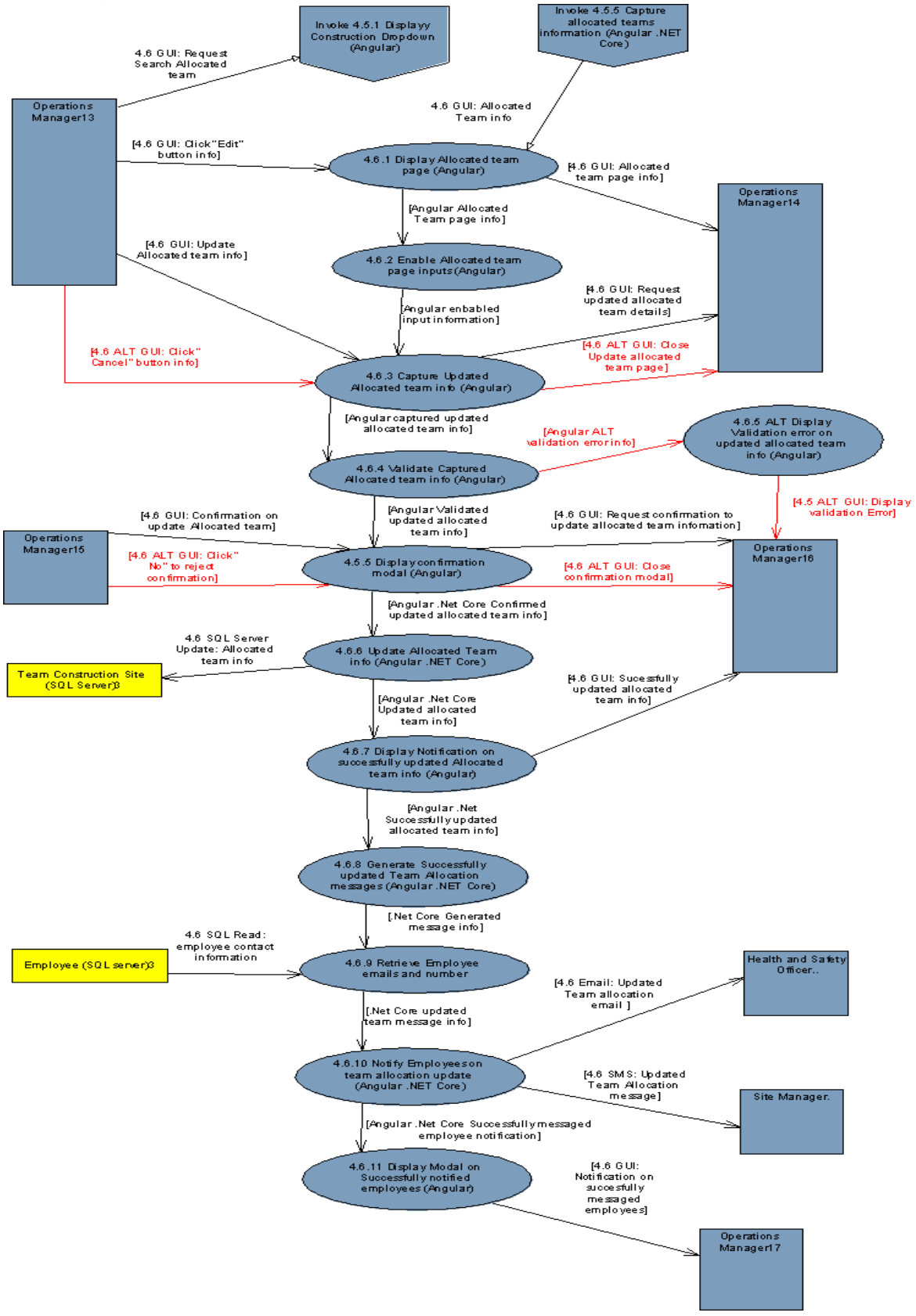


Figure 174: Primitive level data diagram4.6 Update Allocated Team

2.3.3.4.7 Upload Compliance Document

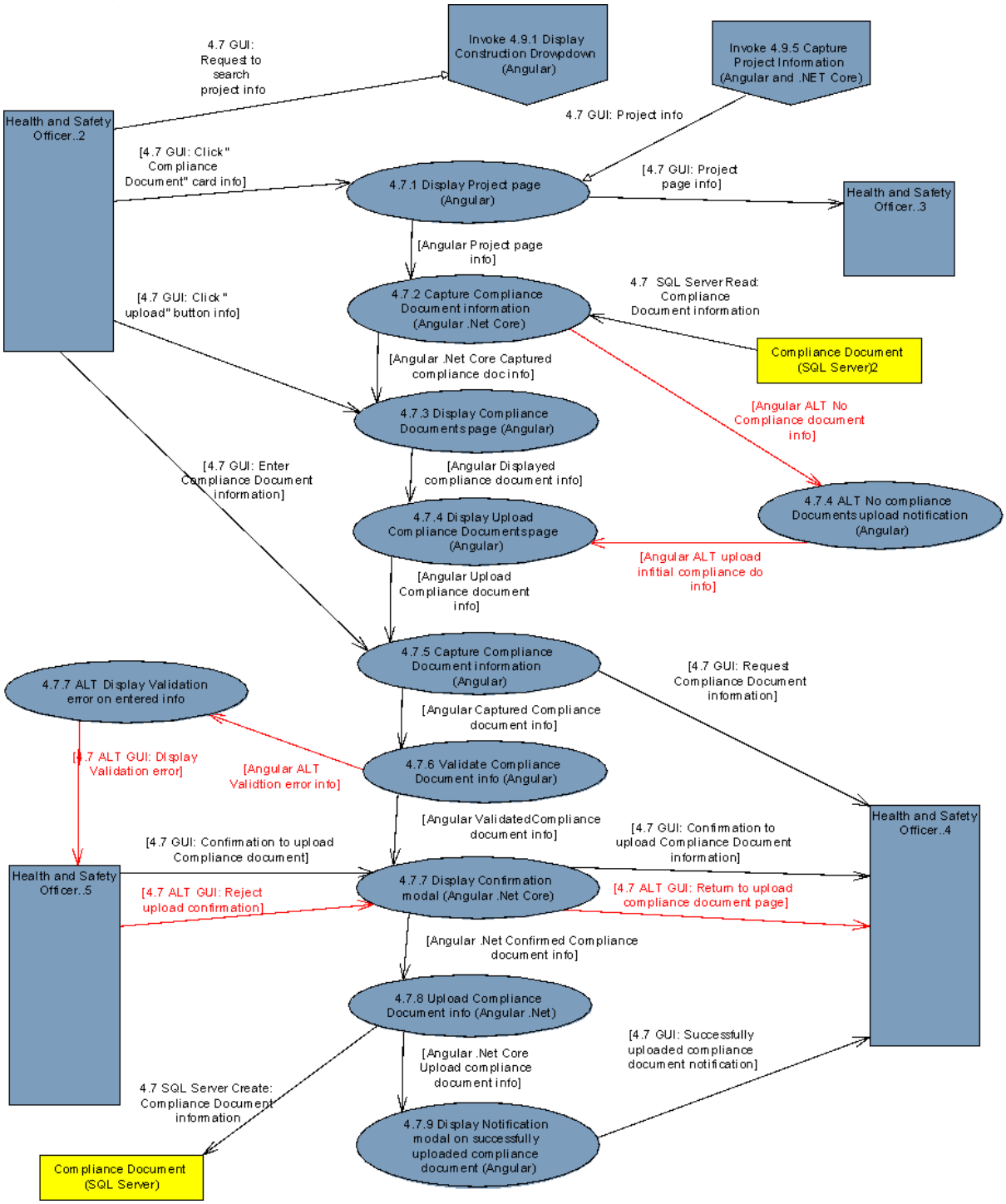


Figure 175: Primitive level data diagram4.7 Upload Compliance Document

2.3.3.4.8 Search Compliance Document

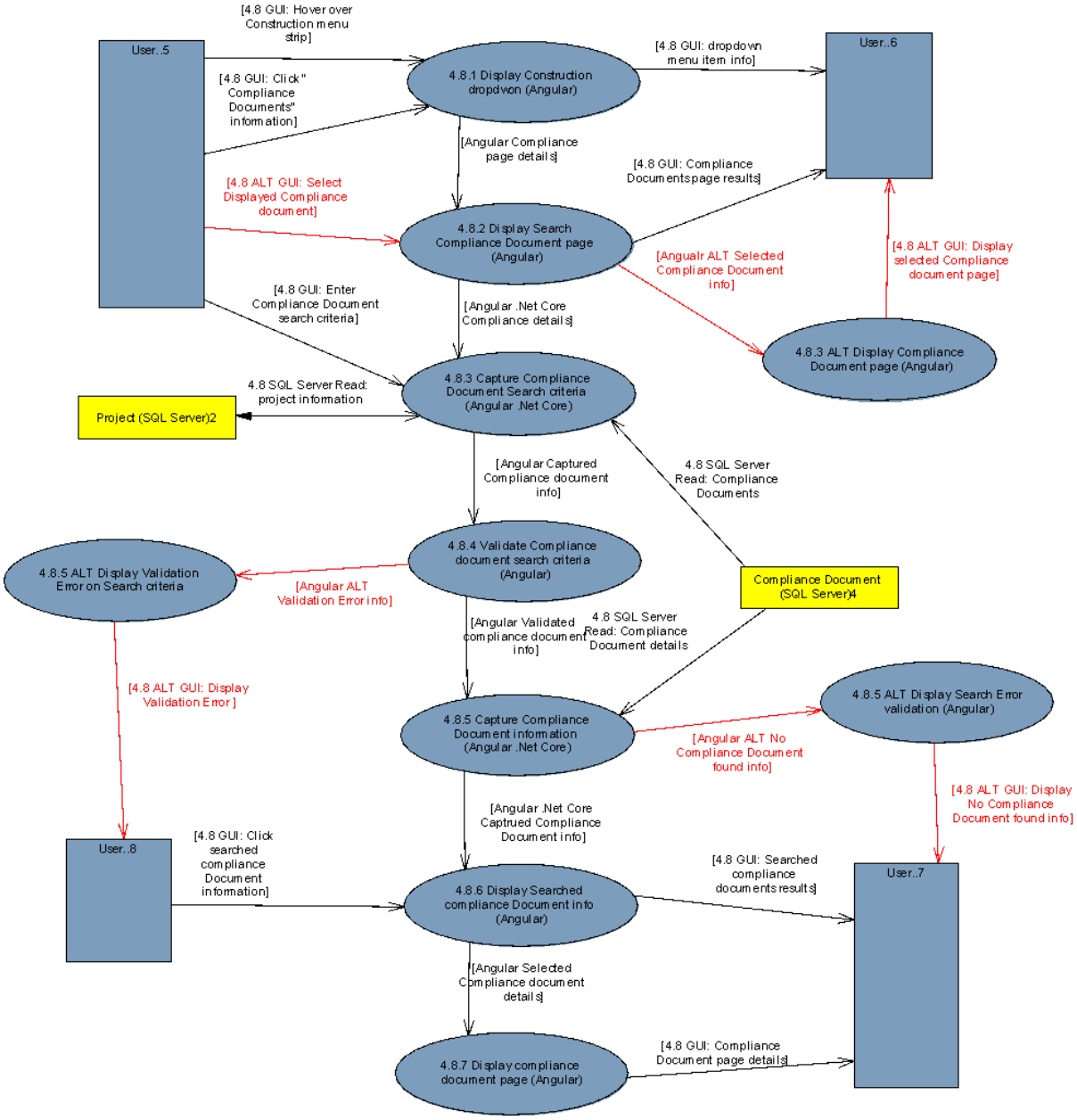


Figure 176: Primitive level data diagram4.8 Search Compliance Document

2.3.3.4.9 Search Project

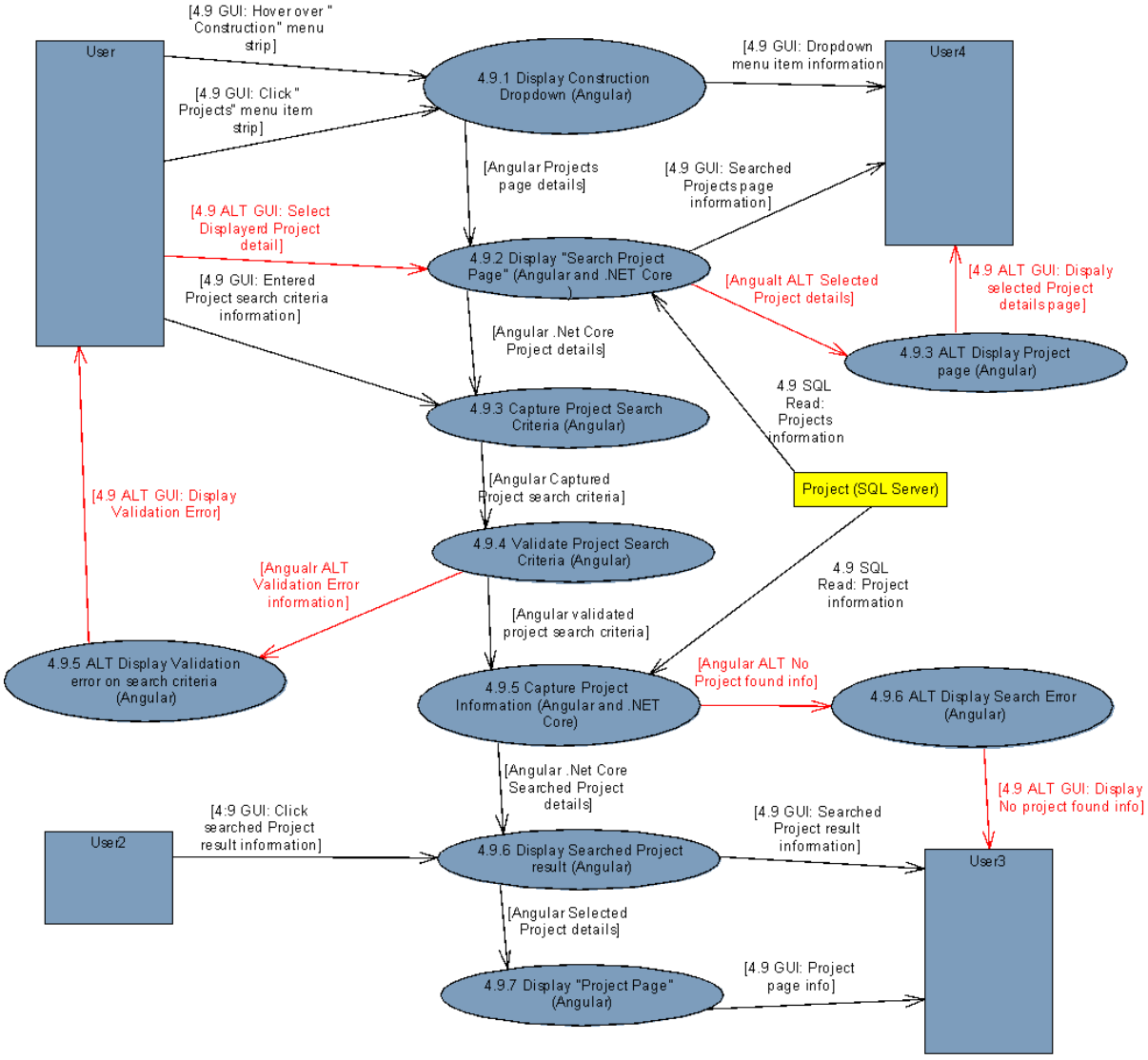


Figure 177: Primitive level data diagram4.9 Search Project

2.3.3.4.10 Create Site Progress description

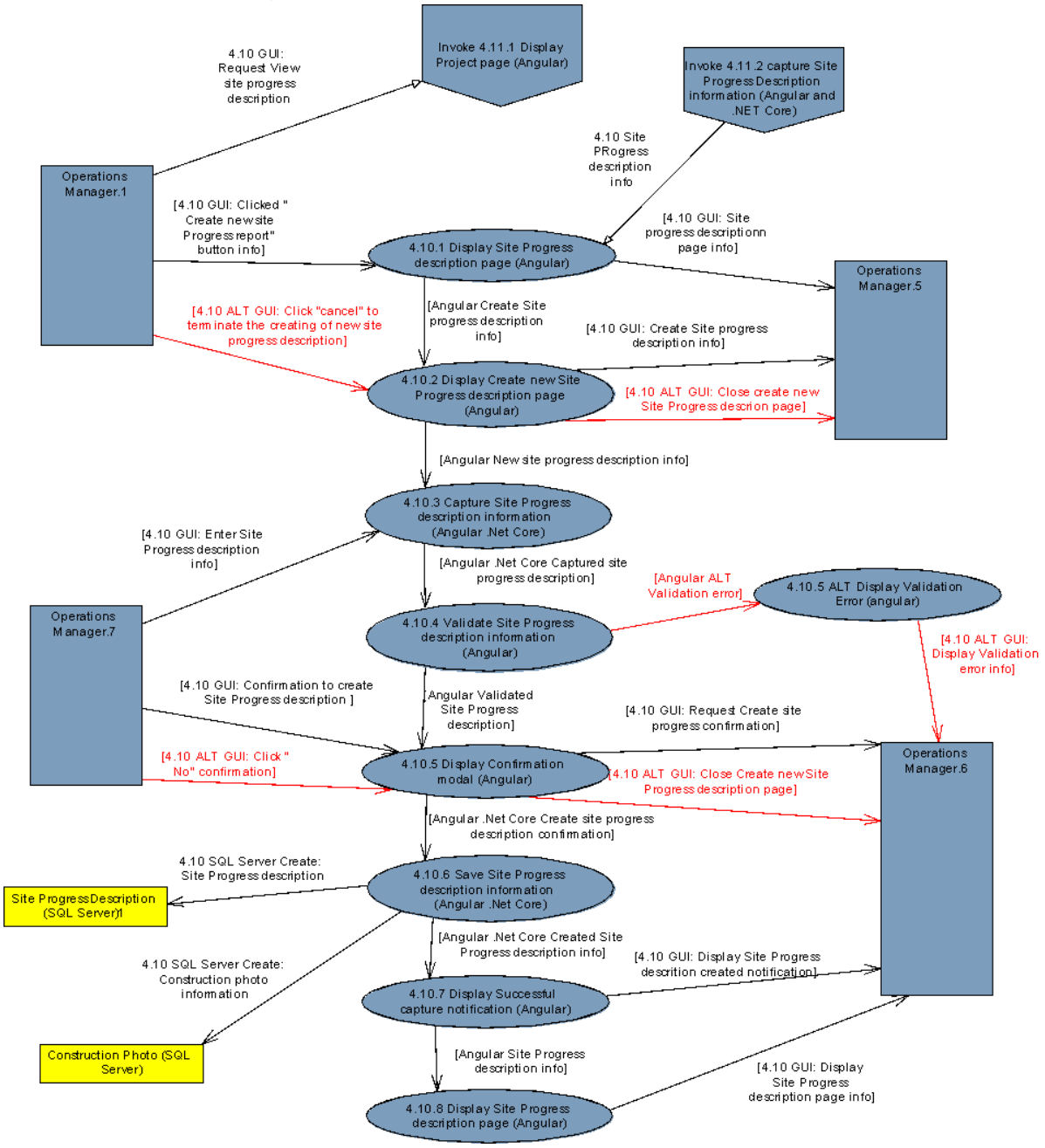


Figure 178: Primitive level data diagram4.10 Create Site Progress description

2.3.3.4.11 View Site Progress description.

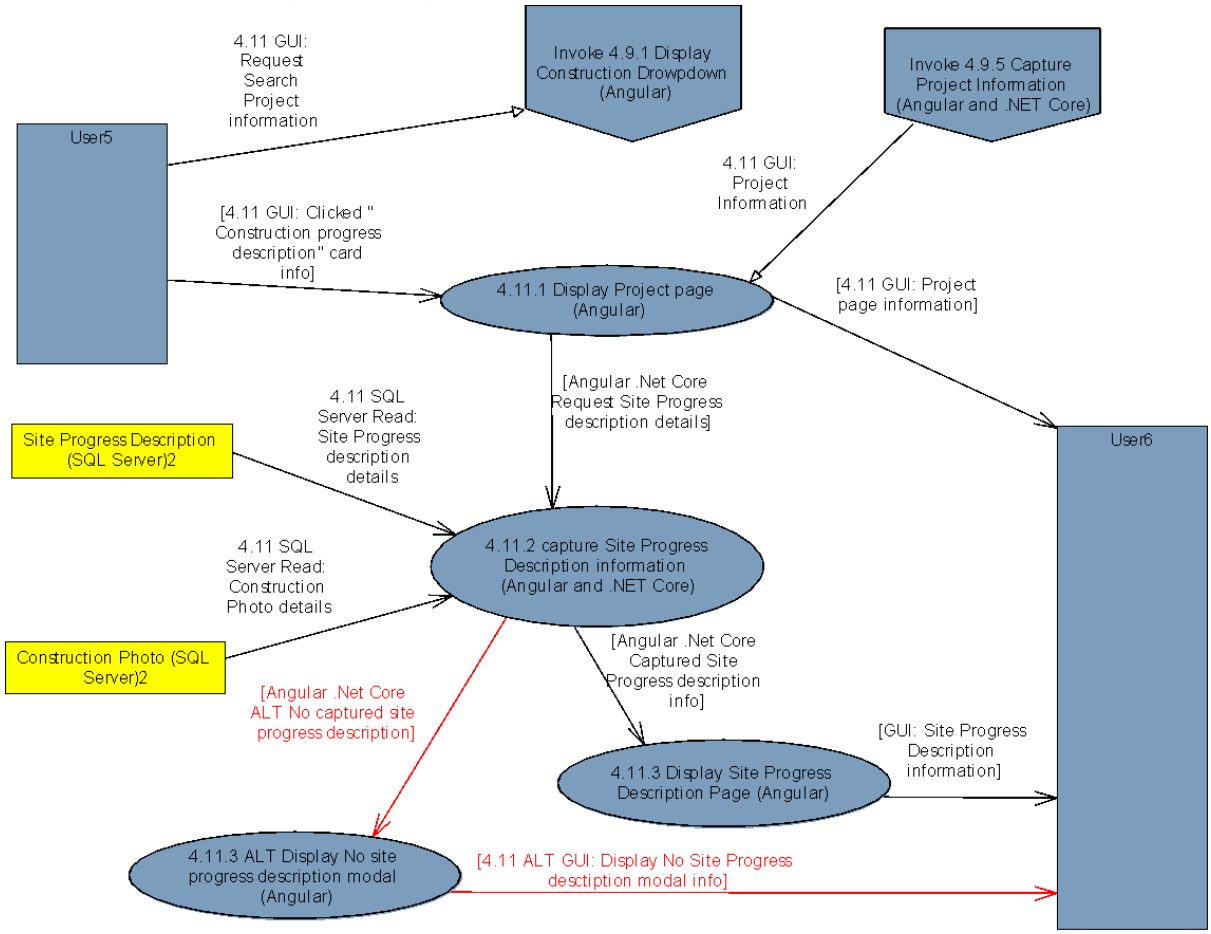


Figure 179: Primitive level data diagram4.11 View Site Progress description.

2.3.3.4.12 Upload complete site photos

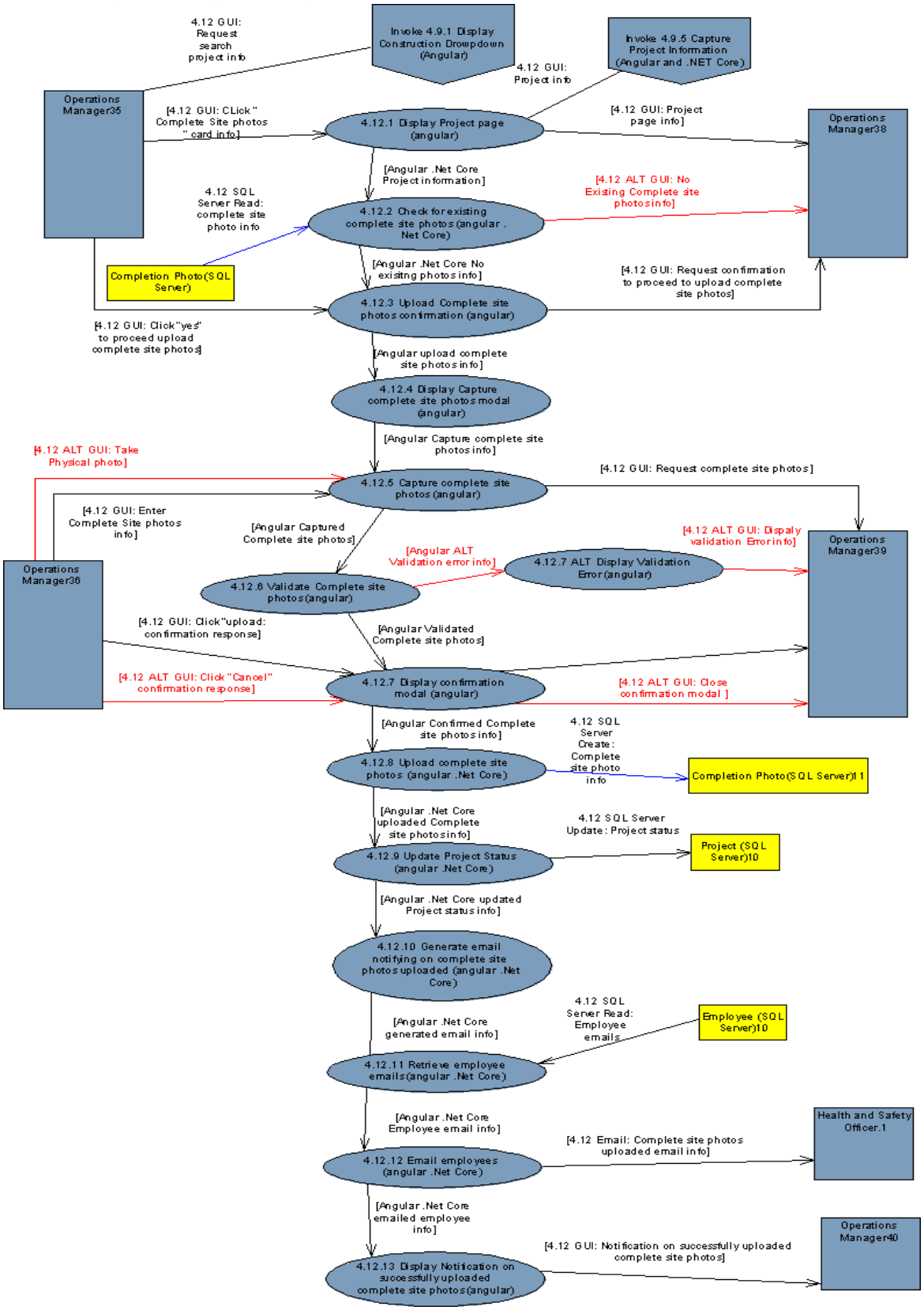


Figure 180: Primitive level data diagram4.12 Upload complete site photos

2.3.3.4.13 Search complete site photos

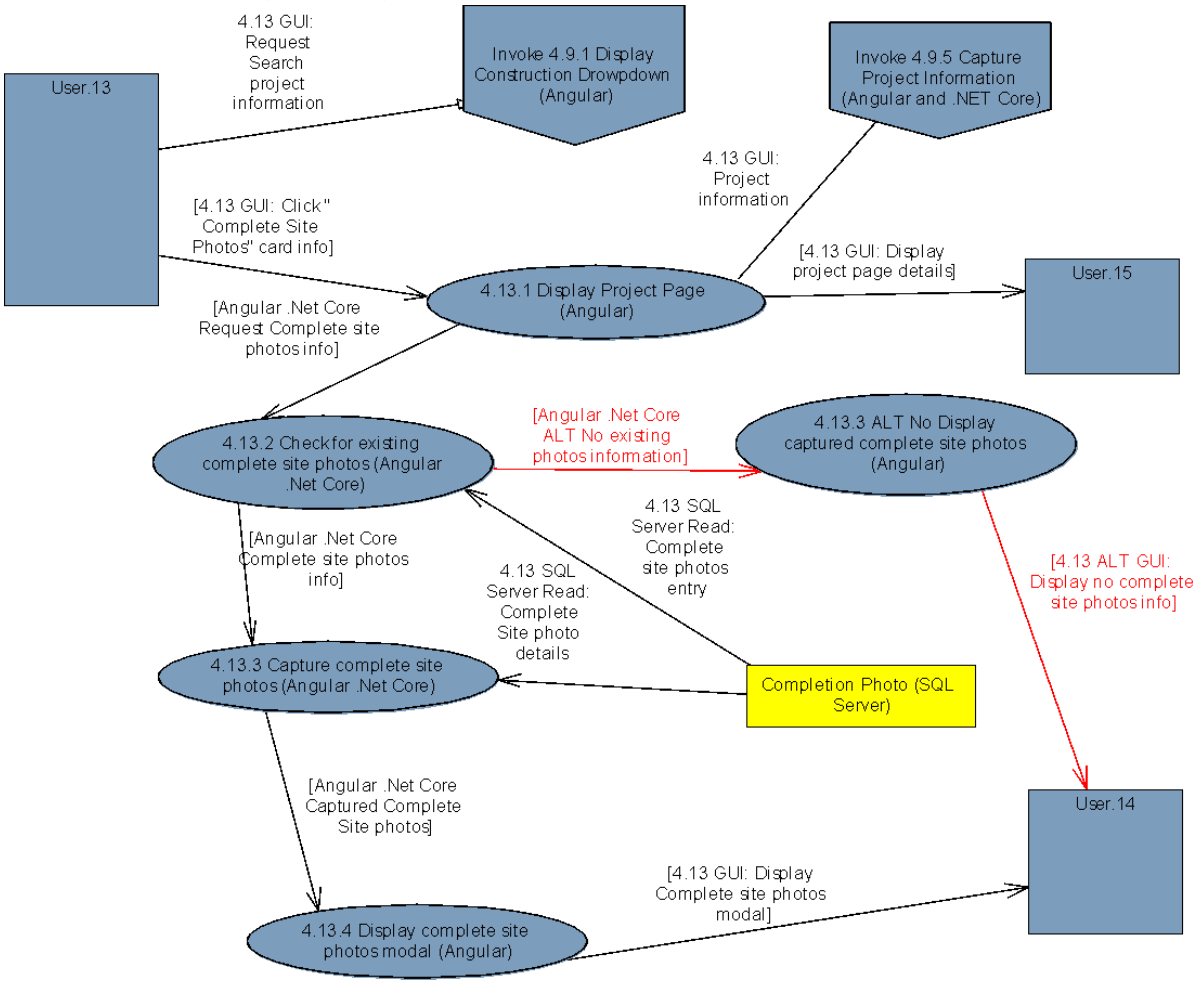


Figure 181: Primitive level data diagram4.13 Search complete site photos

2.3.3.4.14 Upload Radio Frequency report

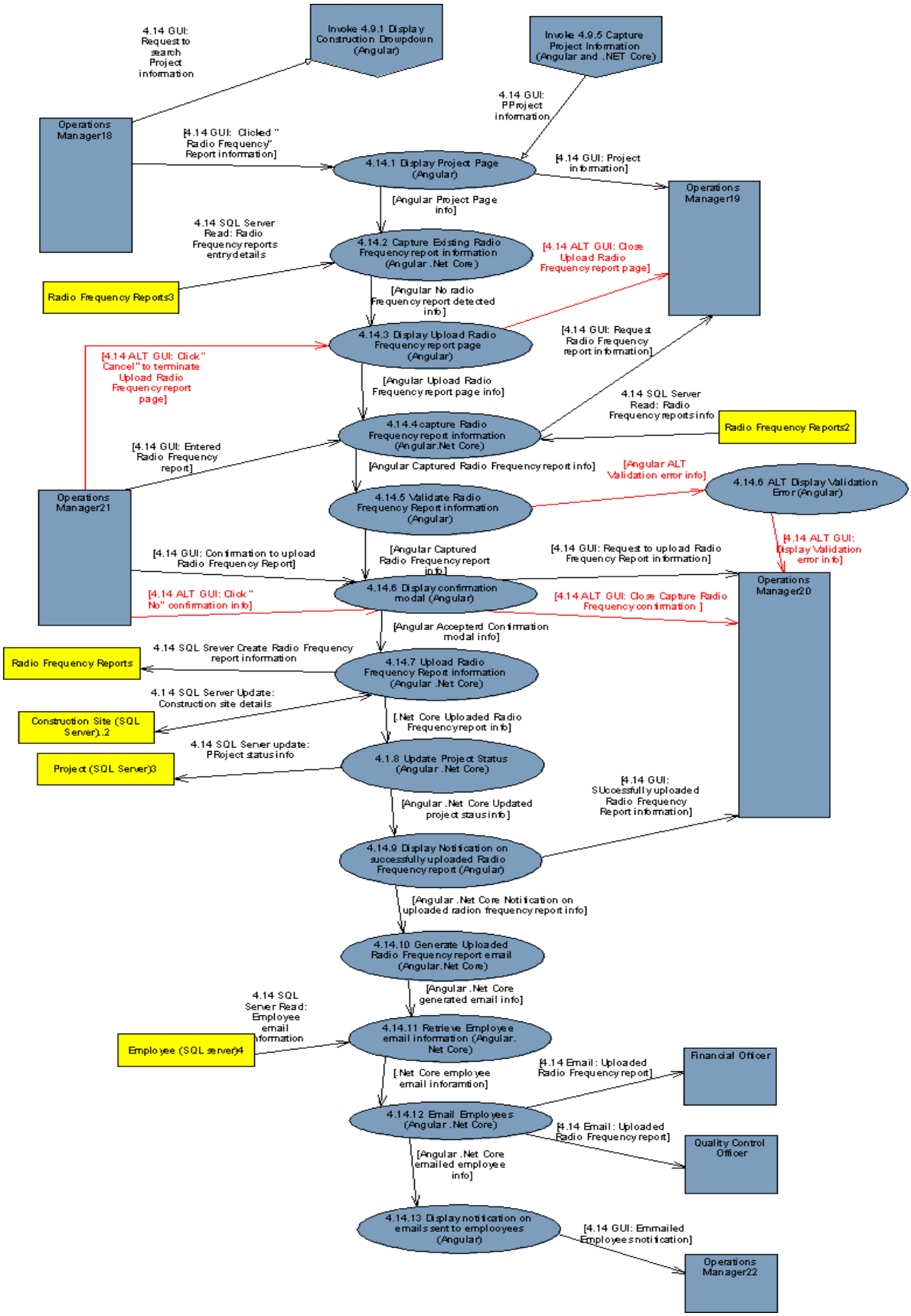


Figure 182: Primitive level data diagram4.14 Upload Radio Frequency report

2.3.3.4.15 Search Radio Frequency report

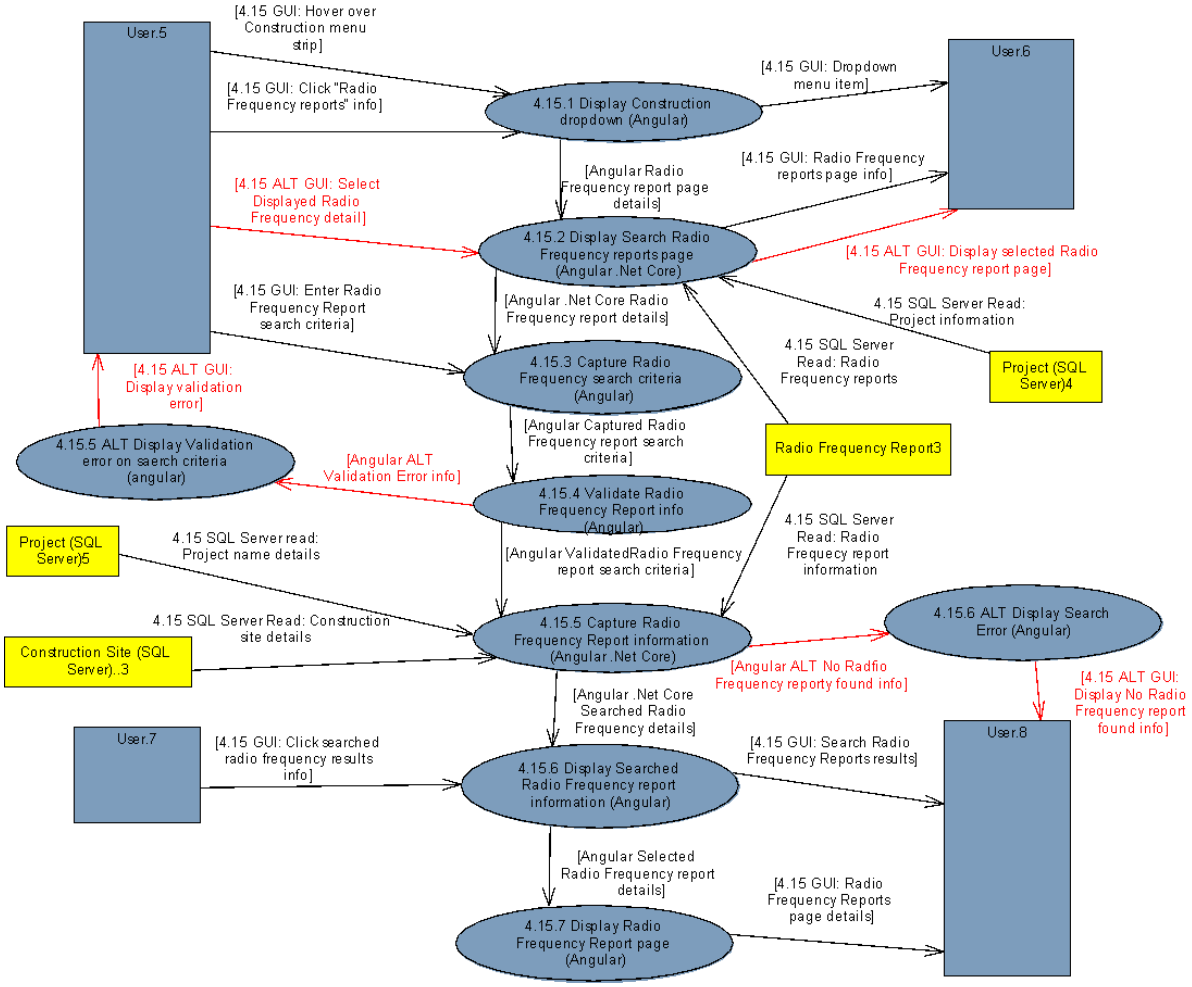


Figure 183: Primitive level data diagram4.15 Search Radio Frequency report

2.3.3.4.16 Request Site Visit

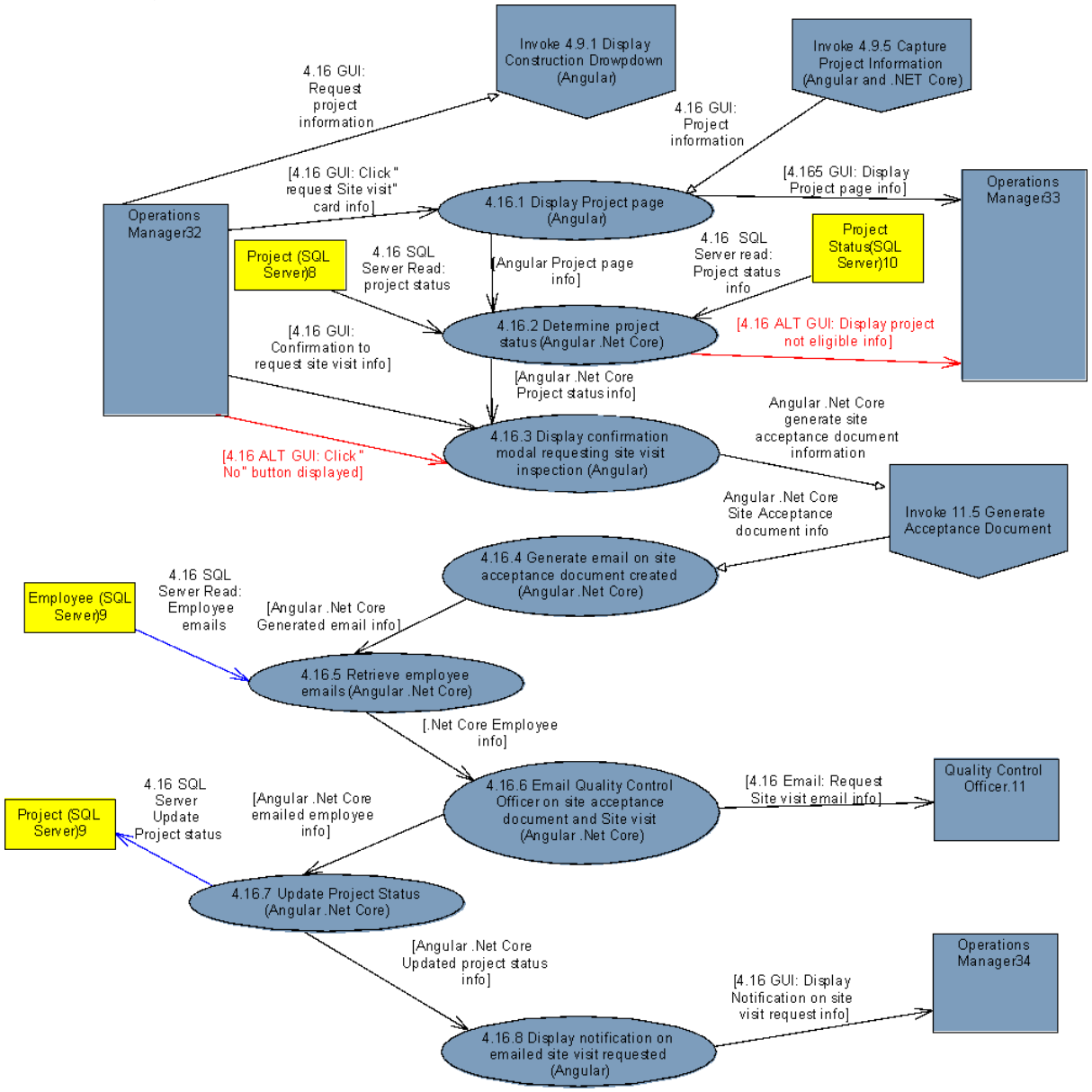


Figure 184: Primitive level data diagram4.16 Request Site Visit

2.3.3.4.17 Create Snag

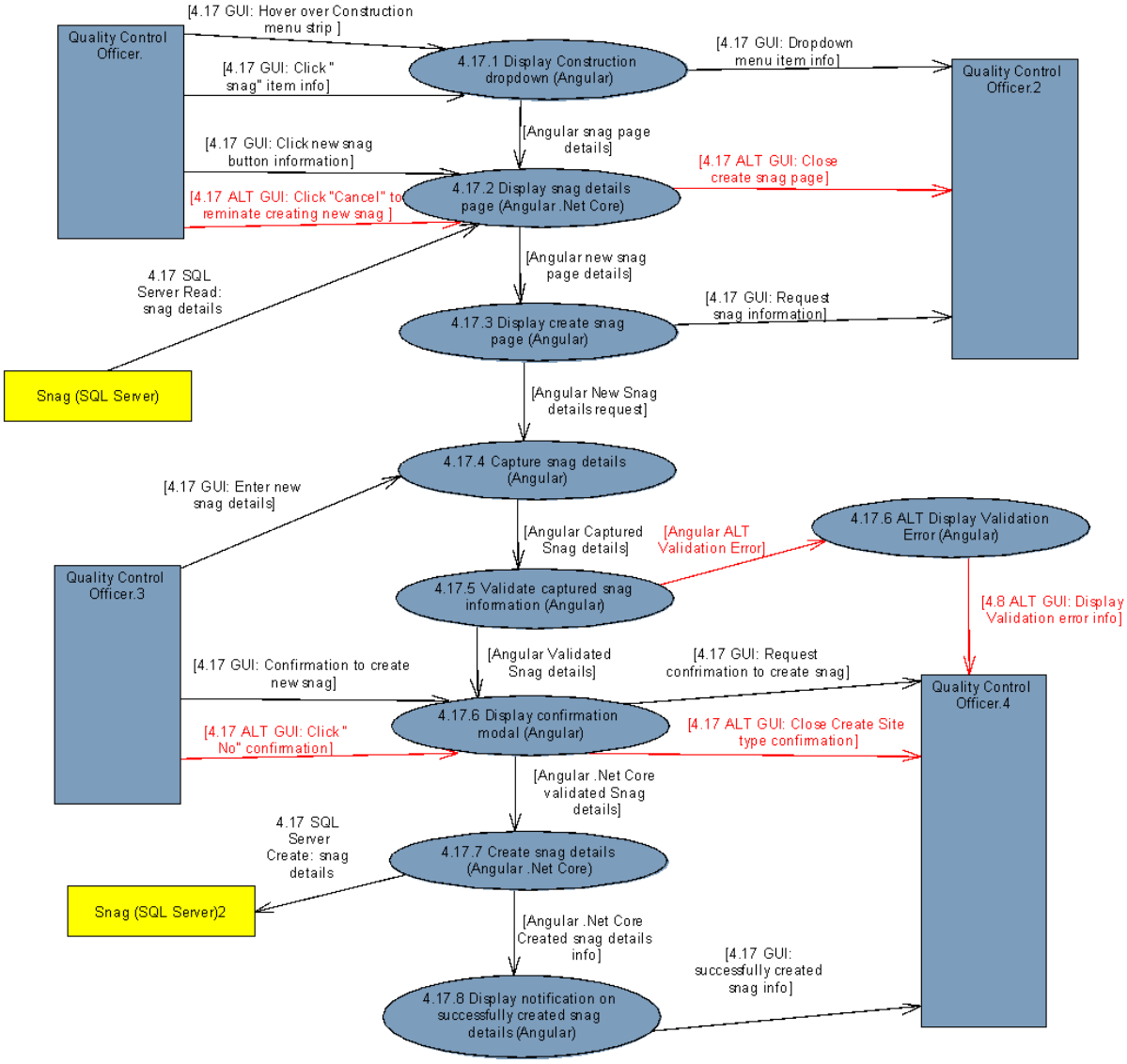


Figure 185: Primitive level data diagram4.17 Create Snag

2.3.3.4.18 Search Snag

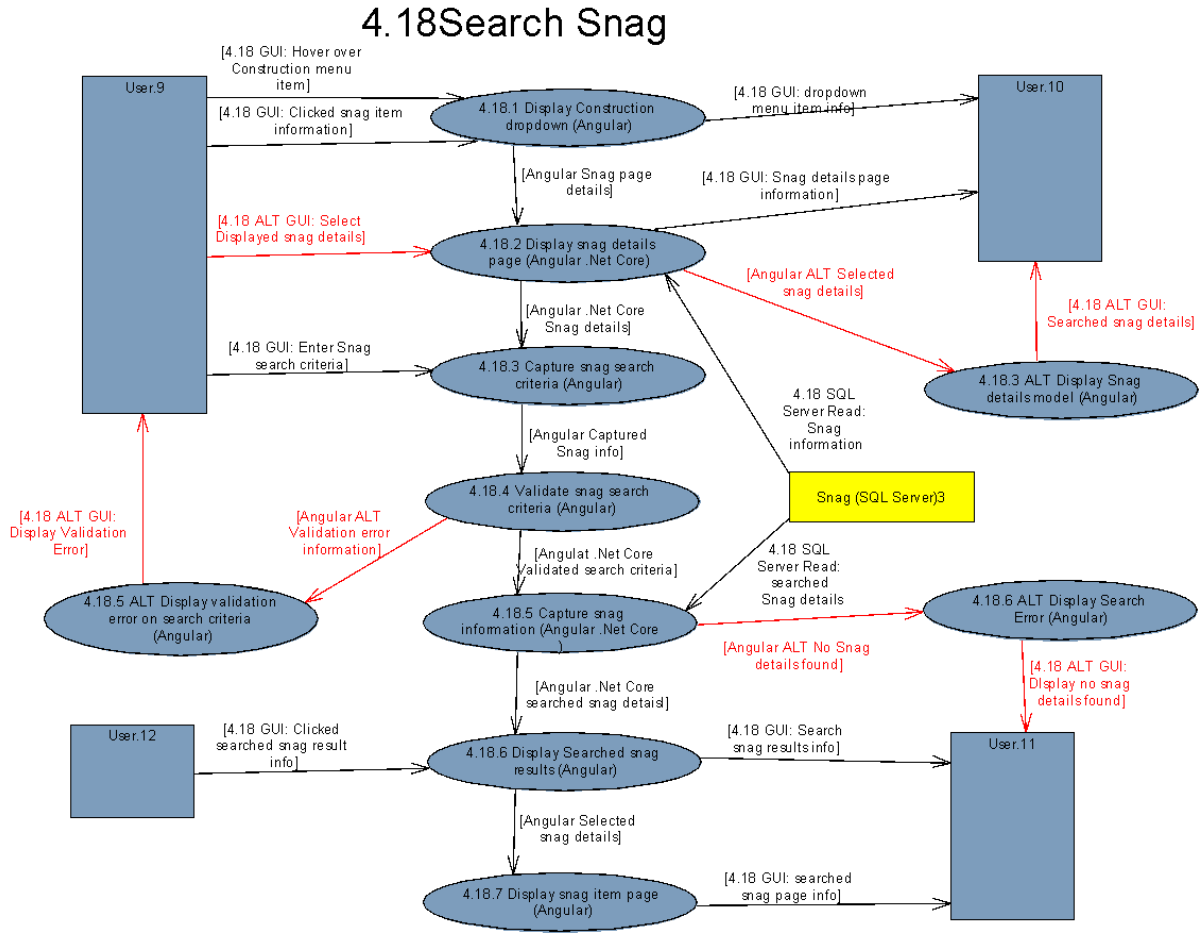


Figure 186: Primitive level data diagram4.18 Search Snag

2.3.3.4.19 Edit Snag

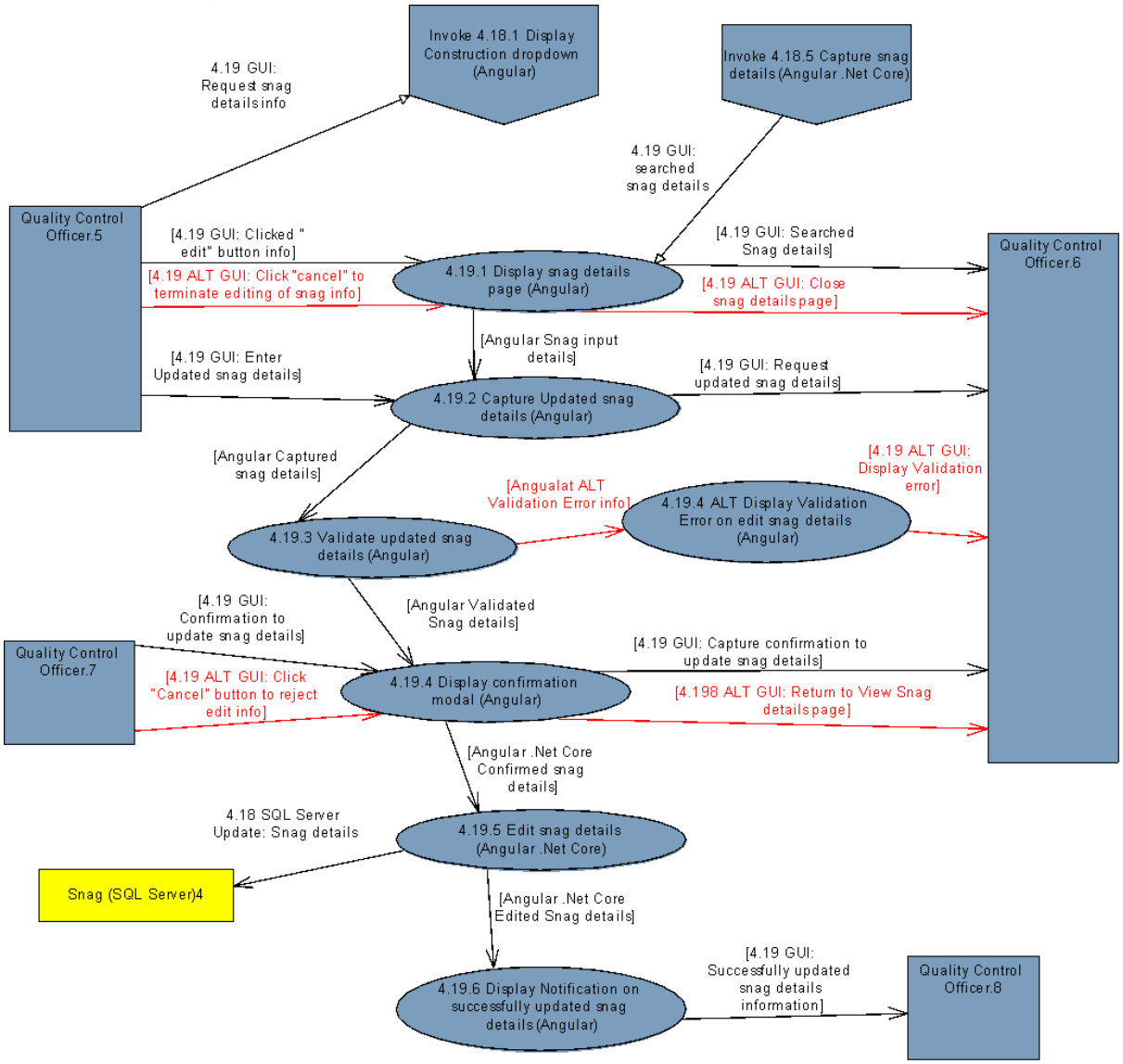


Figure 187: Primitive level data diagram4.19 Edit Snag

2.3.3.4.20 Edit Acceptance Document

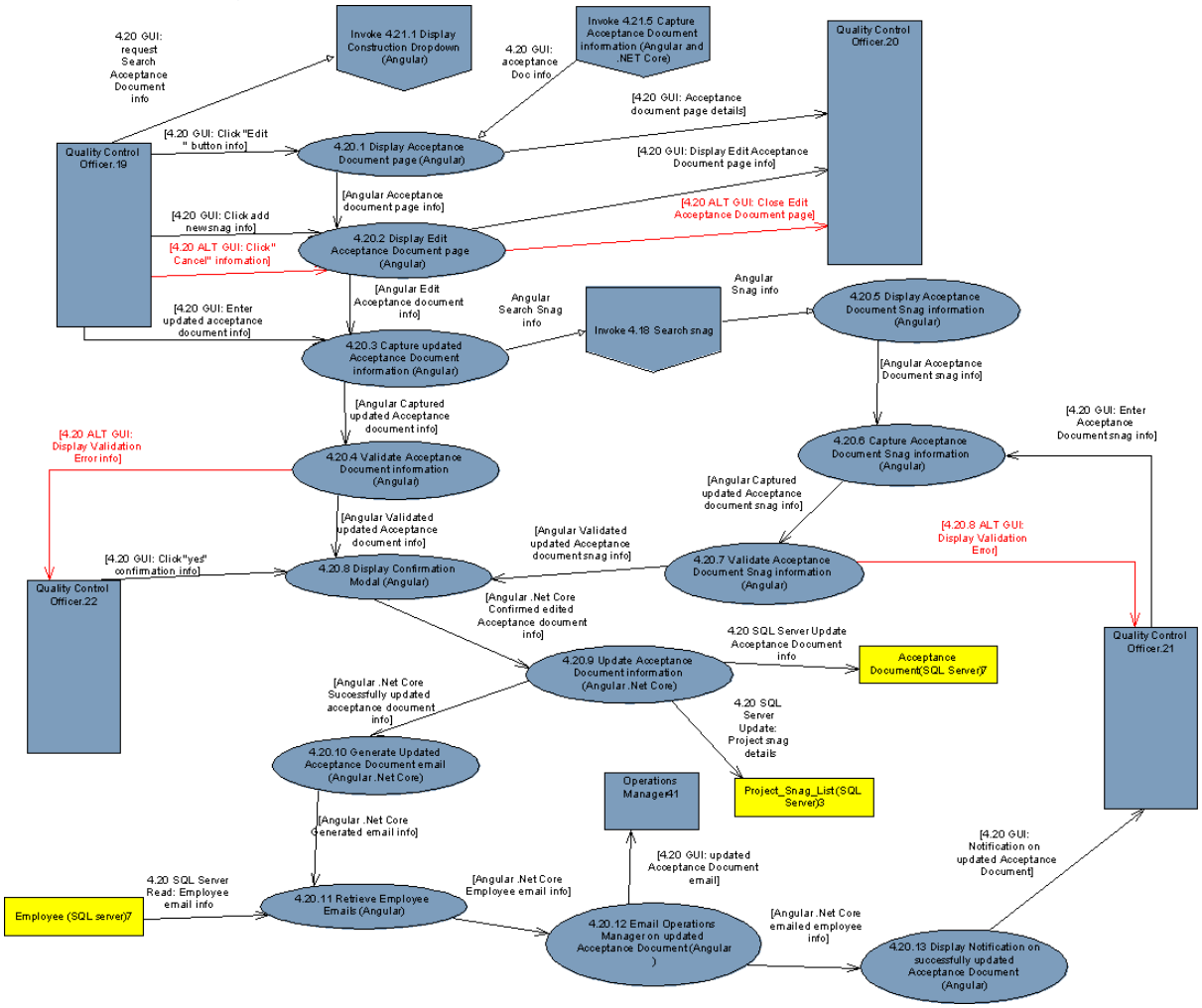


Figure 188: Primitive level data diagram4.20 Edit Acceptance Document

2.3.3.4.21 Search Acceptance document

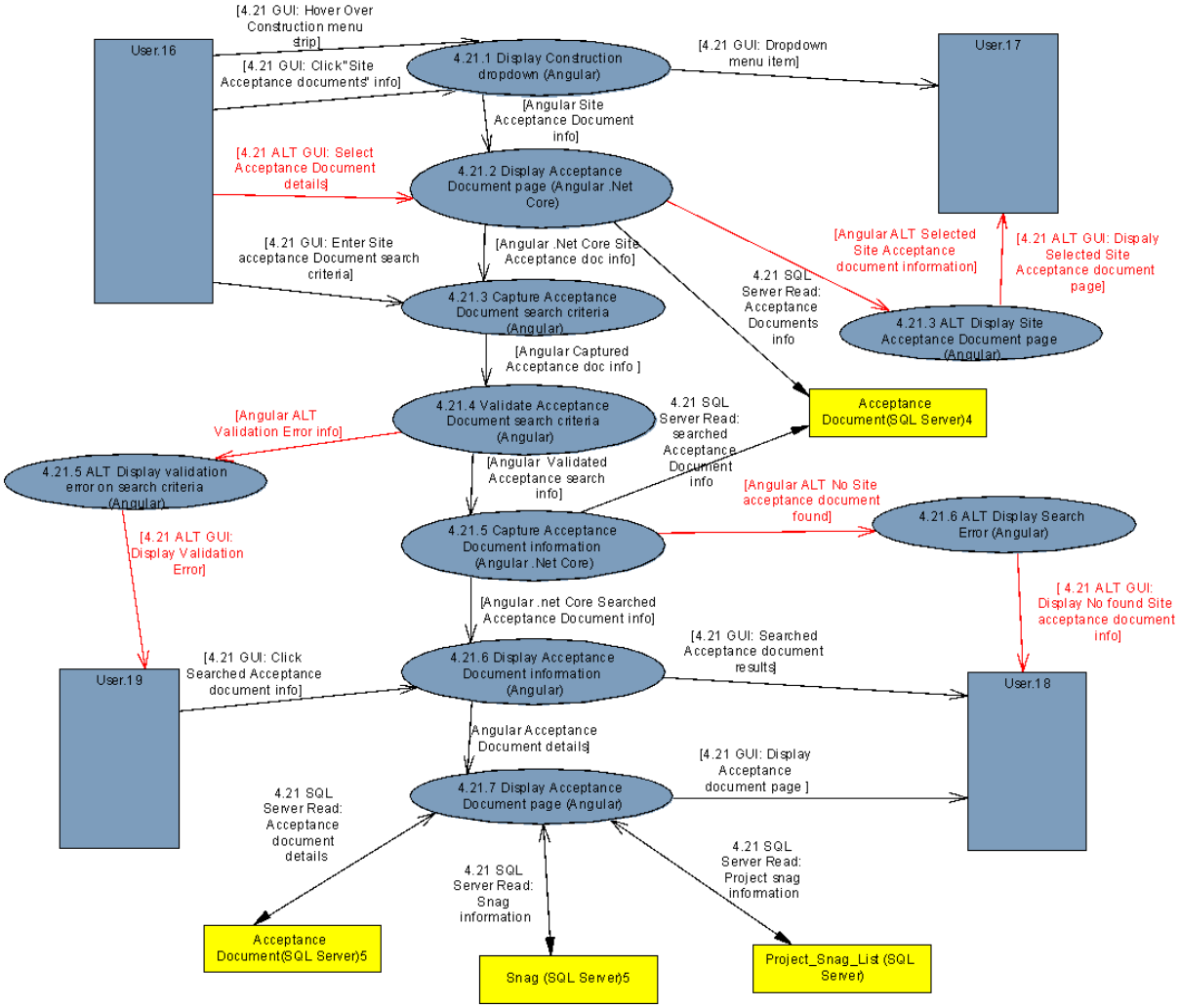


Figure 189: Primitive level data diagram4.21 Search Acceptance document

2.3.3.4.22 Confirm acceptance document

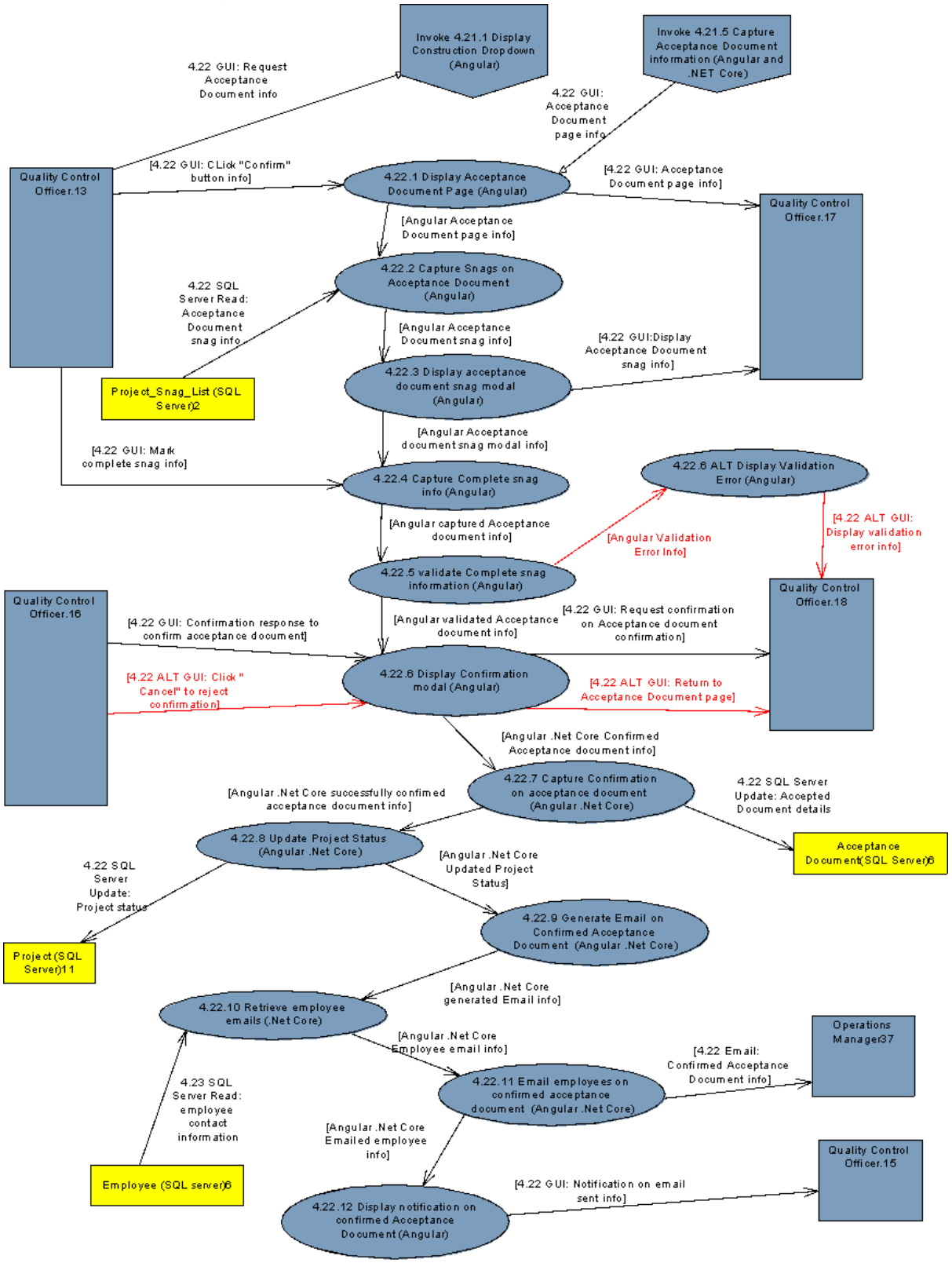


Figure 190: Primitive level data diagram4.22 Confirm acceptance document

2.3.3.4.23 Capture Site Sign-off

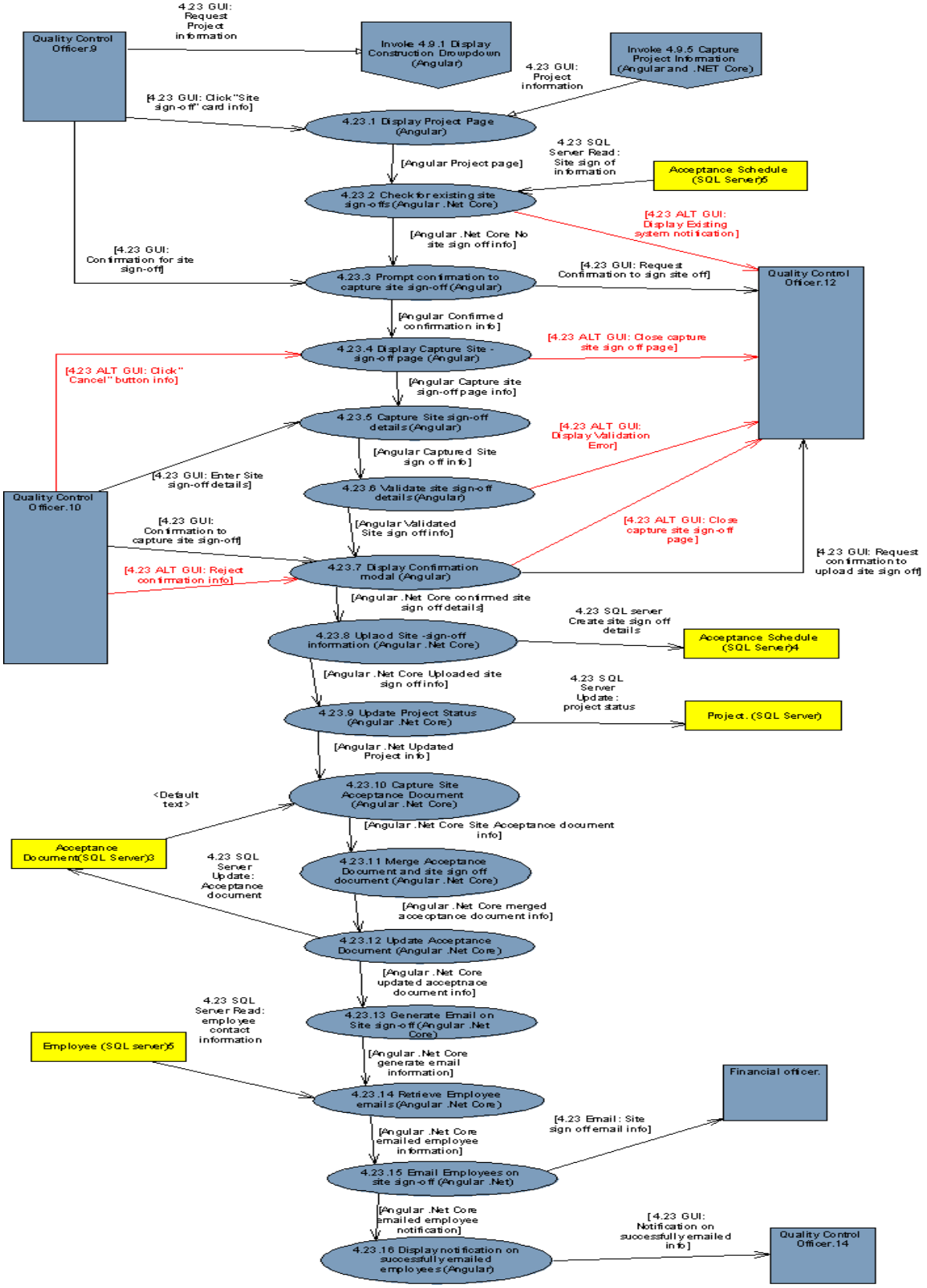


Figure 191: Primitive level data diagram4.23 Capture Site Sign-off

2.3.3.4.24 Search Site Sign-off

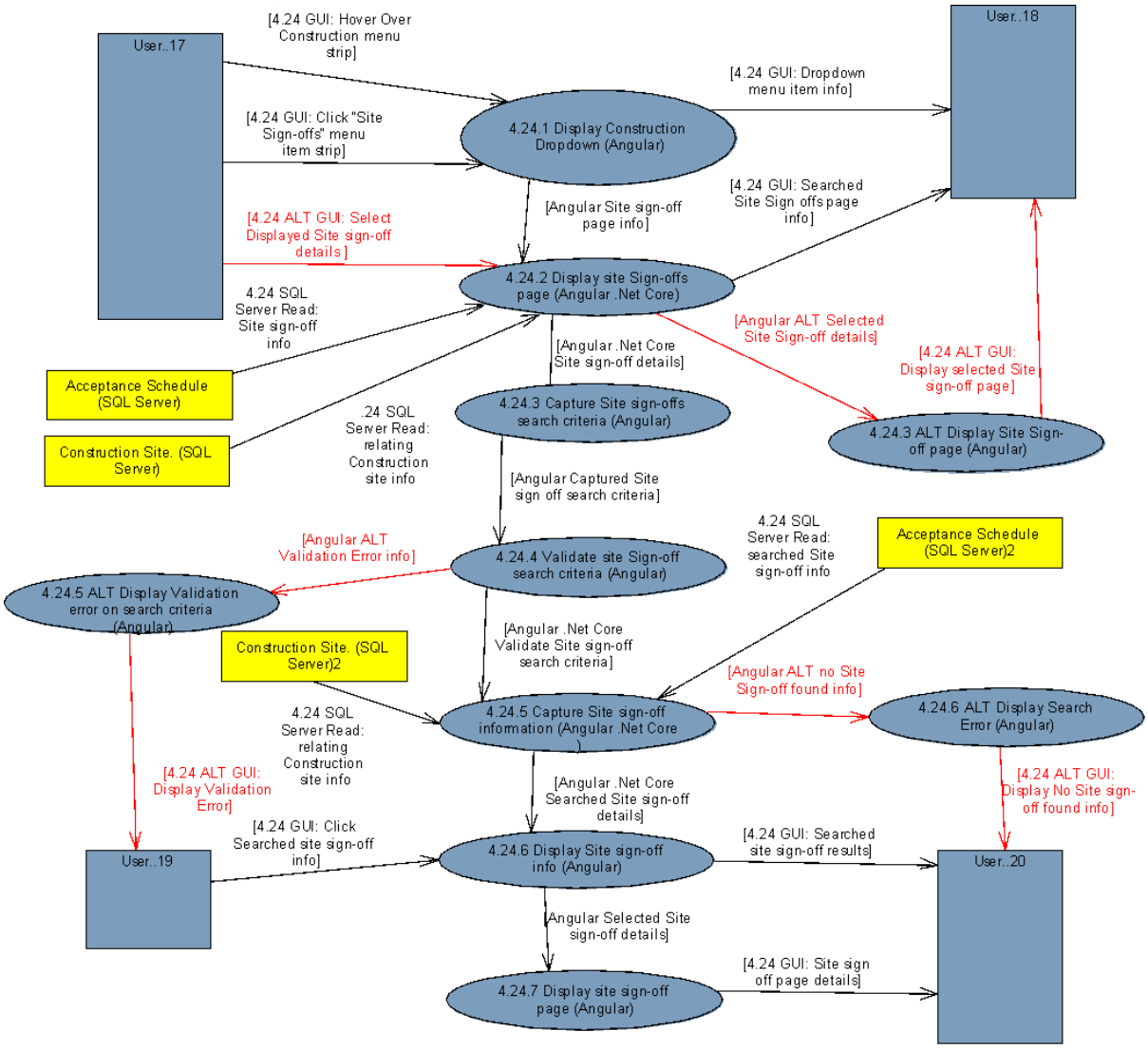


Figure 192: Primitive level data diagram4.24 Search Site Sign-off

2.3.3.4.25 Capture Commission report

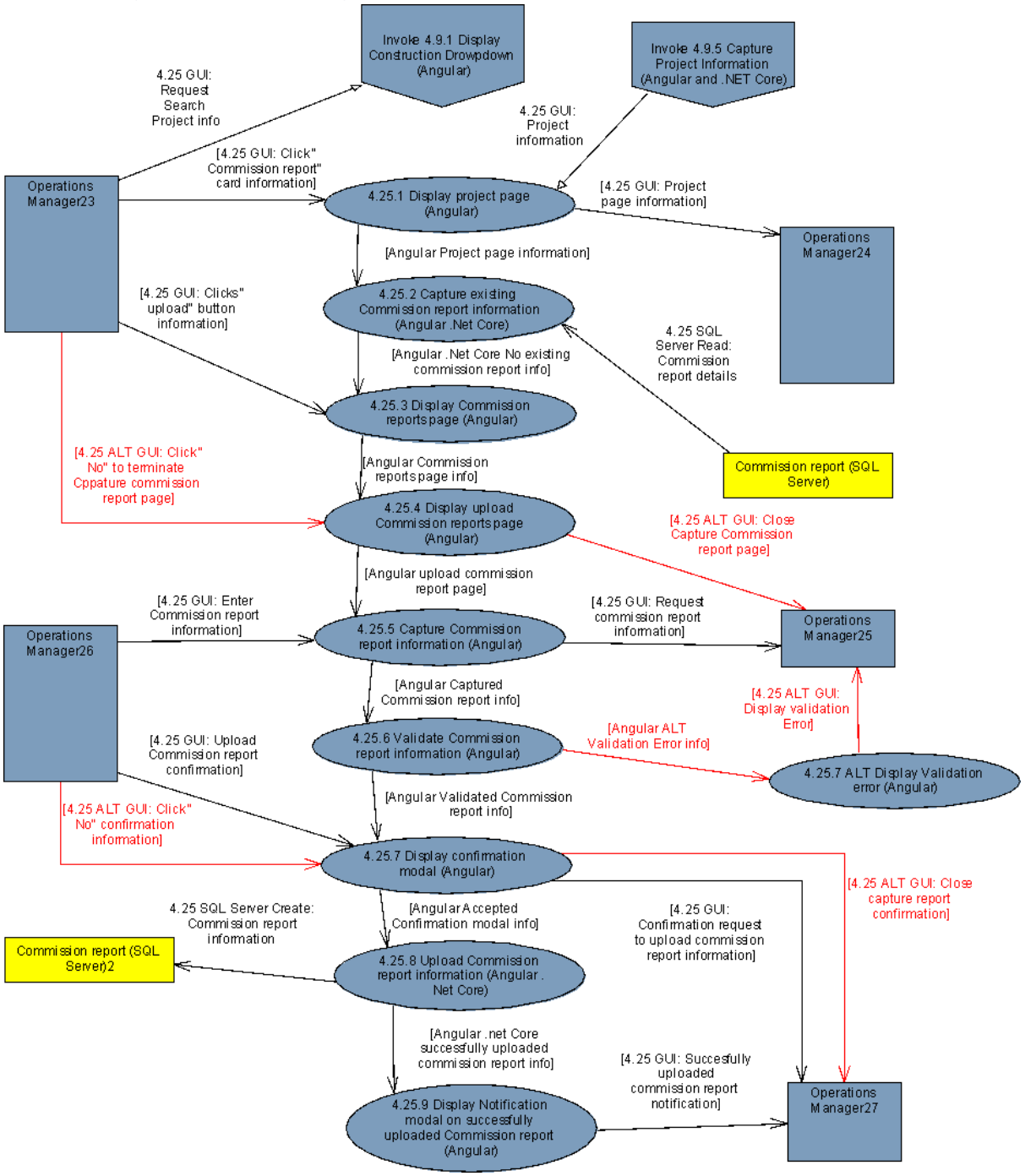


Figure 193: Primitive level data diagram4.25 Capture Commission report

2.3.3.4.26 Search Commission report

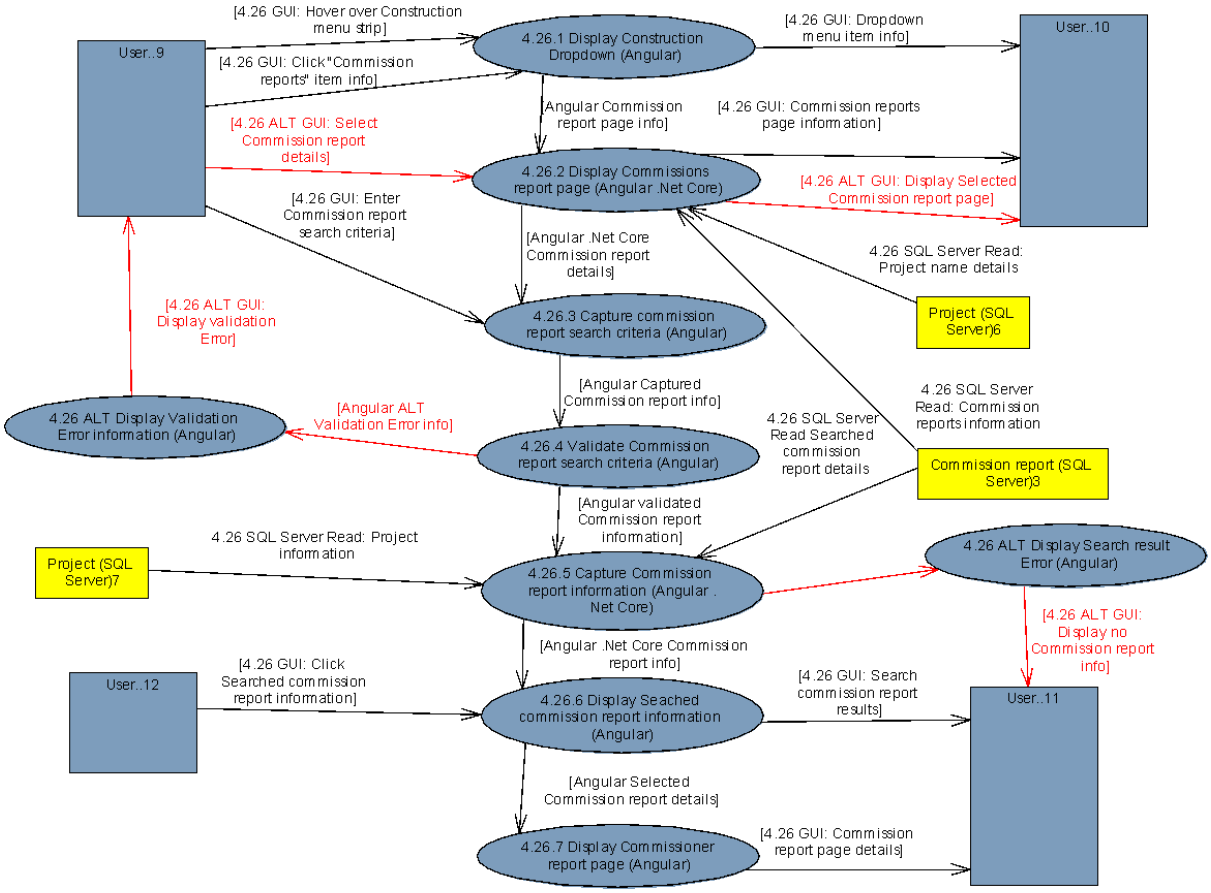


Figure 194: Primitive level data diagram4.26 Search Commission report

2.3.3.4.27 Create Site Type

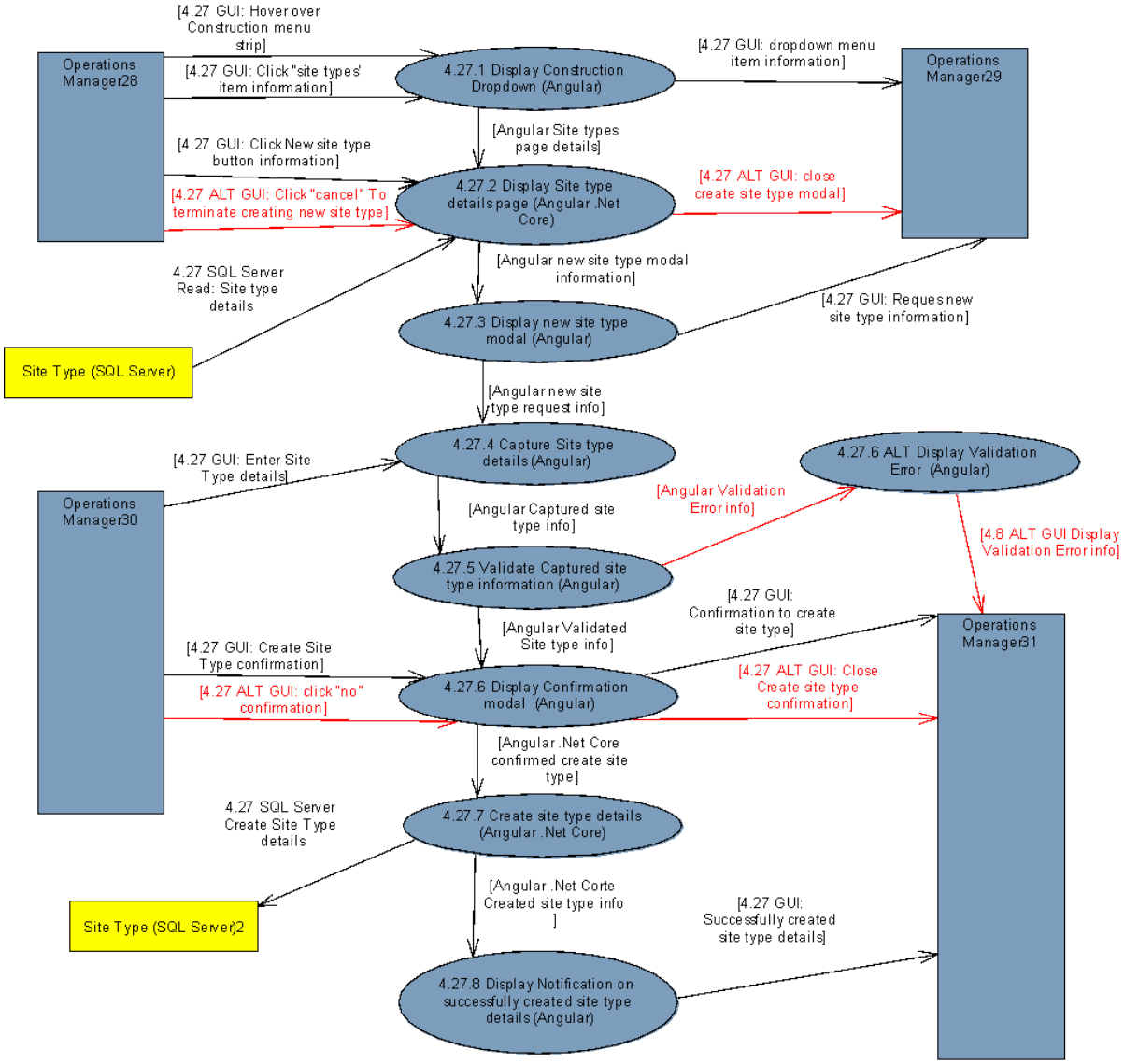


Figure 195: Primitive level data diagram4.27 Create Site Type

2.3.3.4.28 Search Site type

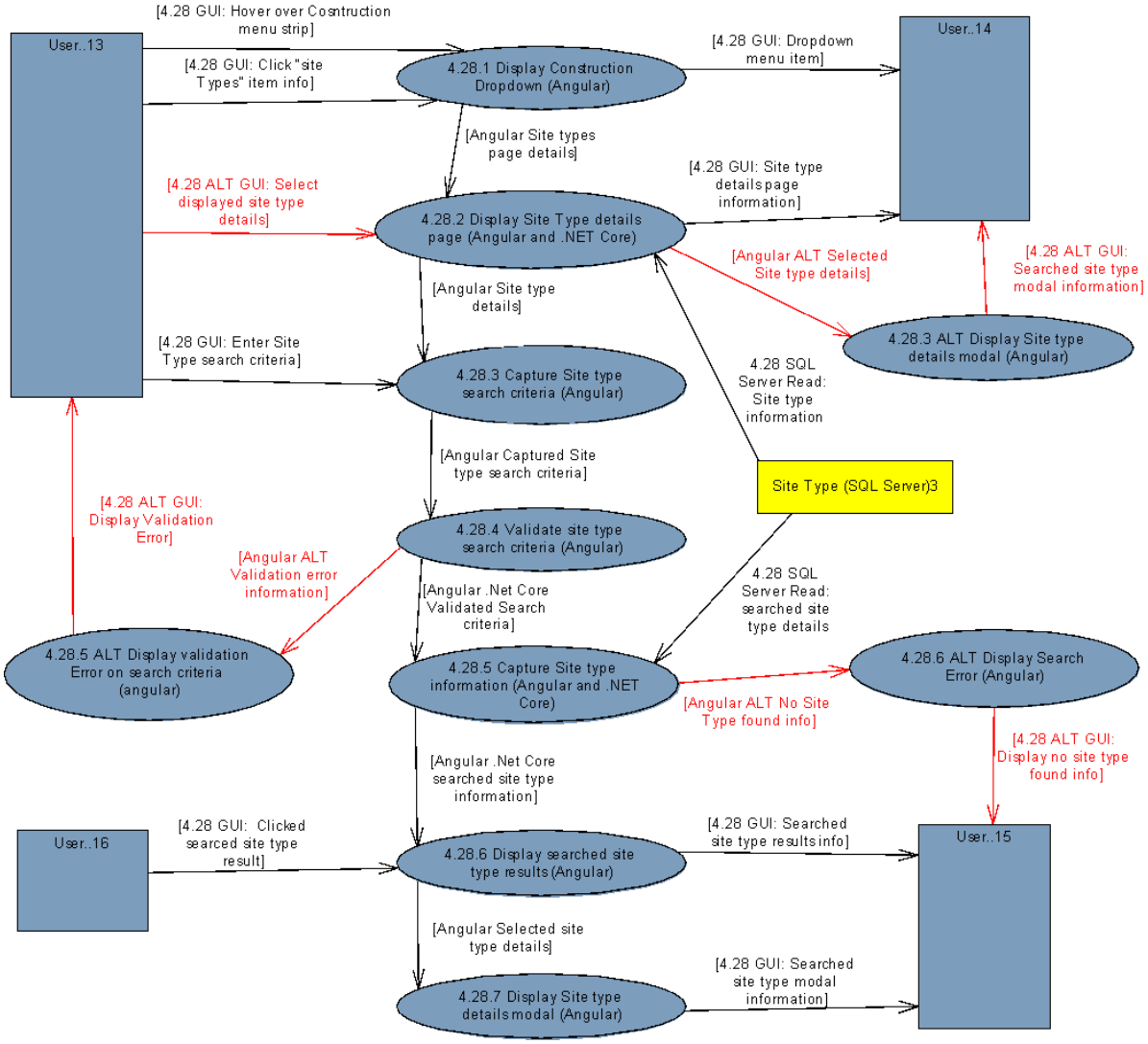


Figure 196: Primitive level data diagram4.28 Search Site type

2.3.3.4.29 Edit Site Type

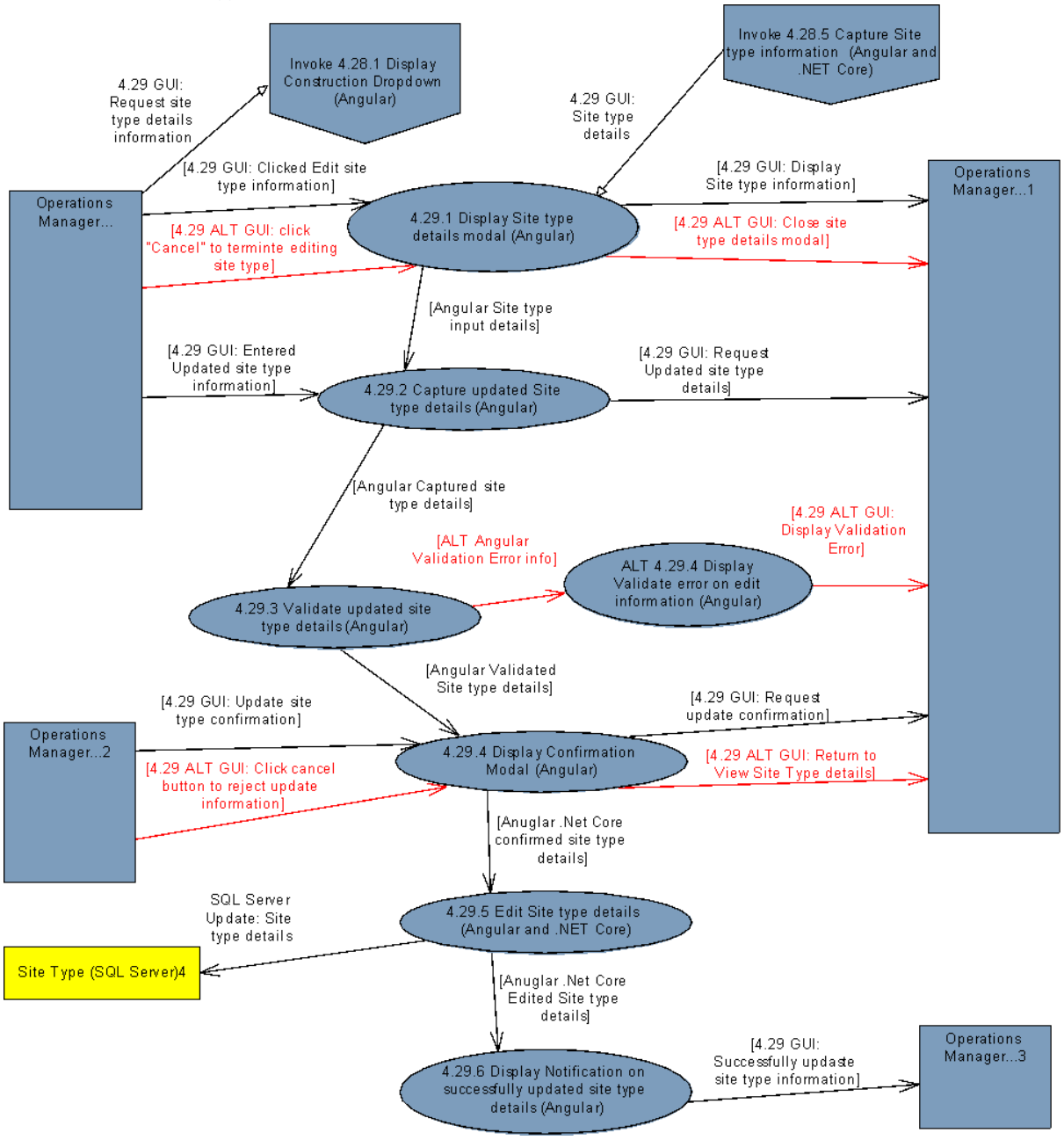


Figure 197: Primitive level data diagram4.29 Edit Site Type

2.3.3.5. Invoicing Subsystem

2.3.3.5.1 Search Invoice

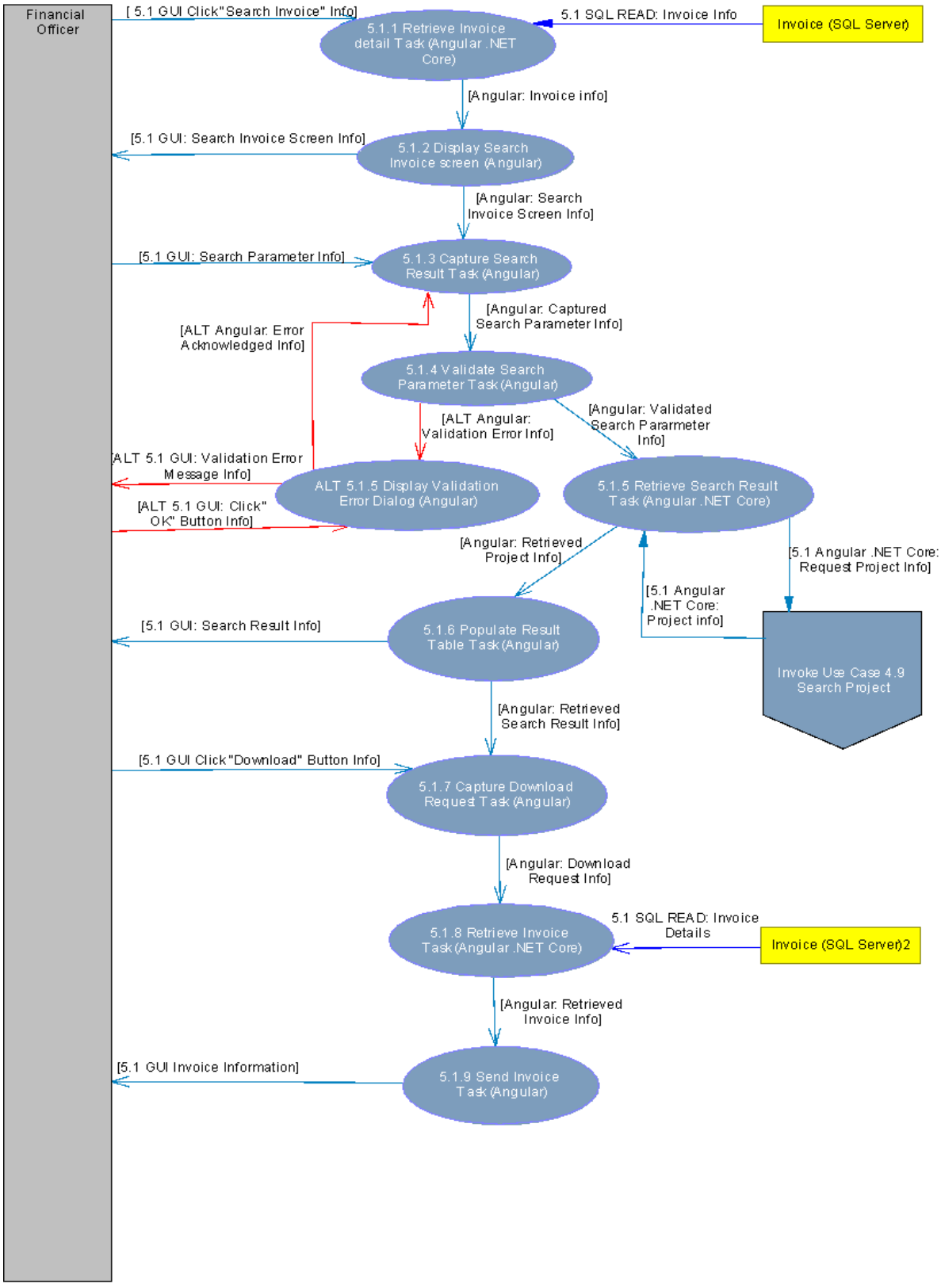


Figure 198: Primitive level data diagram5.1 Search Invoice

2.3.3.5.2 Send Invoice

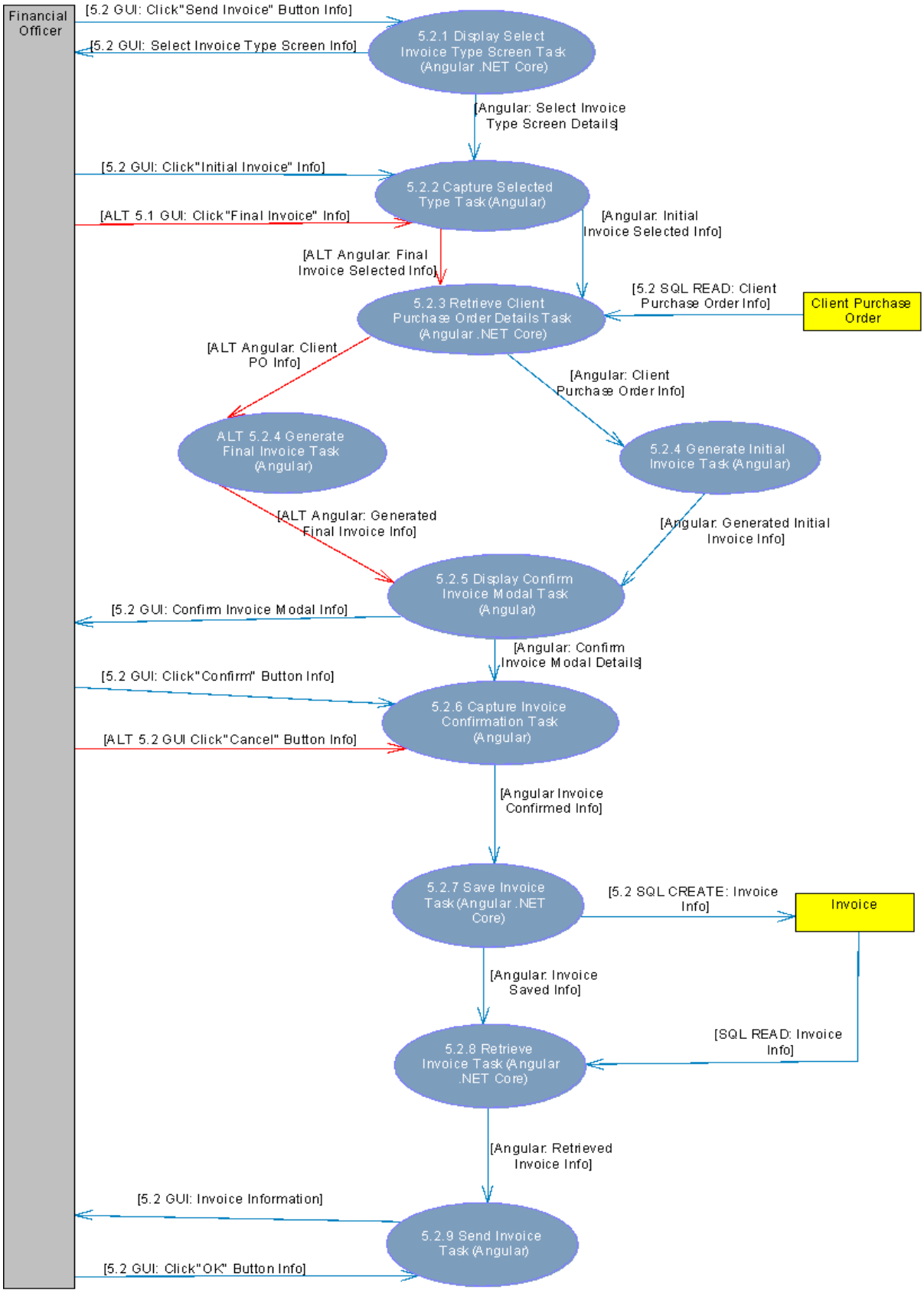


Figure 199: Primitive level data diagram5.2 Send Invoice

2.3.3.5.3 Capture Proof of Payment

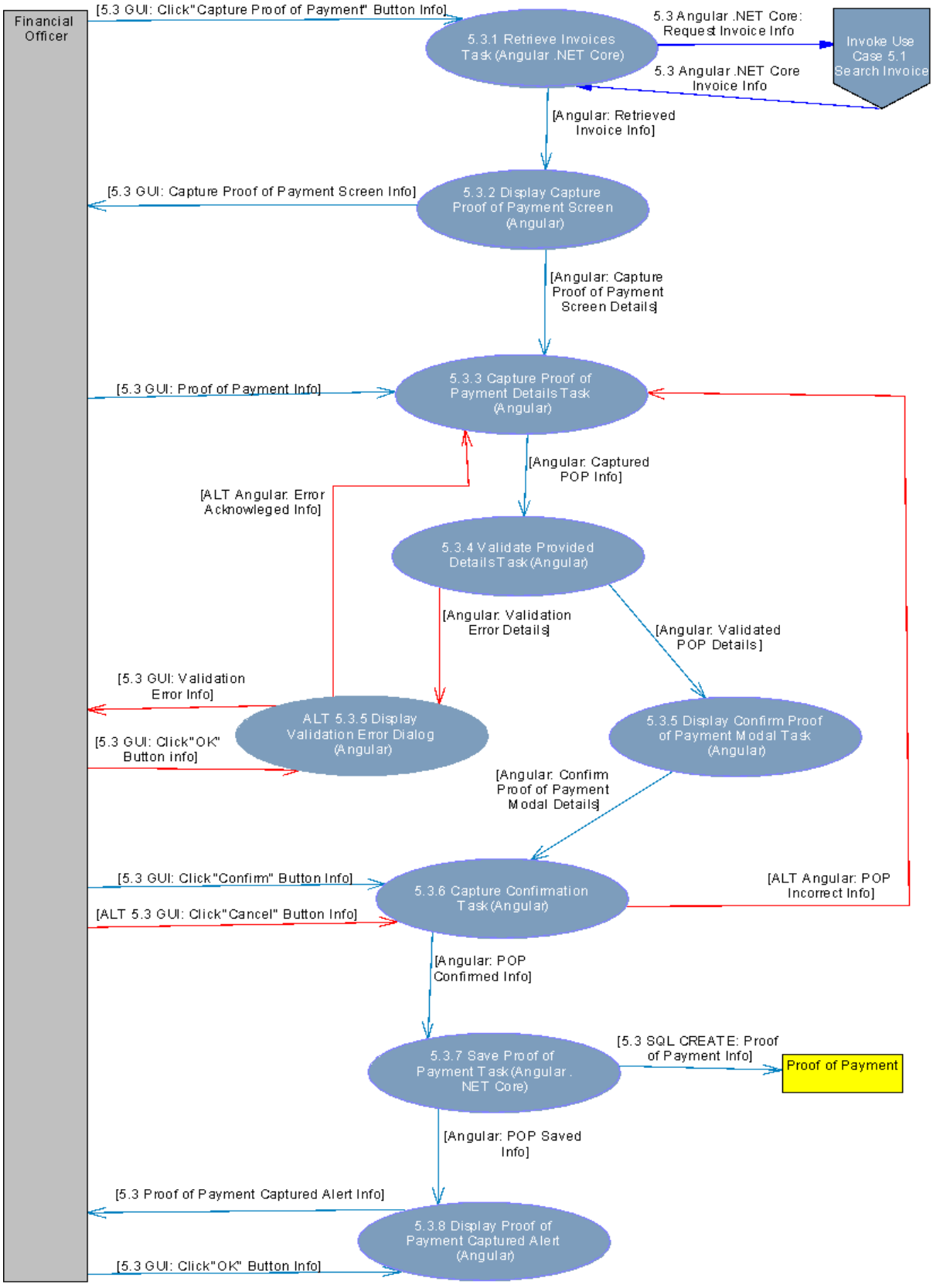


Figure 200: Primitive level data diagram5.3 Capture Proof of Payment

2.3.3.5.4 Capture Claim

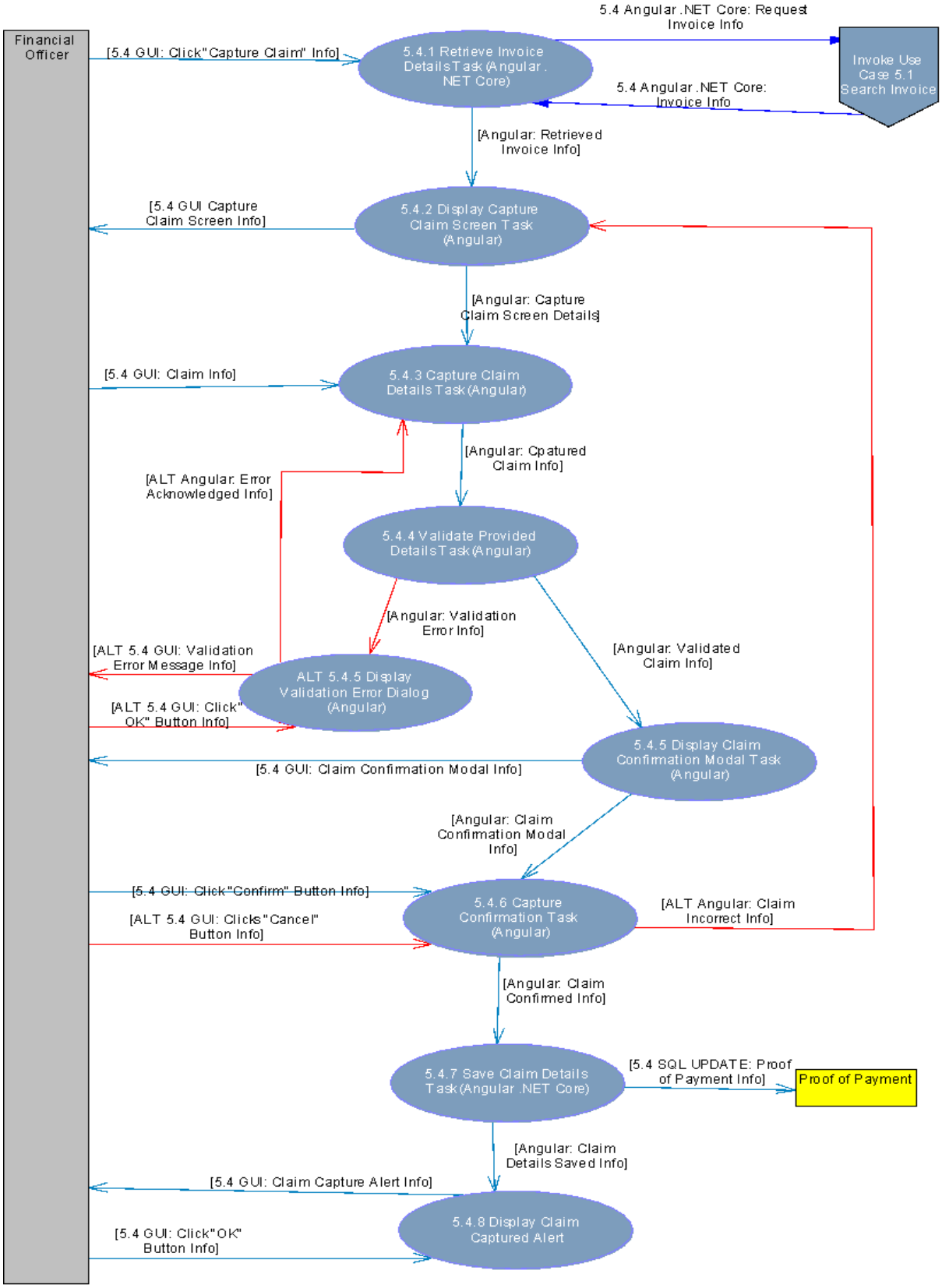


Figure 201: Primitive level data diagram5.4 Capture Claim

2.3.3.5.5 Search Proof of Payment

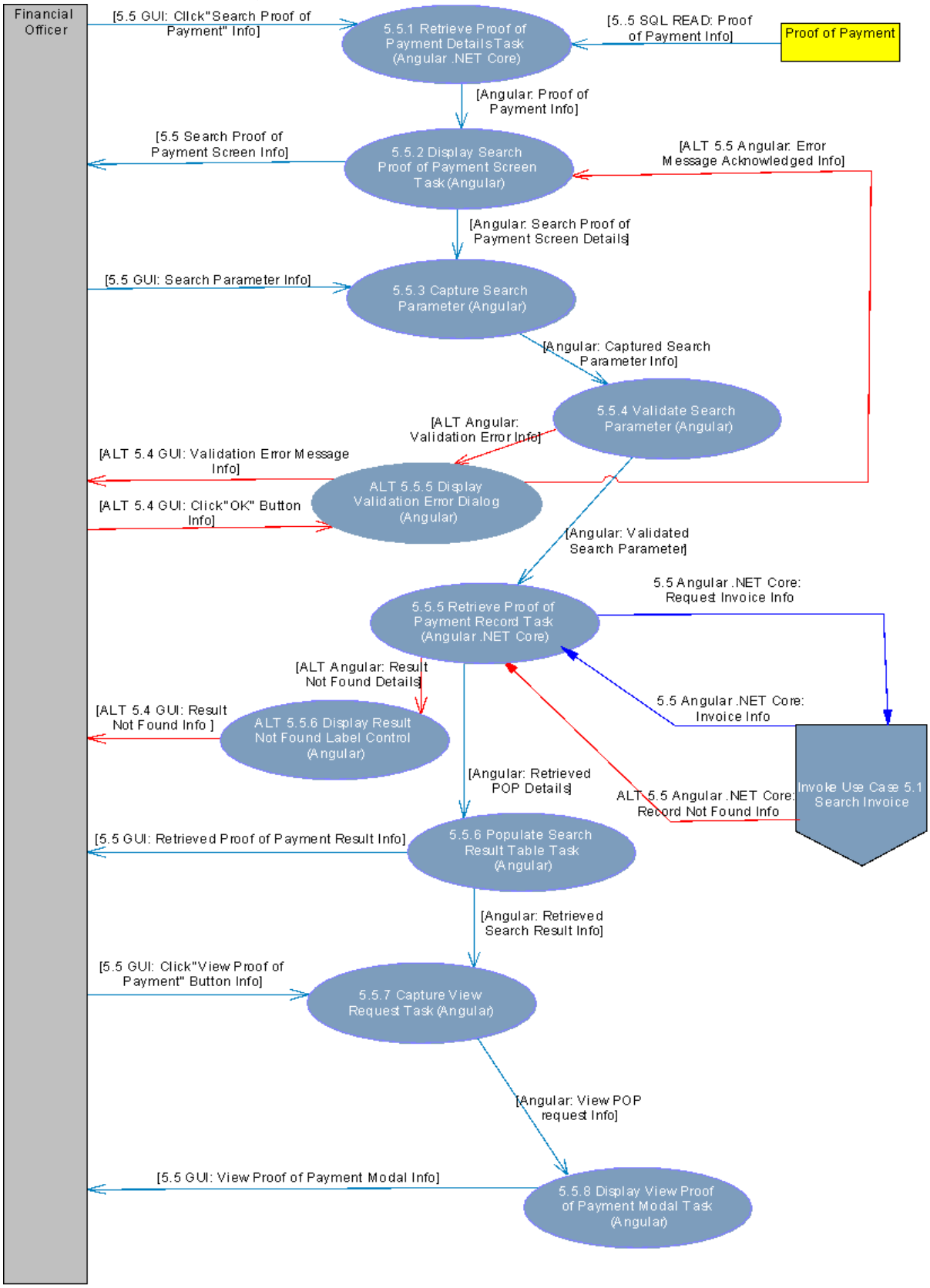


Figure 202: Primitive level data diagram5.5 Search Proof of Payment

2.3.3.5.6 Search Invoice Type

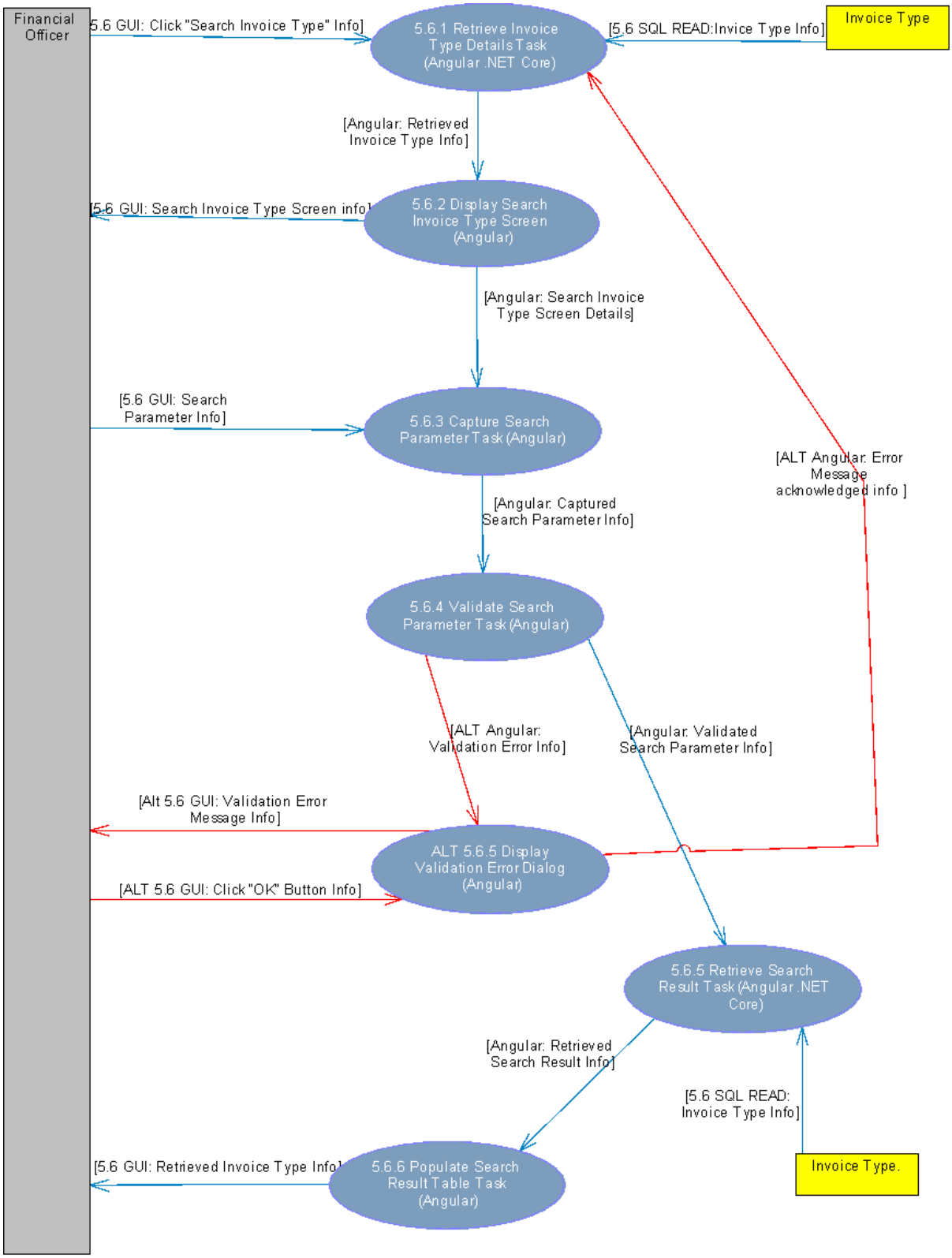


Figure 203: Primitive level data diagram5.6 Search Invoice Type

2.3.3.5.7 Create Invoice Type

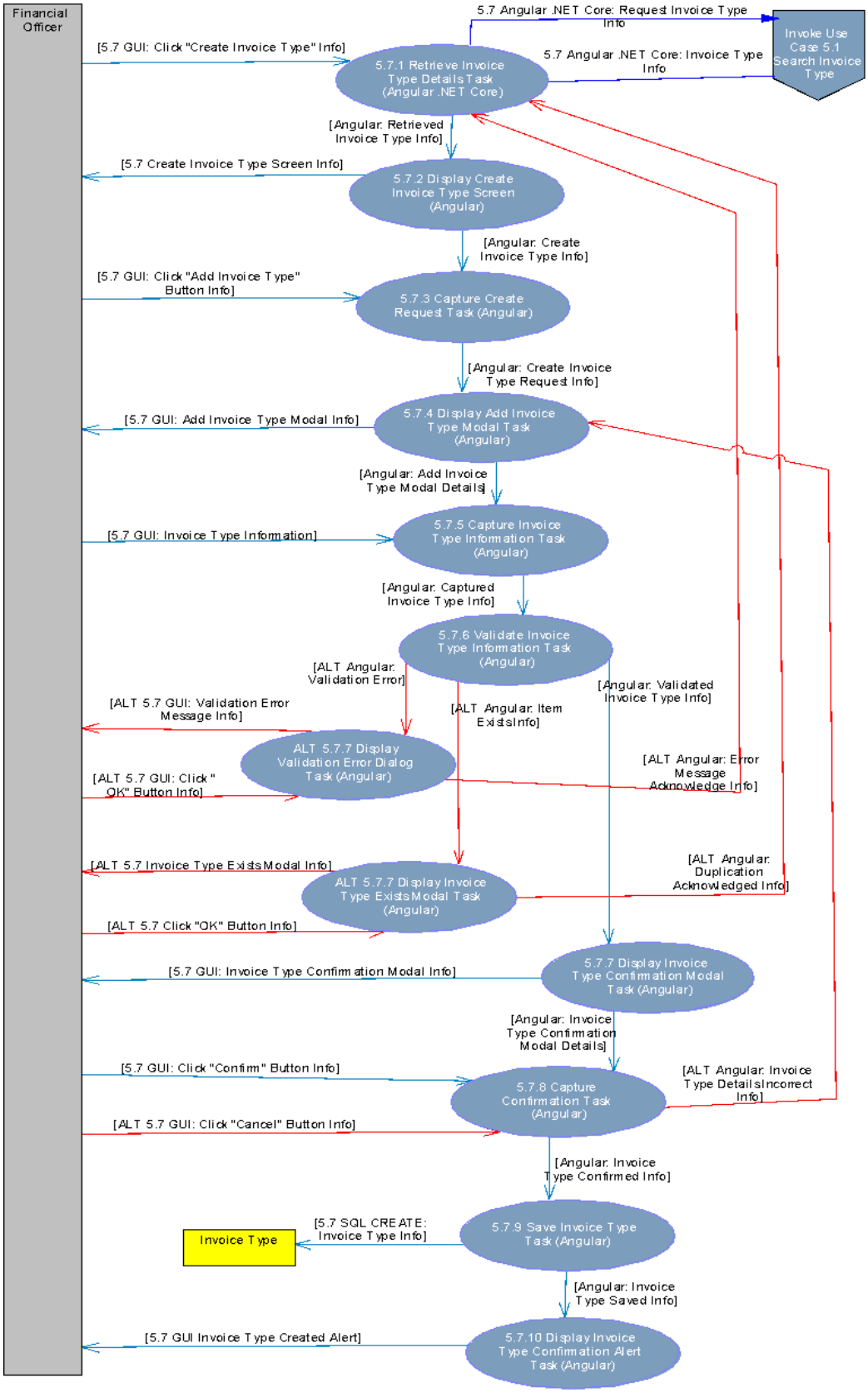


Figure 204: Primitive level data diagram5.7 Create Invoice Type

2.3.3.6. Human Resource Subsystem

2.3.3.6.1 Add employee

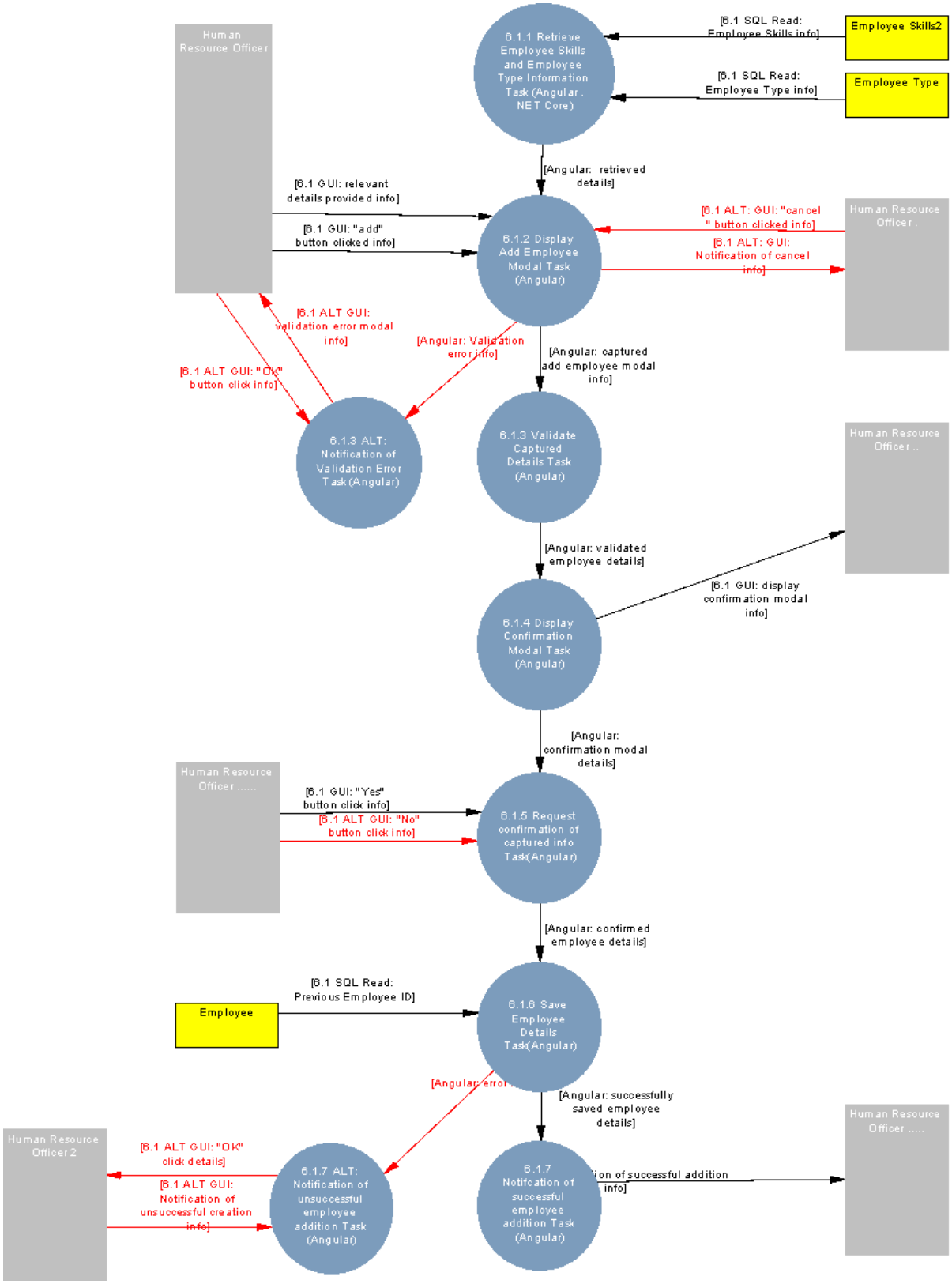


Figure 205: Primitive level data diagram6.1 Add employee

2.3.3.6.2 Edit employee

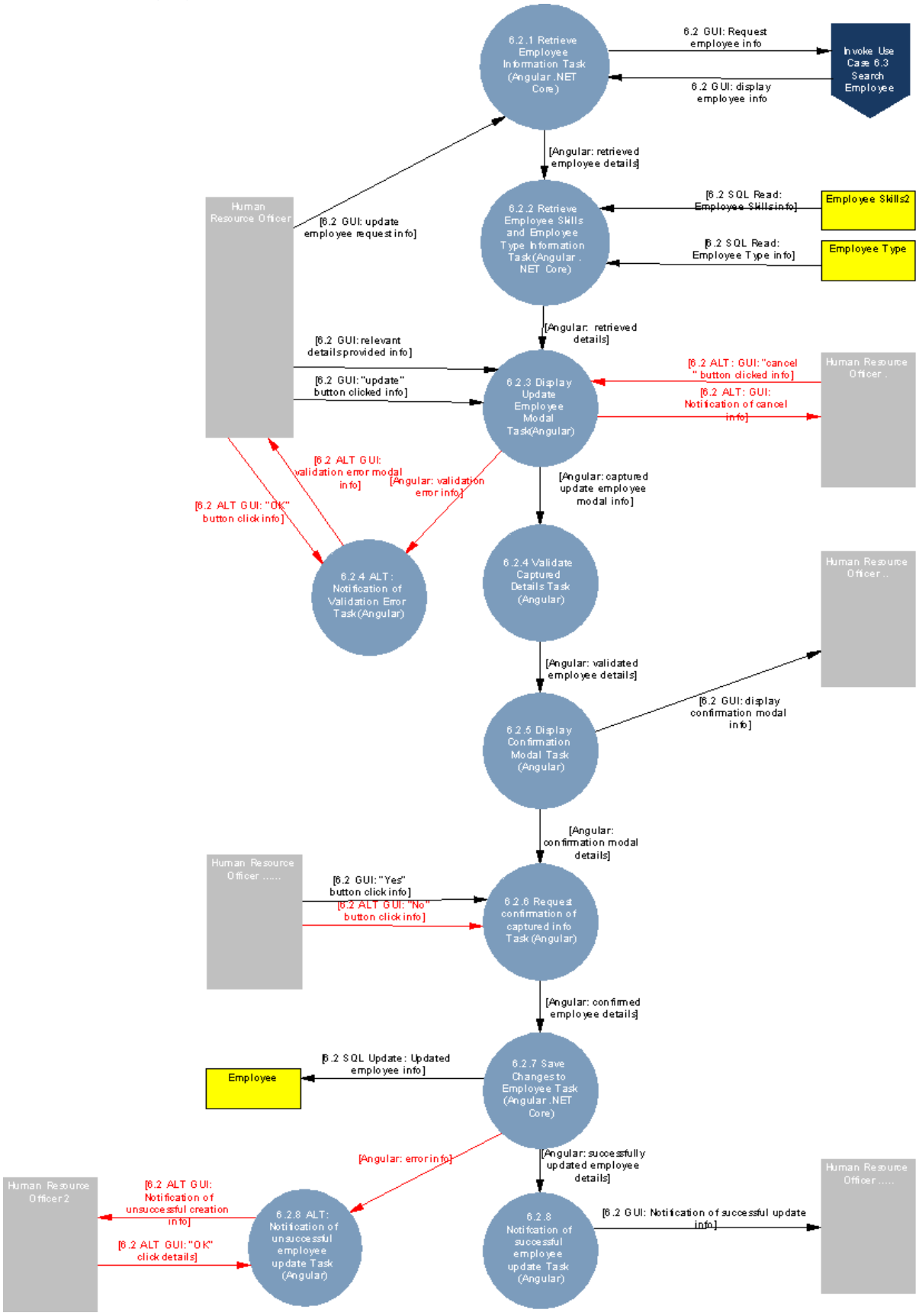


Figure 206: Primitive level data diagram6.2 Edit employee

2.3.3.6.3 Search employee

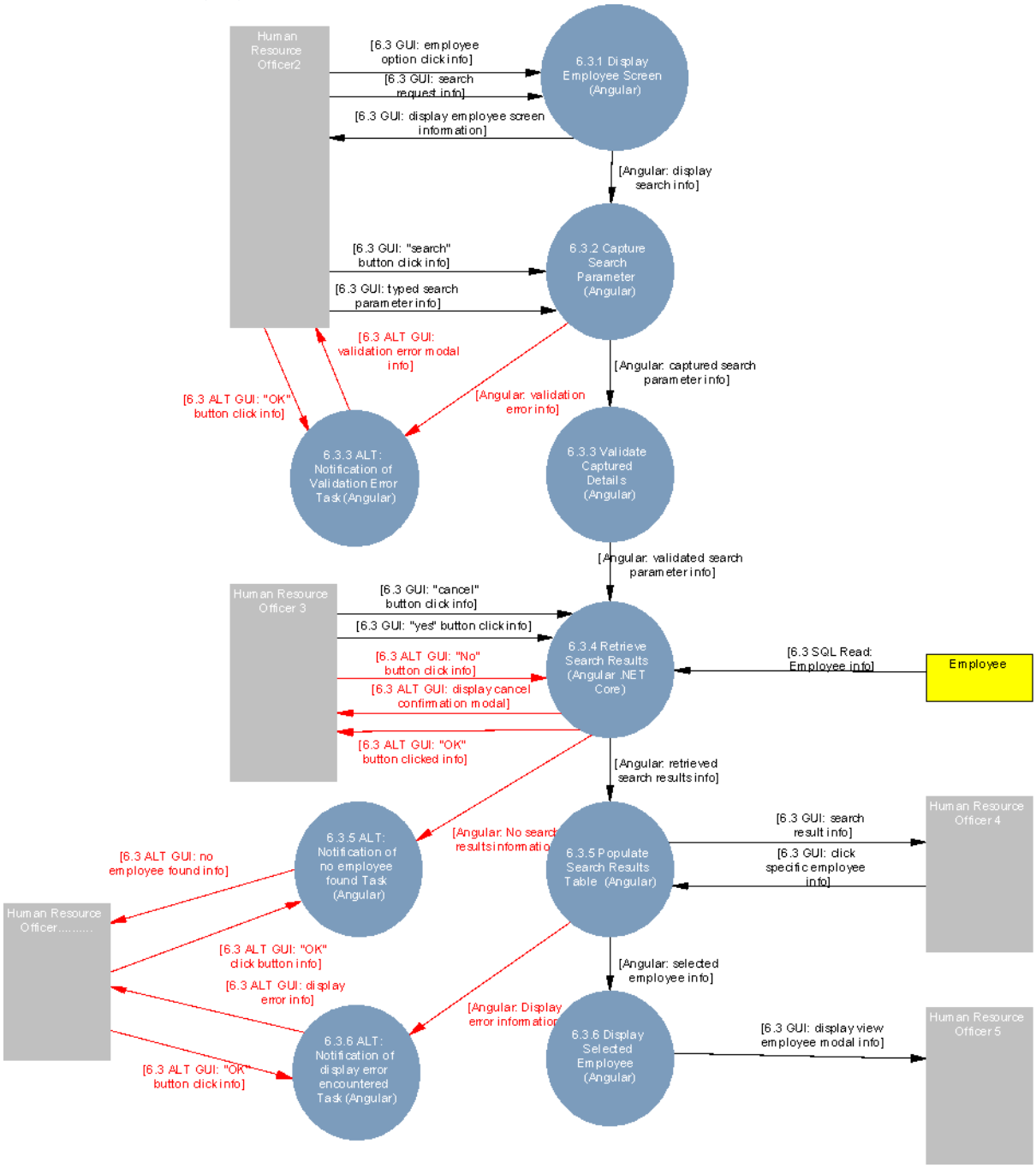


Figure 207: Primitive level data diagram6.3 Search employee

2.3.3.6.4 Upload Employee Document

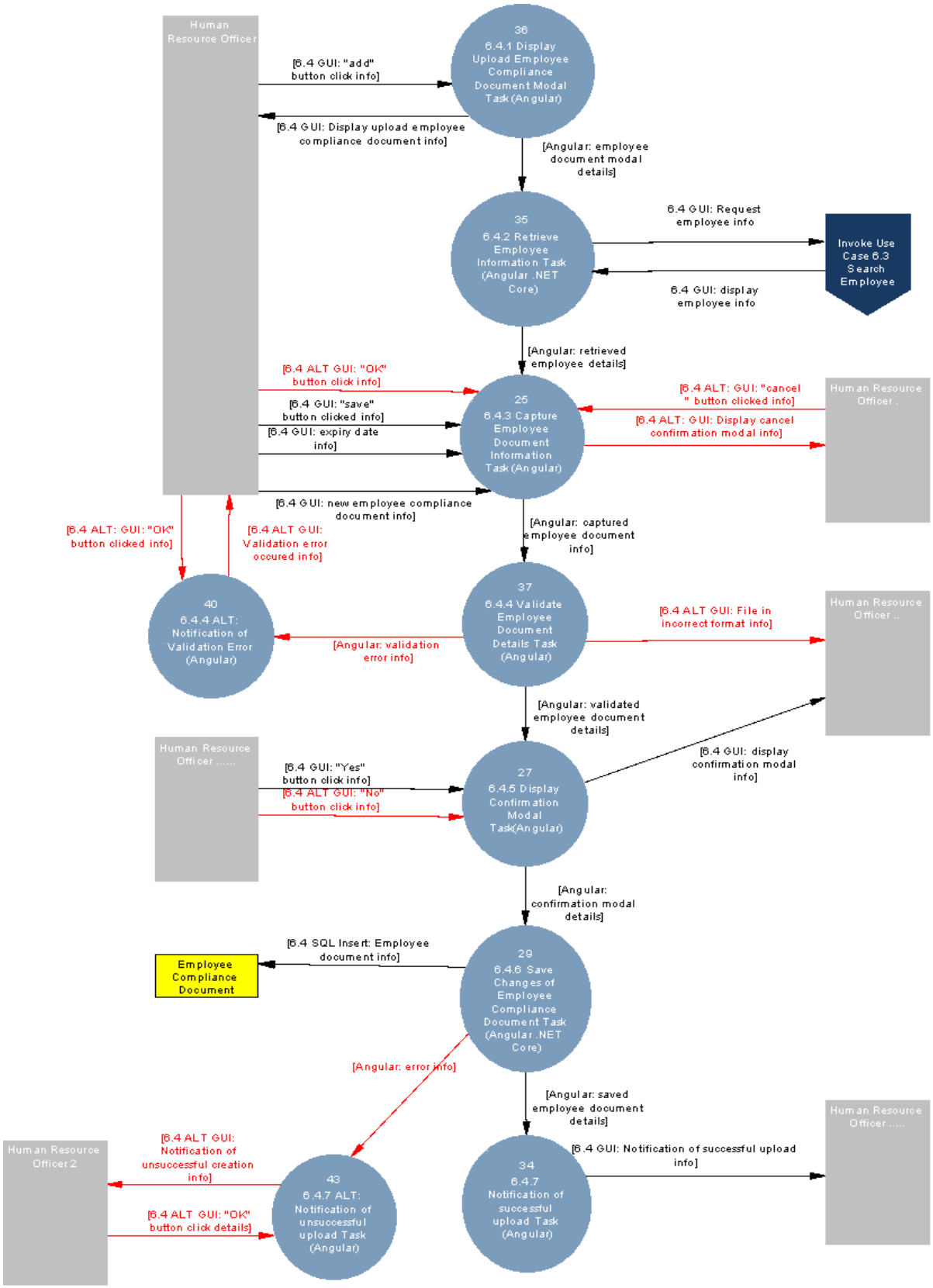


Figure 208: Primitive level data diagram6.4 Upload Employee Document

2.3.3.6.5 Edit Employee Document

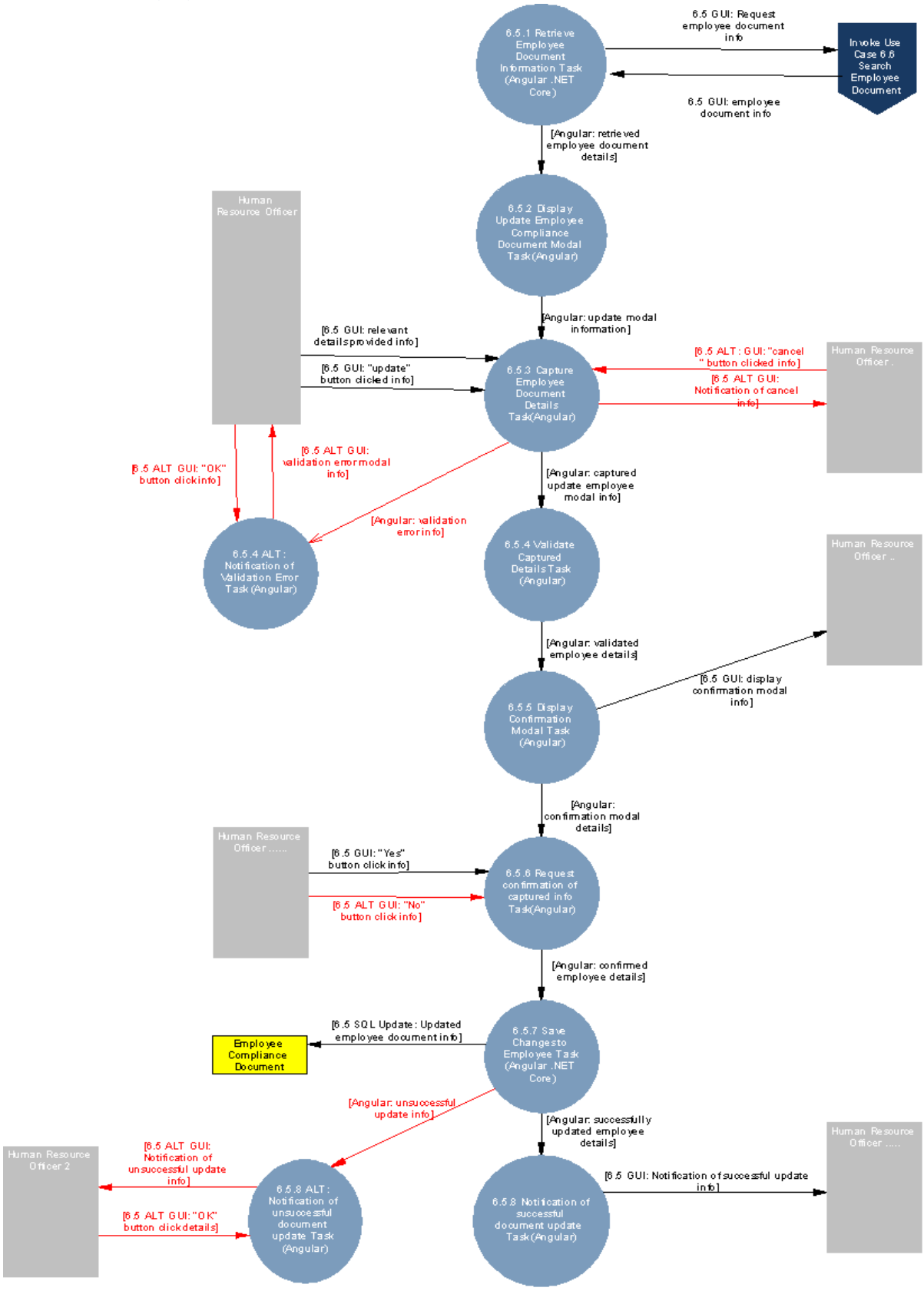


Figure 209: Primitive level data diagram6.5 Edit Employee Document

2.3.3.6.6 Search Employee document

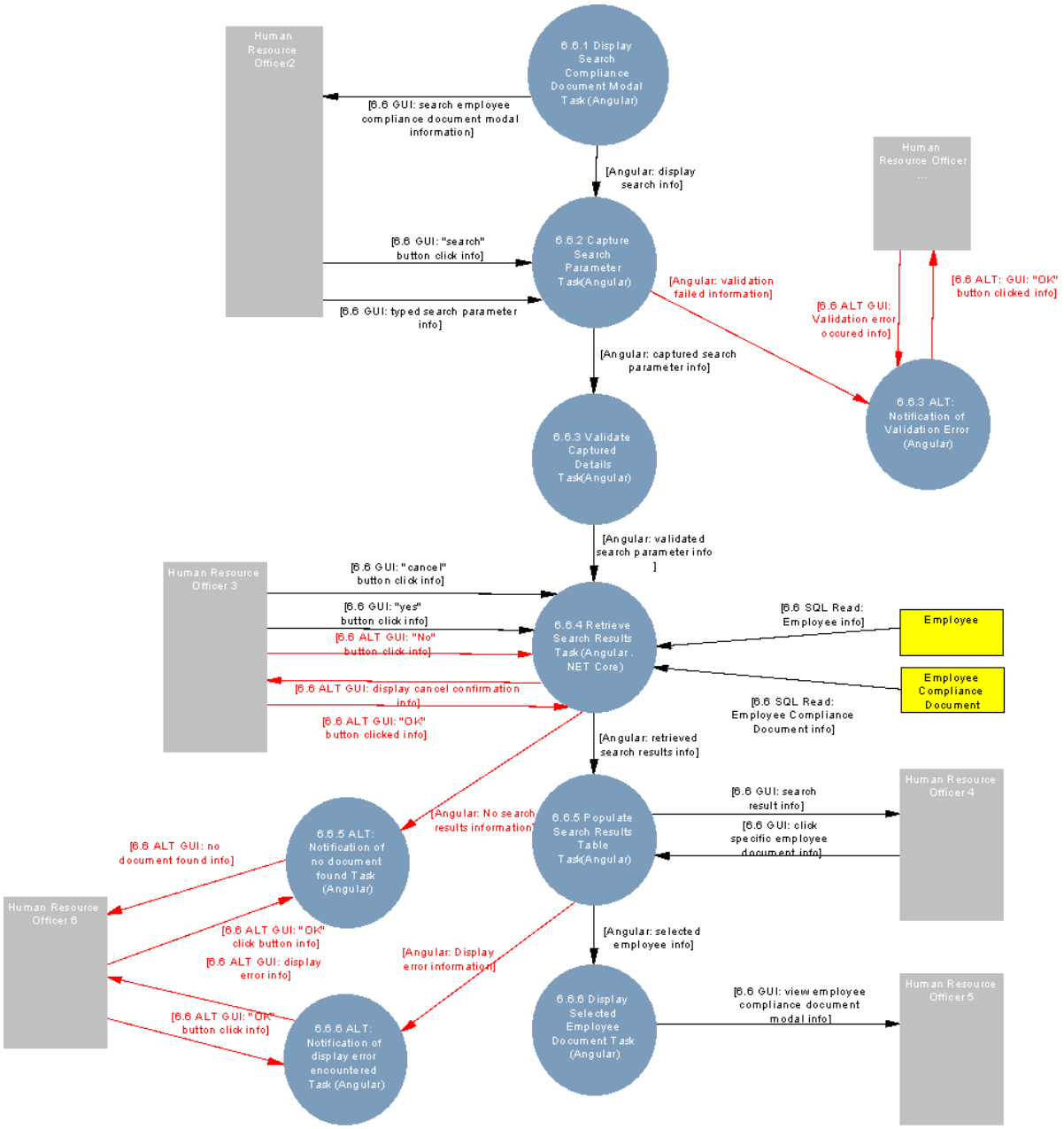


Figure 210: Primitive level data diagram6.6 Search Employee document

2.3.3.6.7 Create Account

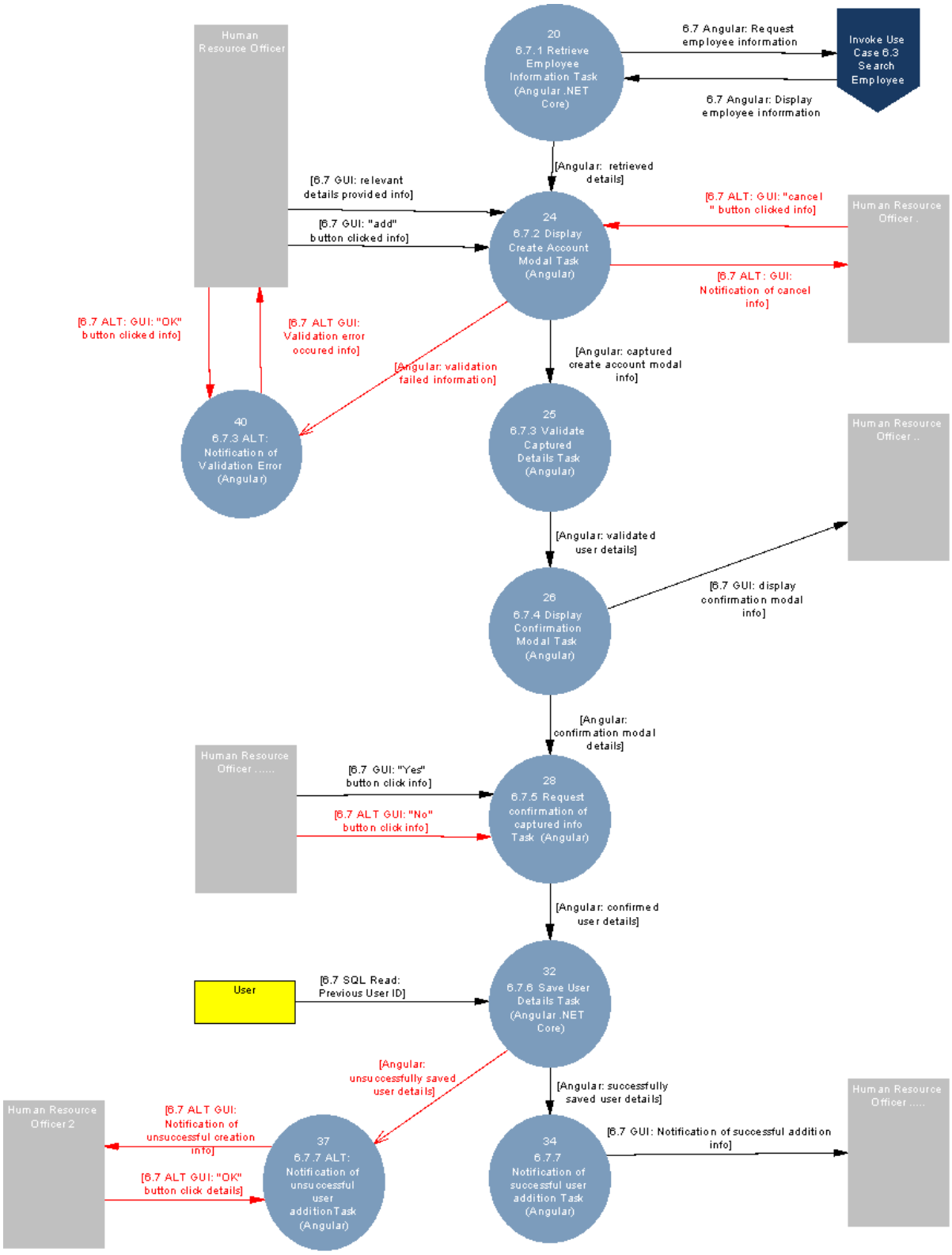


Figure 211: Primitive level data diagram6.7 Create Account

2.3.3.6.8 Edit Account

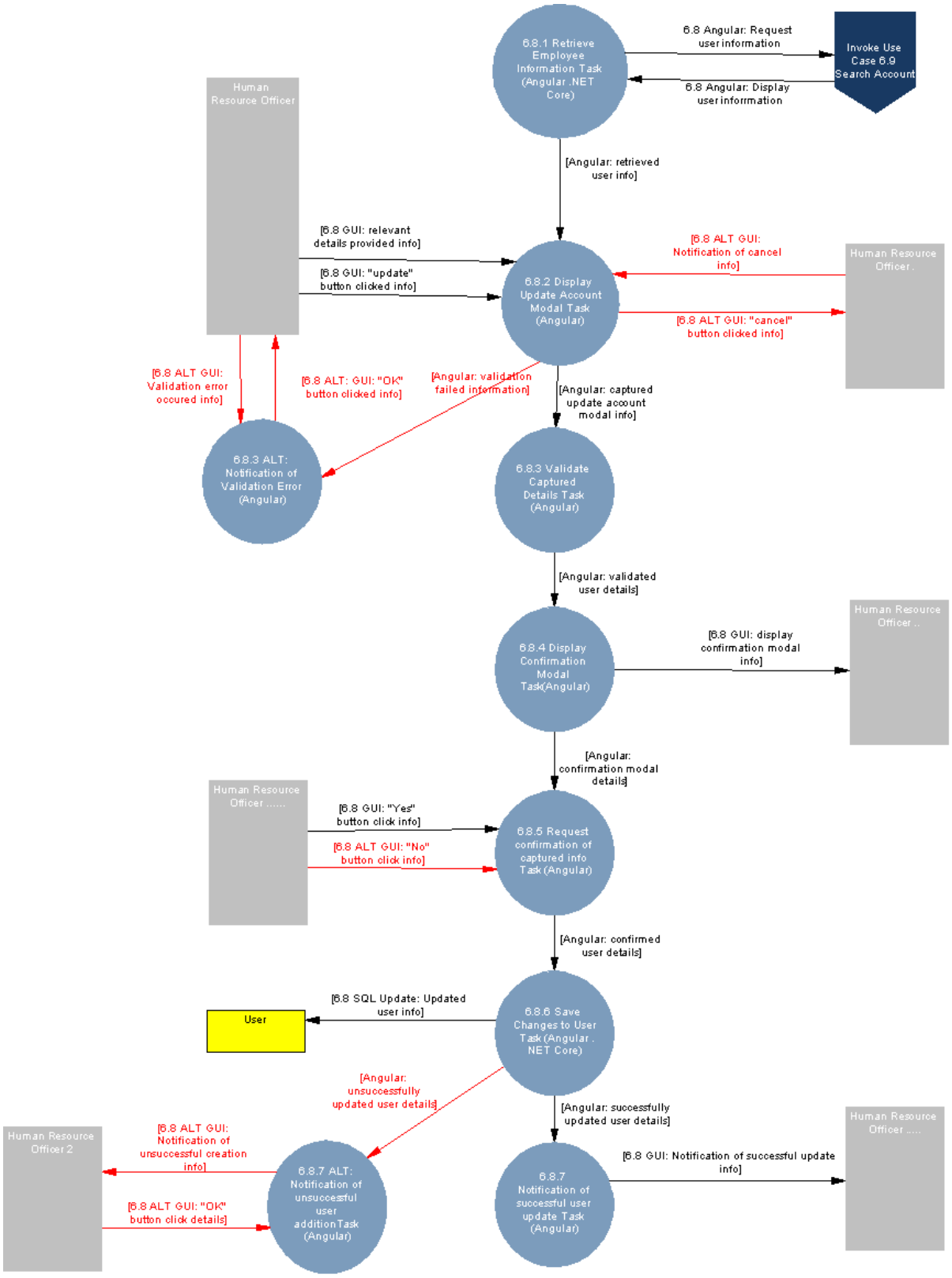


Figure 212: Primitive level data diagram6.8 Edit Account

2.3.3.6.9 Search Account

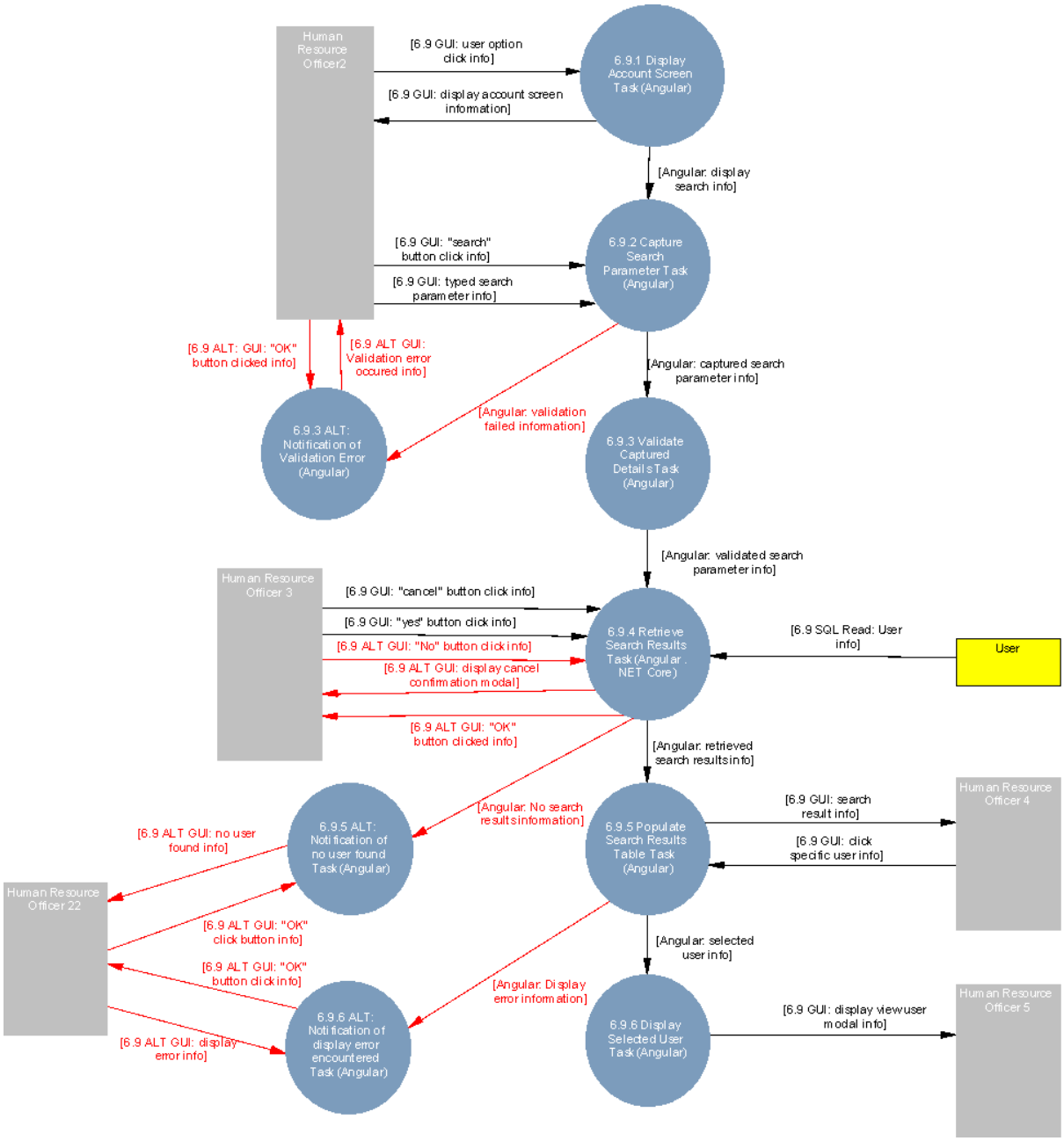


Figure 213: Primitive level data diagram6.9 Search Account

2.3.3.6.10 Create Employee Type

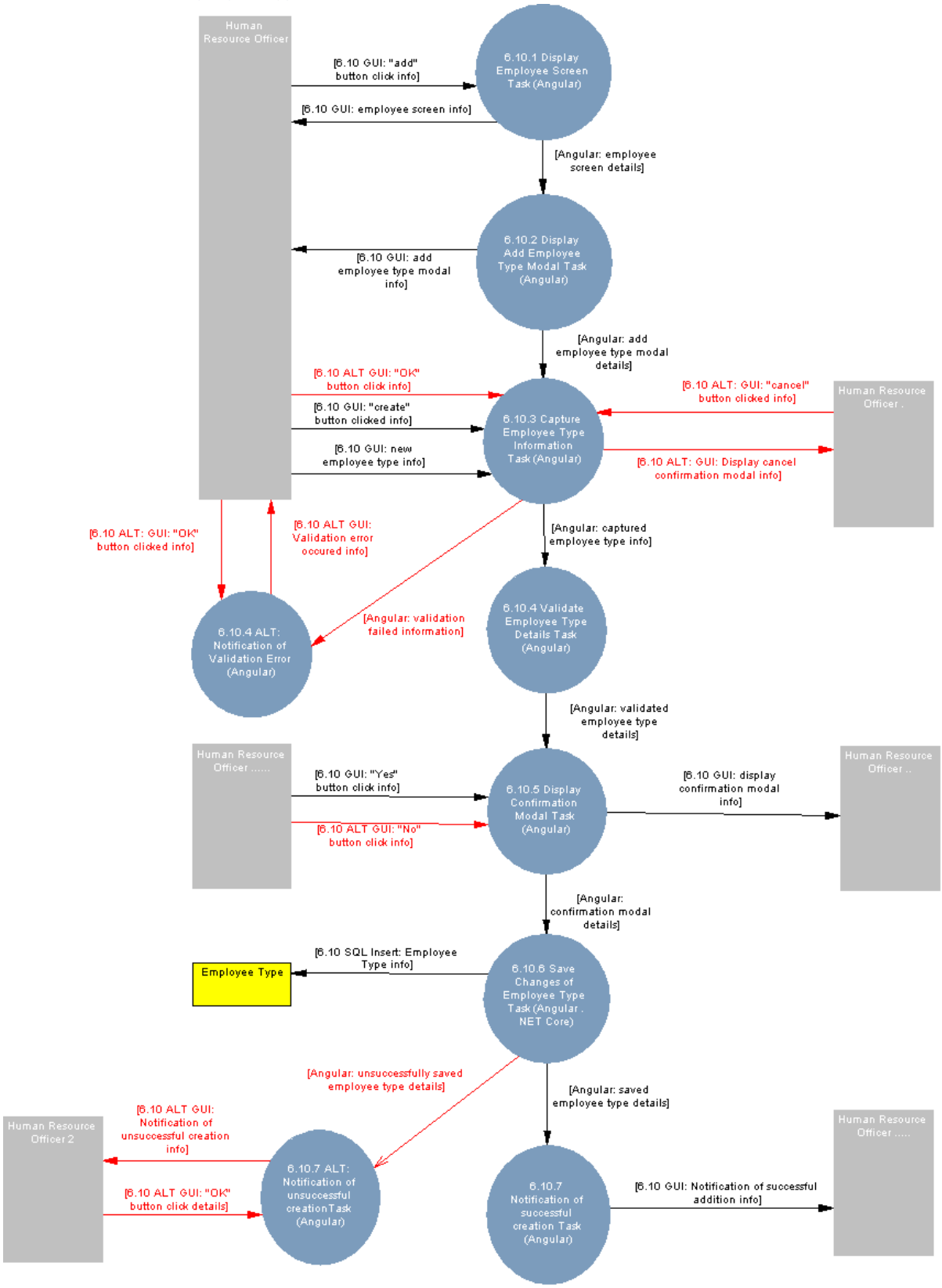


Figure 214: Primitive level data diagram6.10 Create Employee Type

2.3.3.6.11 Search Employee Type

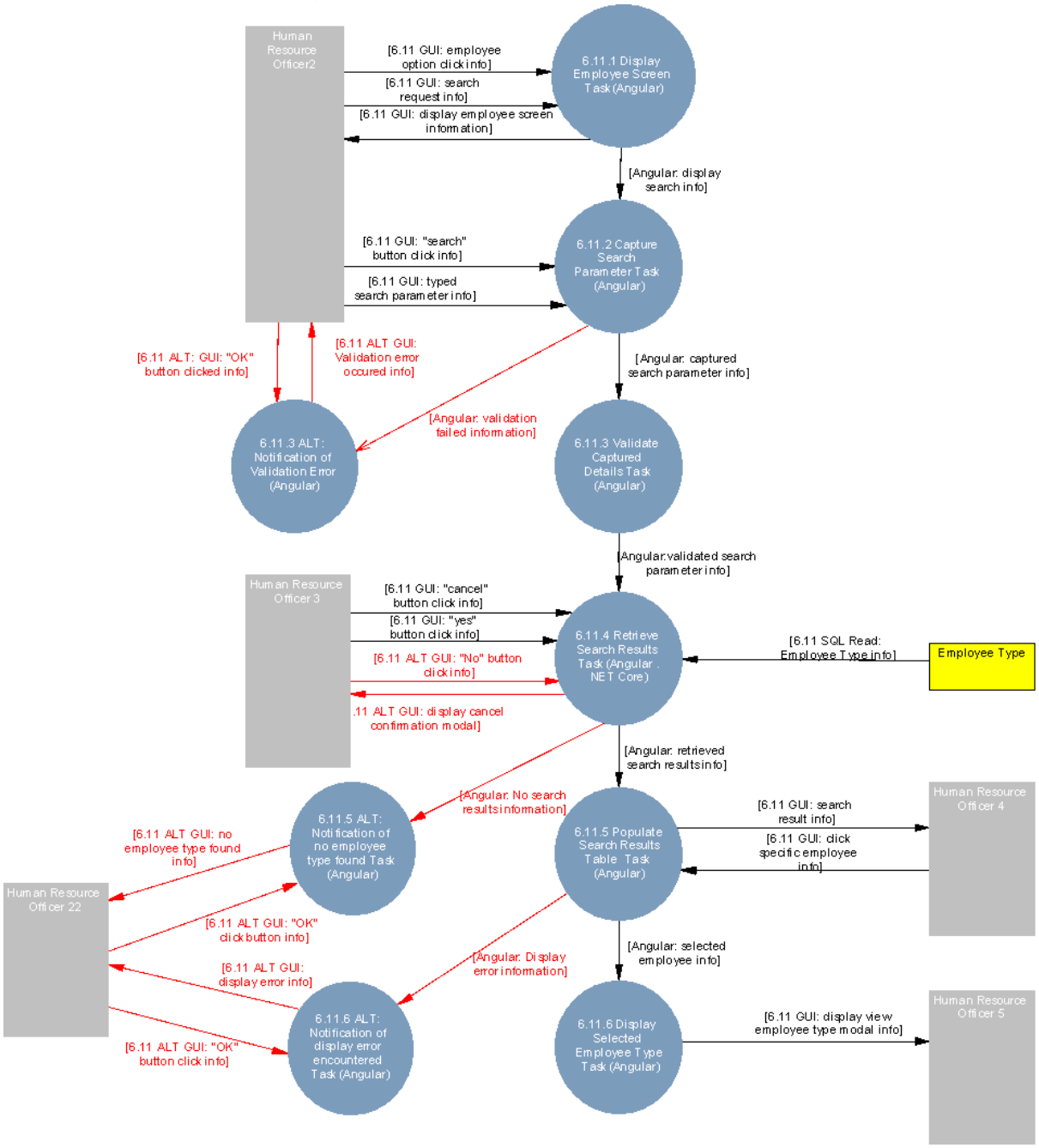


Figure 215: Primitive level data diagram6.11 Search Employee Type

2.3.3.6.12 Create Employee Skill

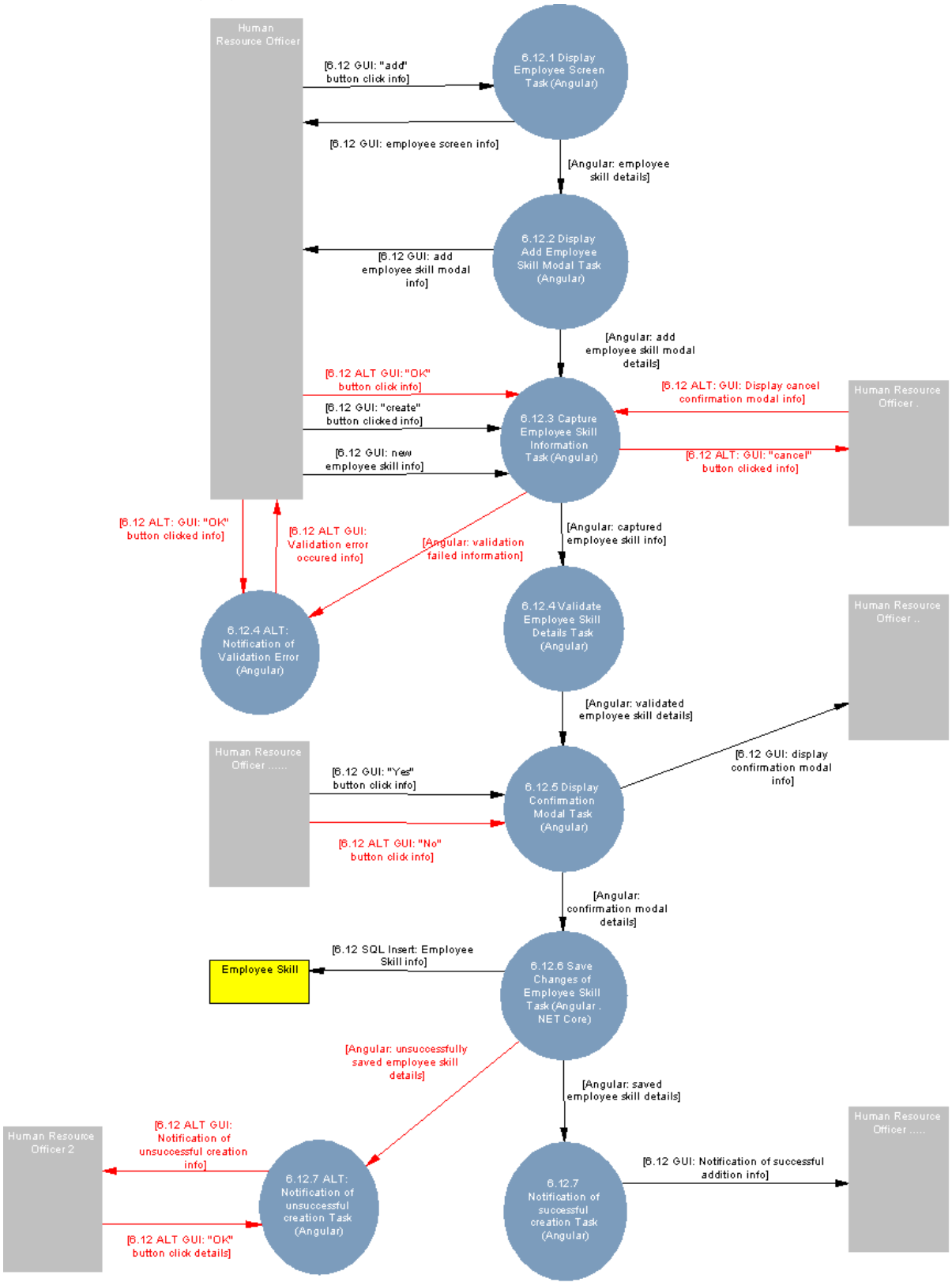


Figure 216: Primitive level data diagram6.12 Create Employee Skill

2.3.3.6.13 Search Employee Skill

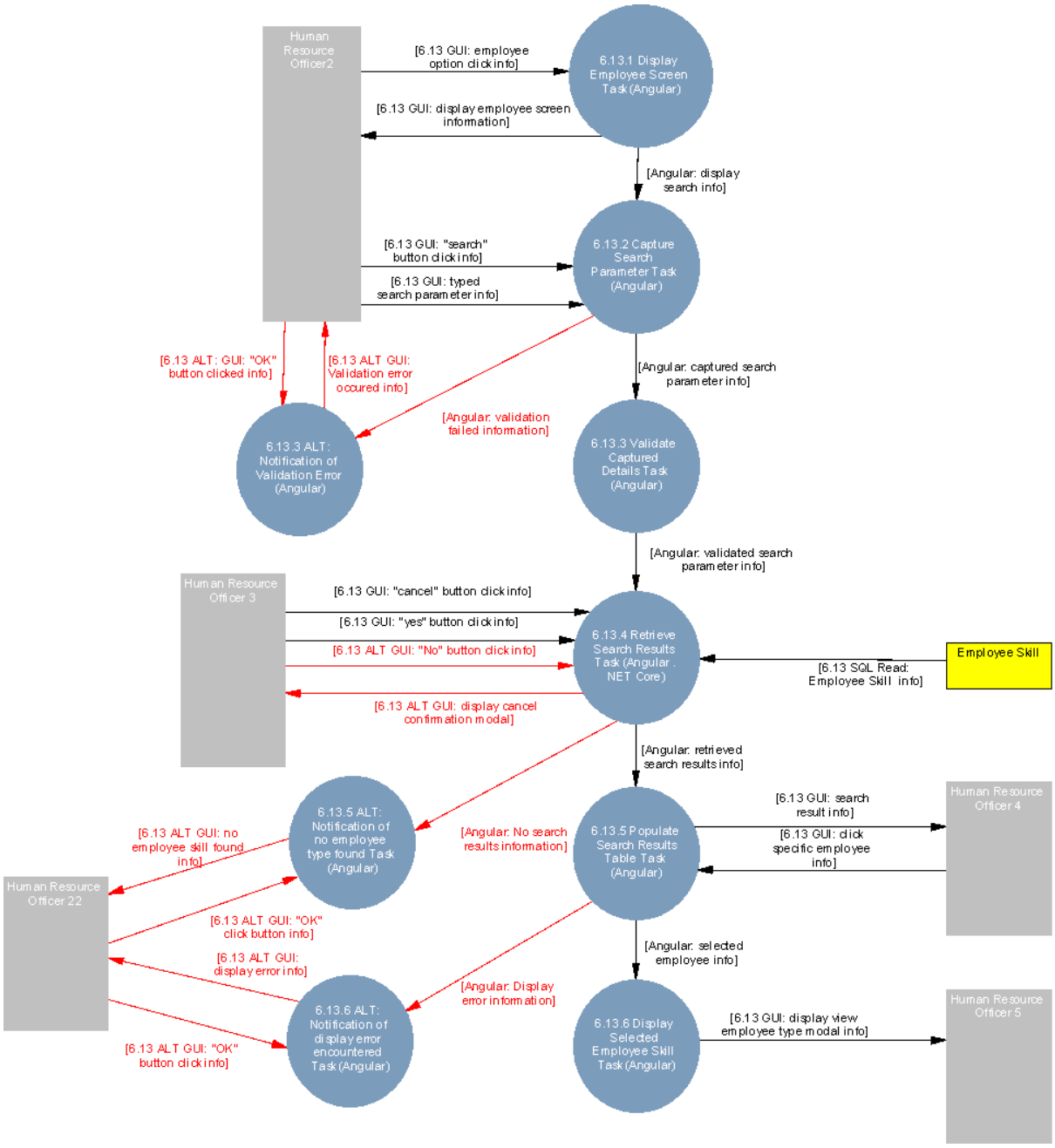


Figure 217: Primitive level data diagram6.13 Search Employee Skill

2.3.3.7. Team Subsystem

2.3.3.7.1 Create Team

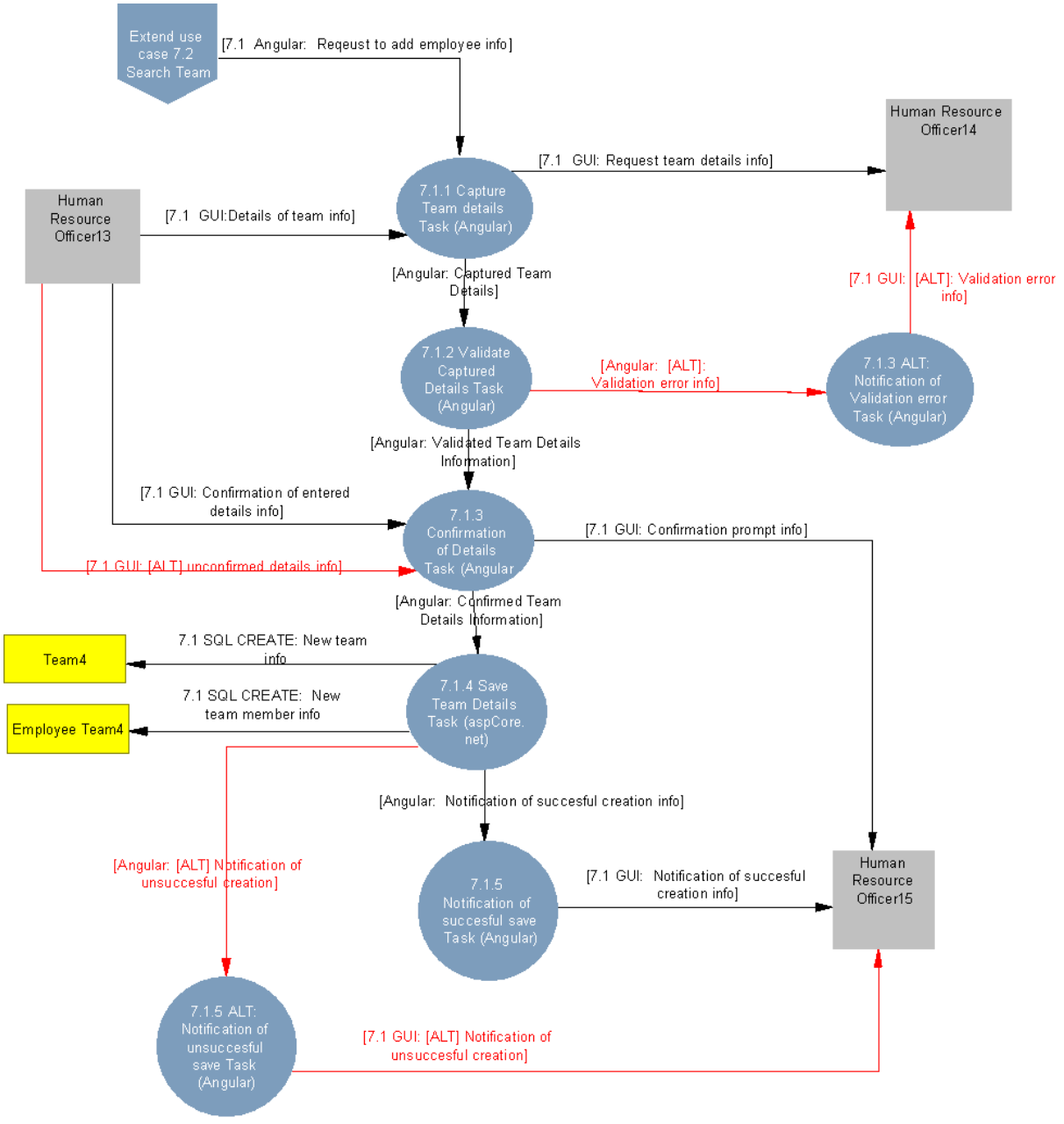


Figure 218: Primitive level data diagram7.1 Create Team

2.3.3.7.2 Search Team

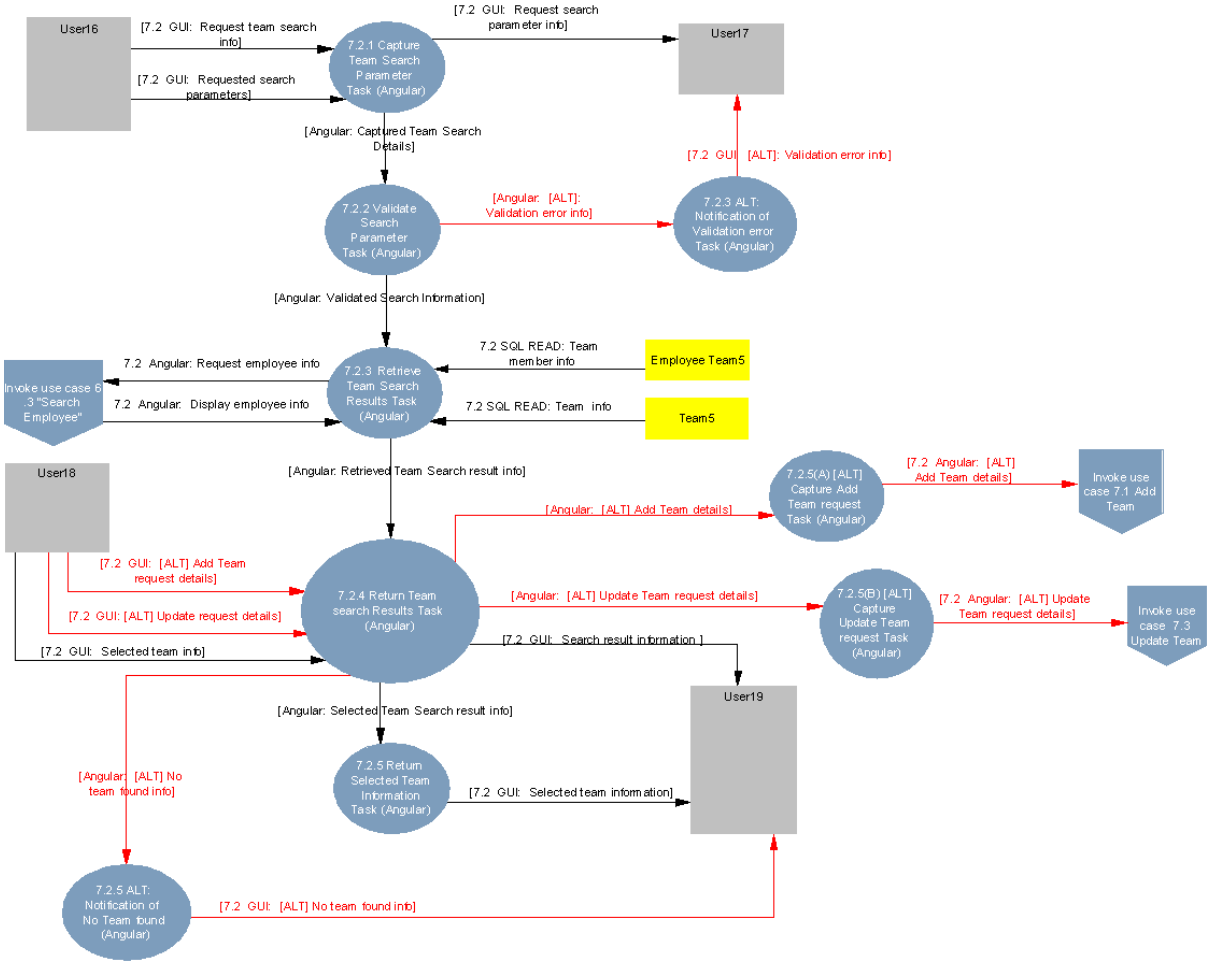


Figure 219: Primitive level data diagram7.2 Search Team

2.3.3.7.3 Edit Team

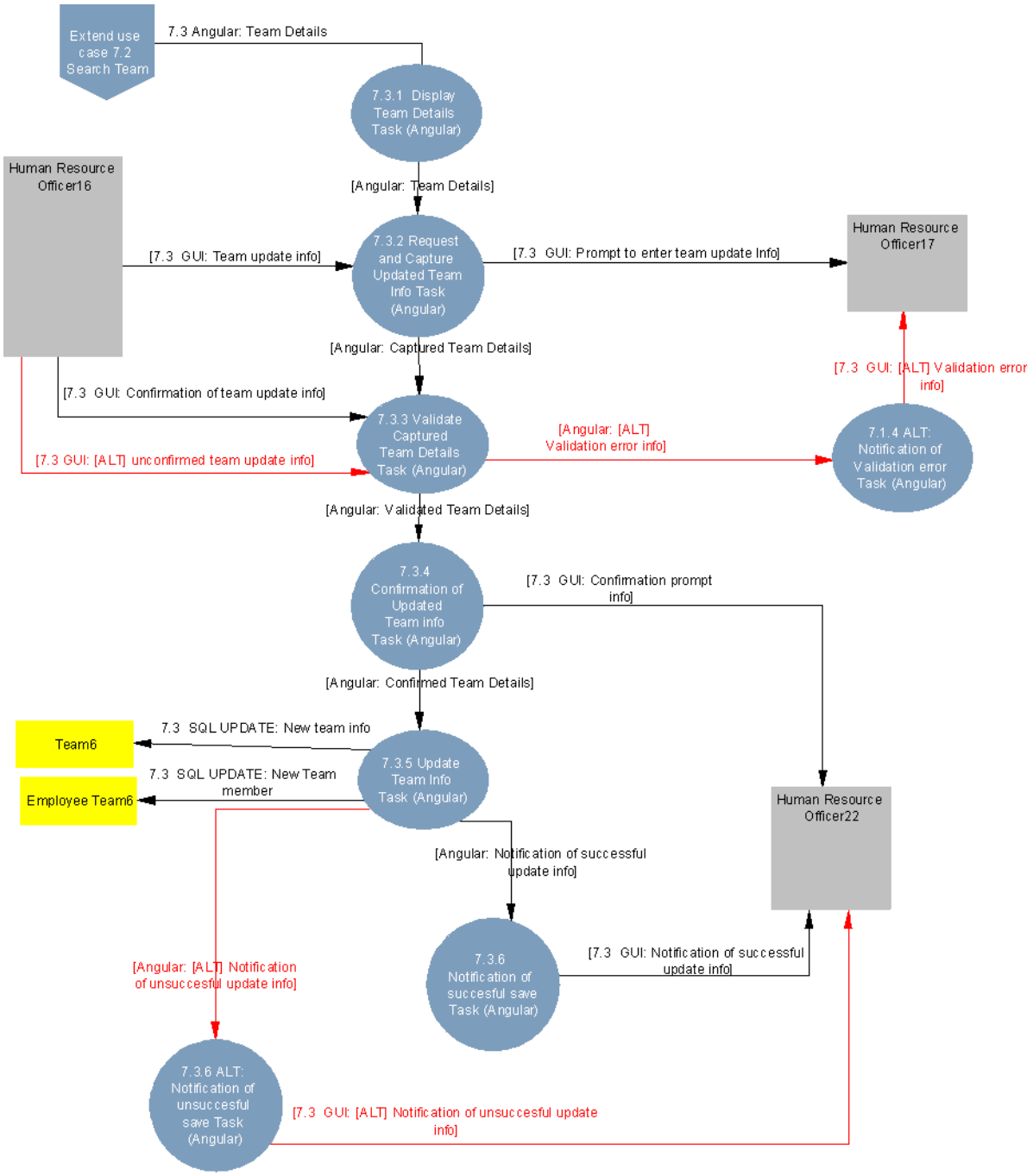


Figure 220: Primitive level data diagram7.3 Edit Team

2.3.3.8. Invoicing Subsystem

2.3.3.8.1 Search Project Line-Item List

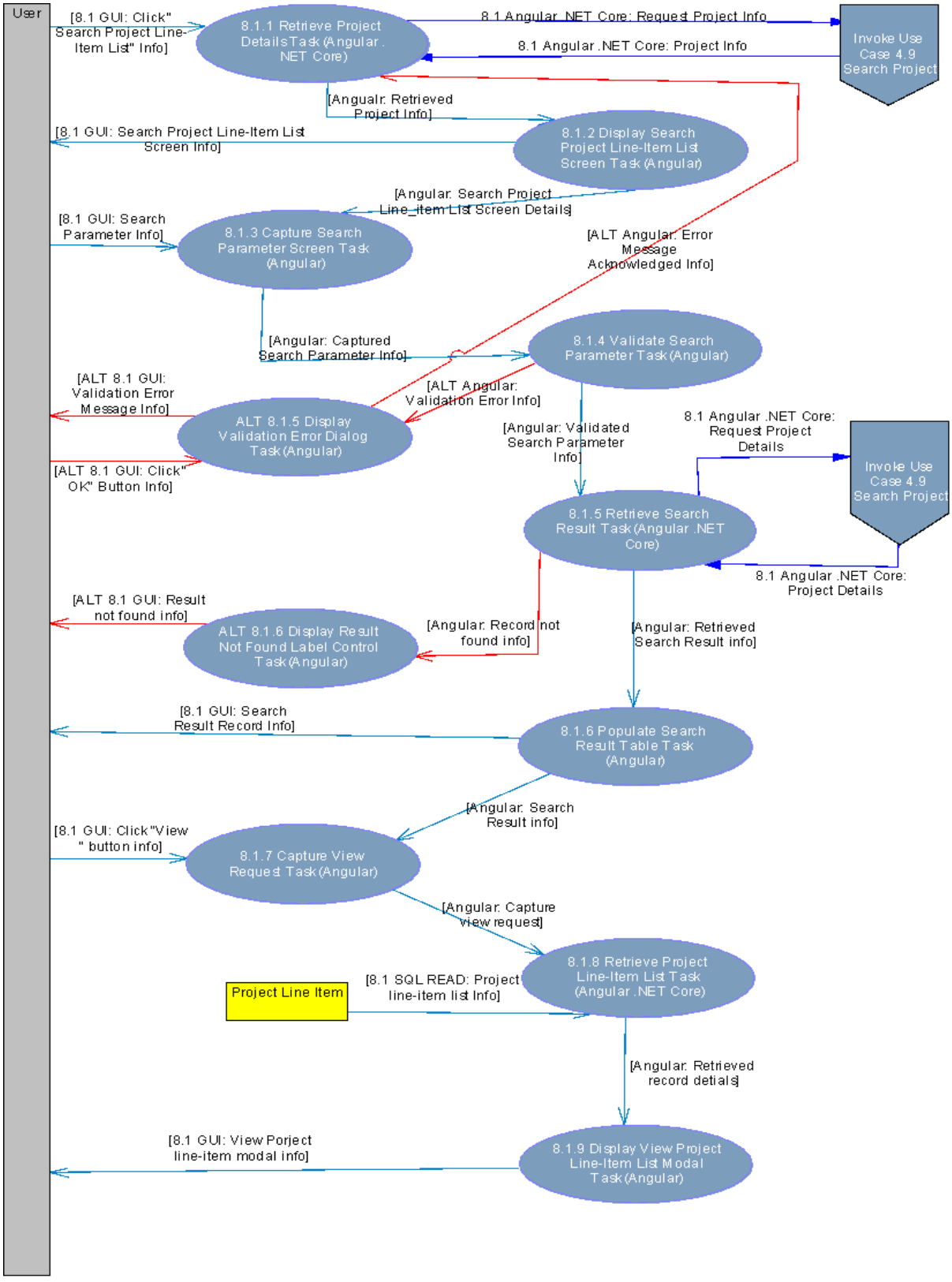


Figure 221: Primitive level data diagram8.1 Search Project Line-Item List

2.3.3.8.2 Edit Project Line-Item List

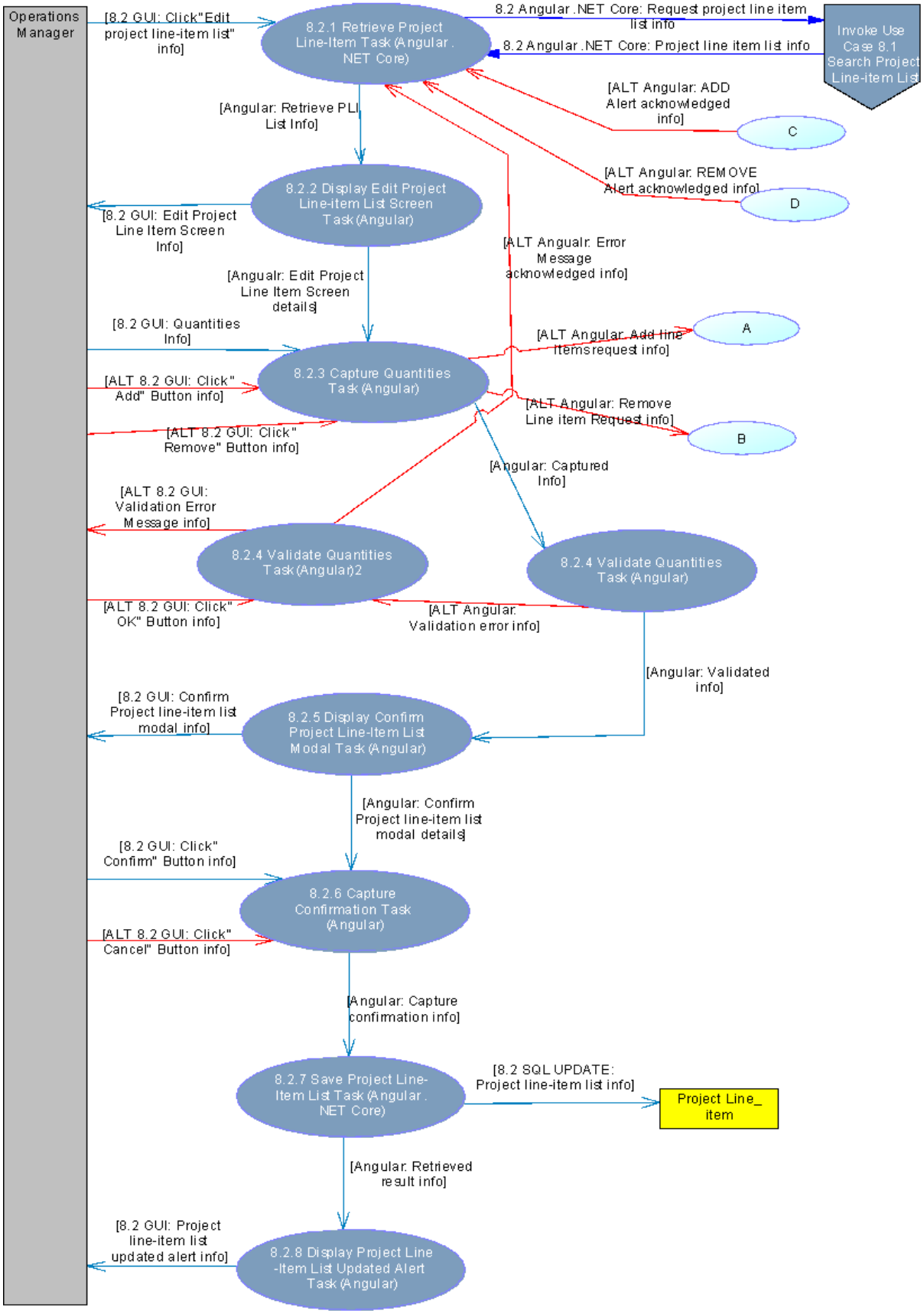


Figure 222: Primitive level data diagram8.2 Edit Project Line-Item List part1

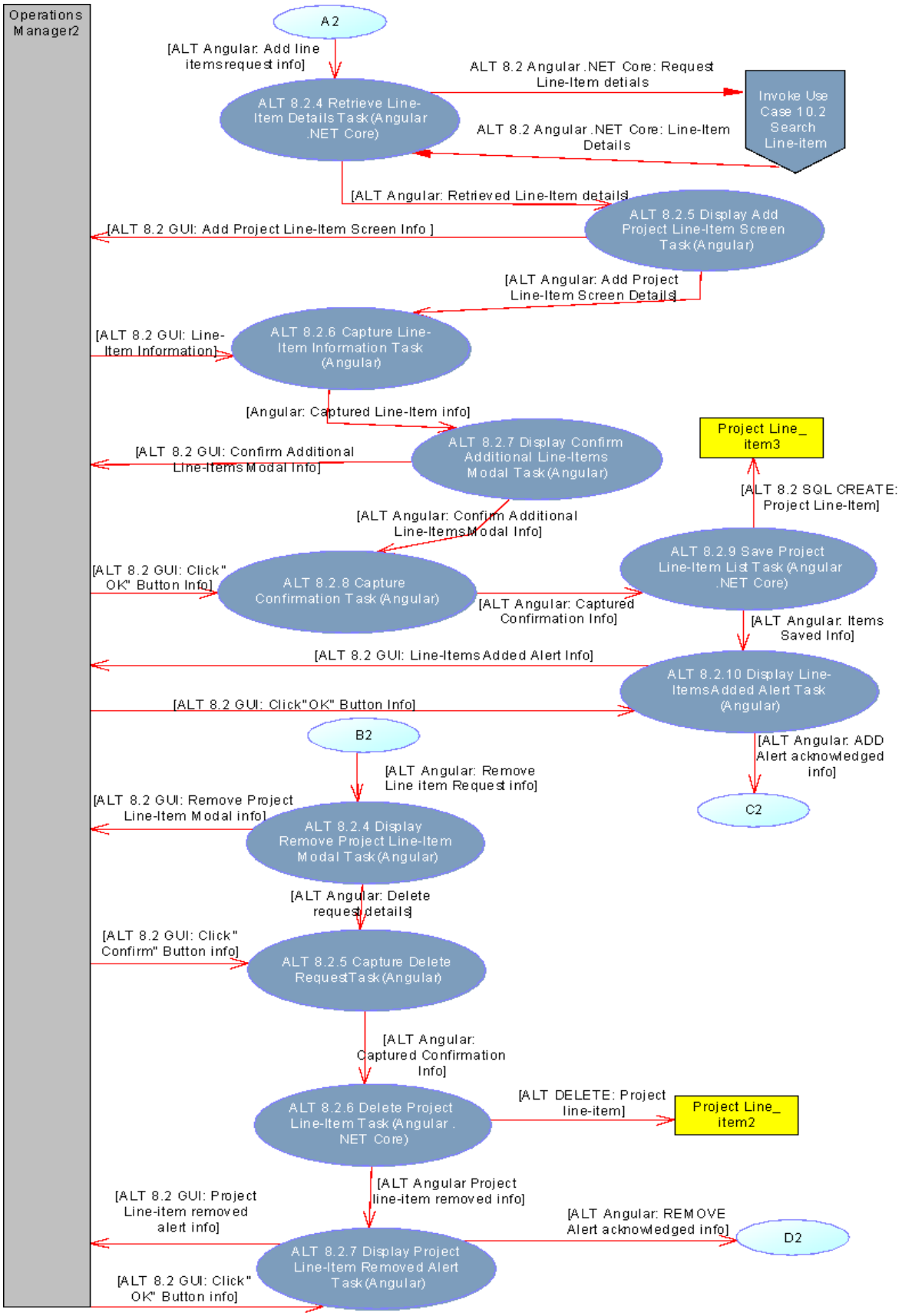


Figure 223: Primitive level data diagram8.2 Edit Project Line-Item List part2

2.3.3.8.3 Create Supplier Purchase Order

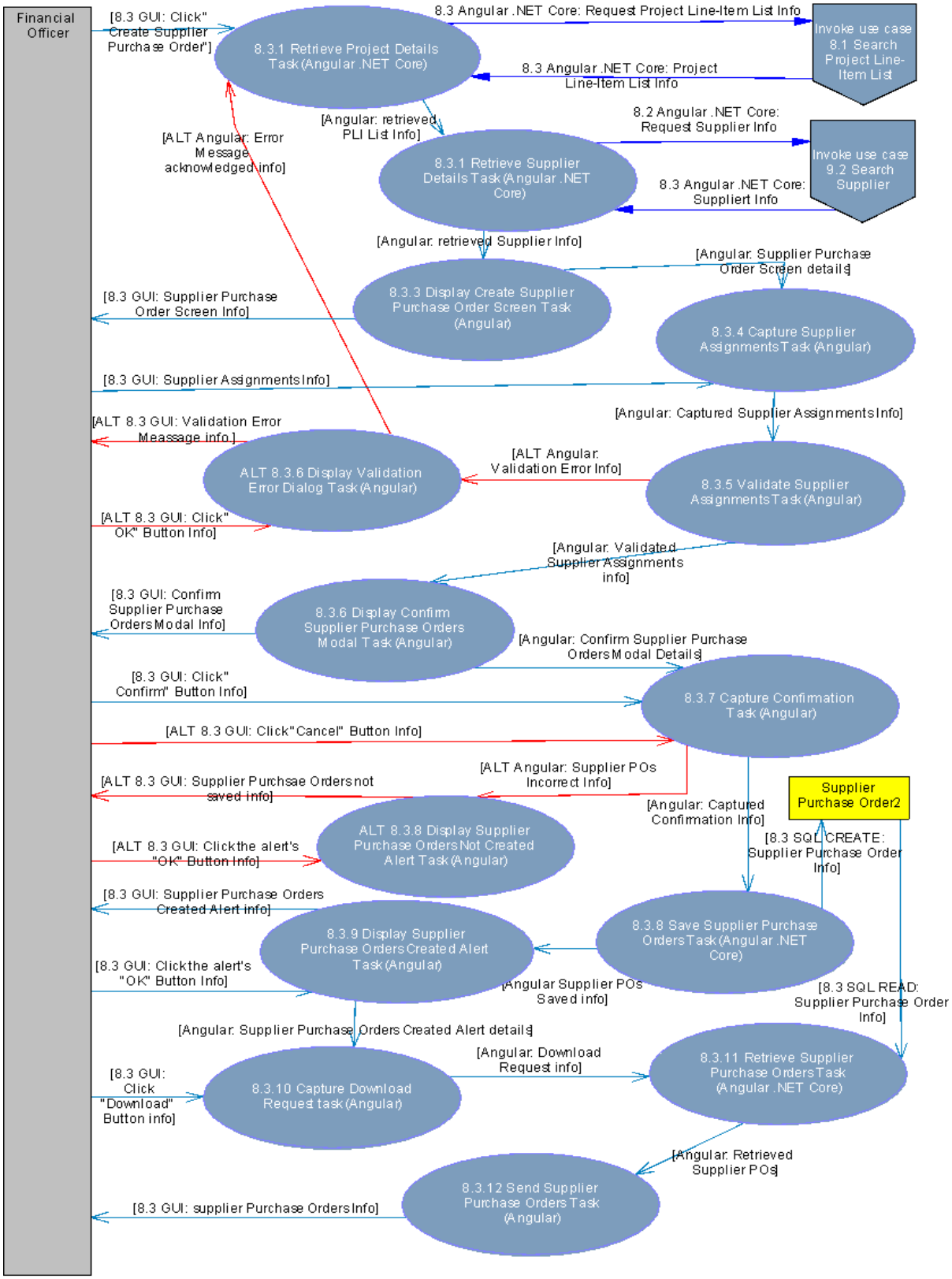


Figure 224: Primitive level data diagram8.3 Create Supplier Purchase Order

2.3.3.8.4 Capture Supplier Quote

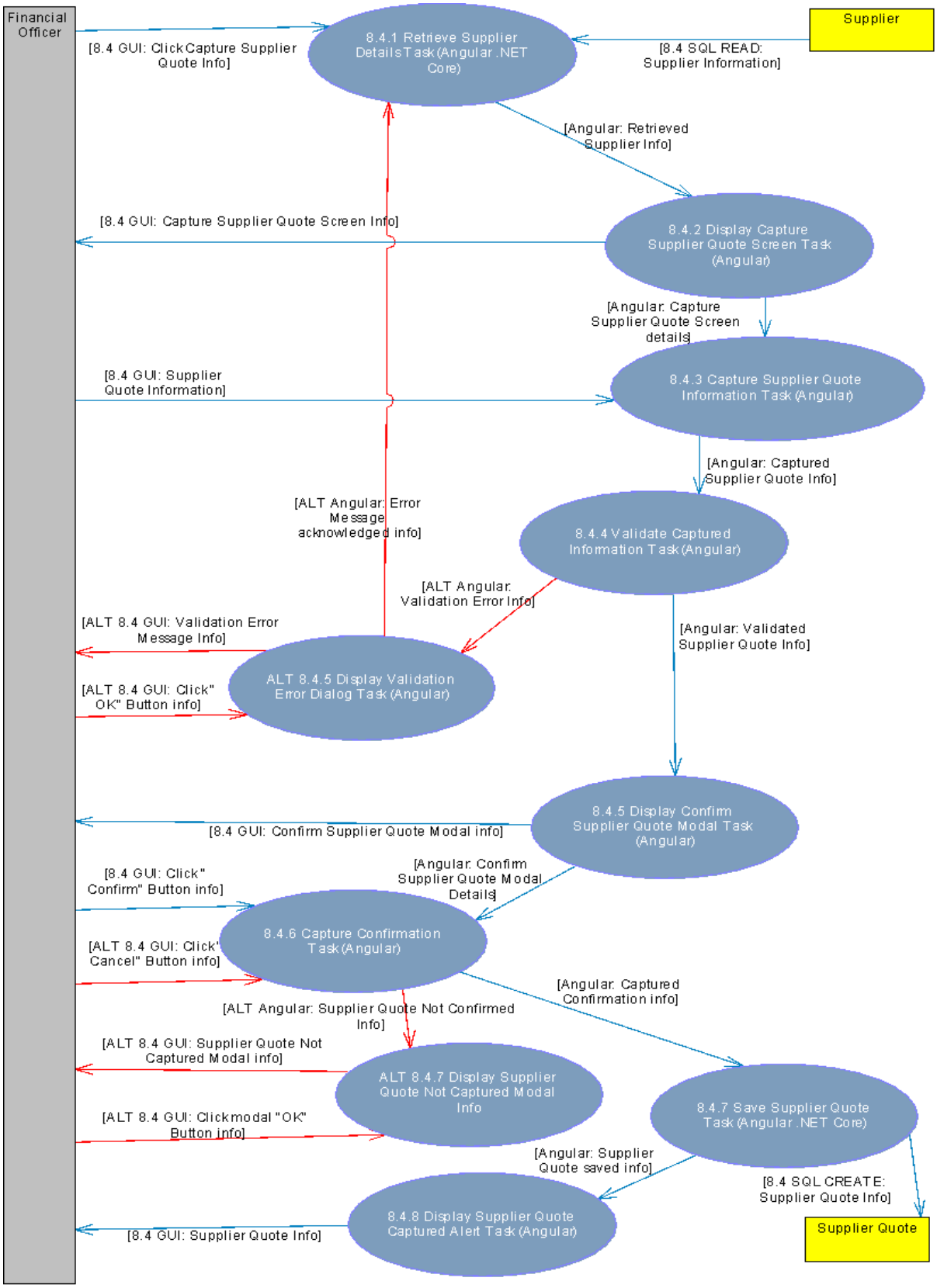


Figure 225: Primitive level data diagram8.4 Capture Supplier Quote

2.3.3.8.5 Search Supplier Quote

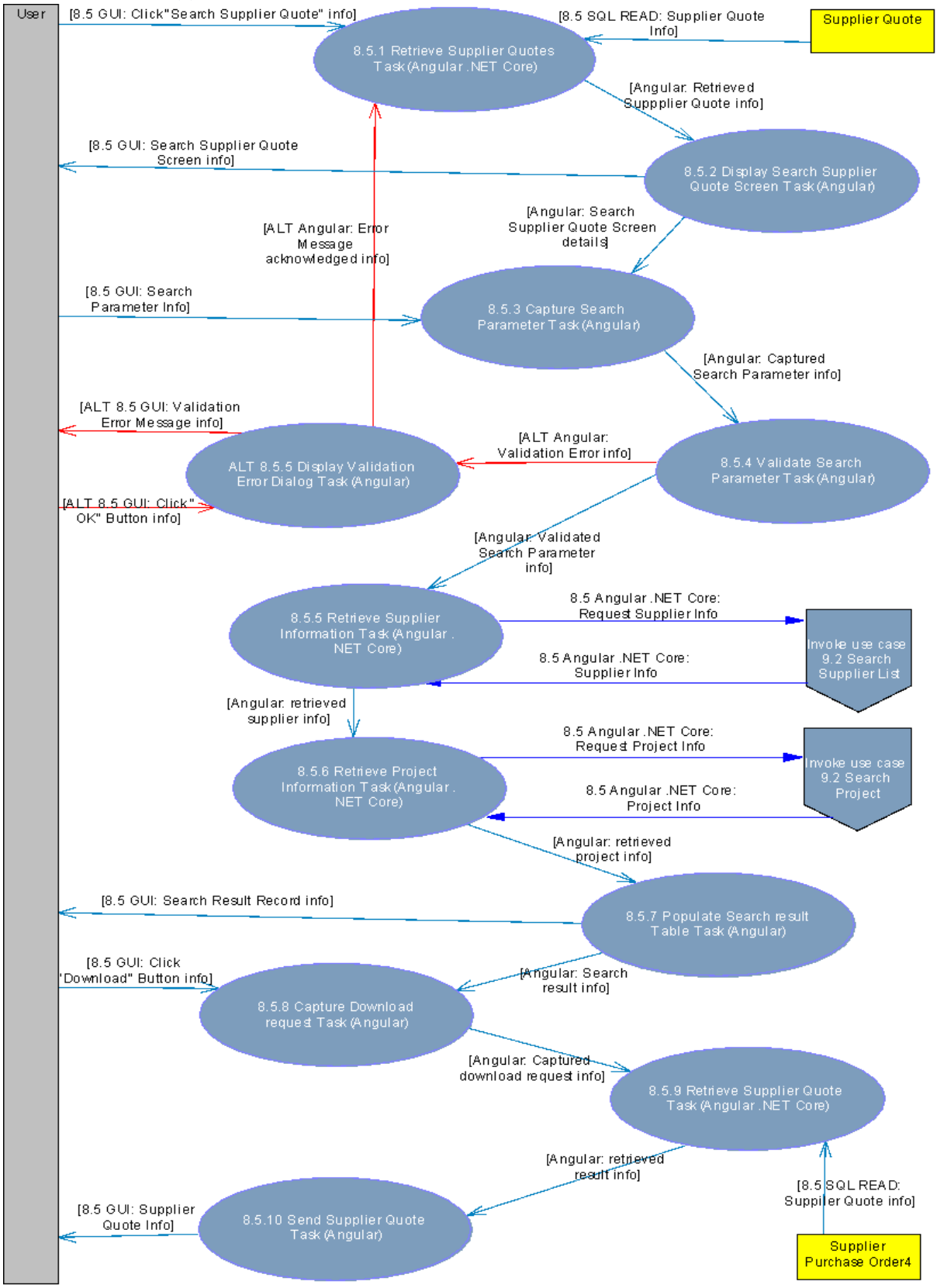


Figure 226: Primitive level data diagram8.5 Search Supplier Quote

2.3.3.8.6 Search Supplier Purchase Order

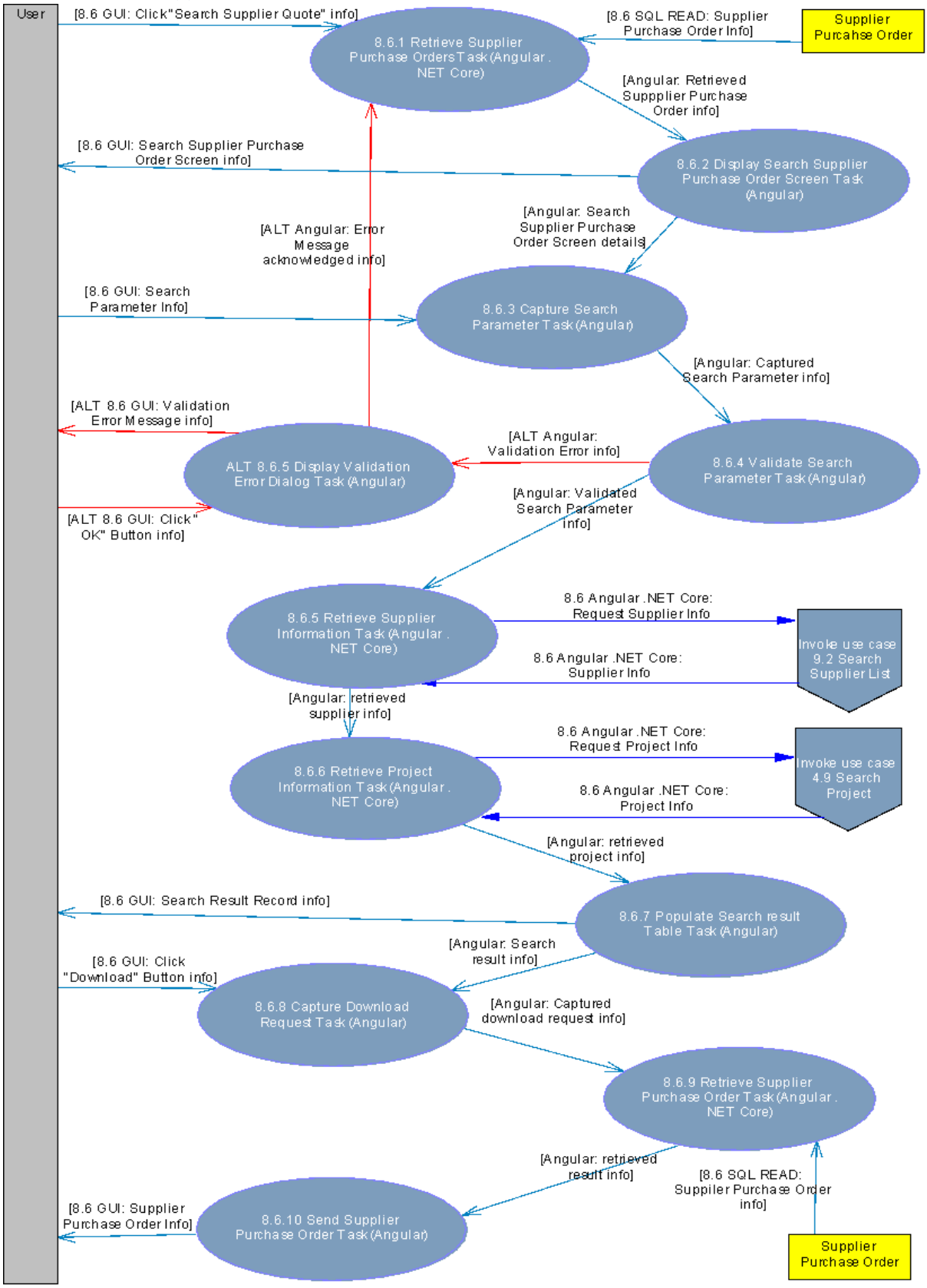


Figure 227: Primitive level data diagram8.6 Search Supplier Purchase Order

2.3.3.9. Supplier Subsystem

2.3.3.9.1 Add Supplier

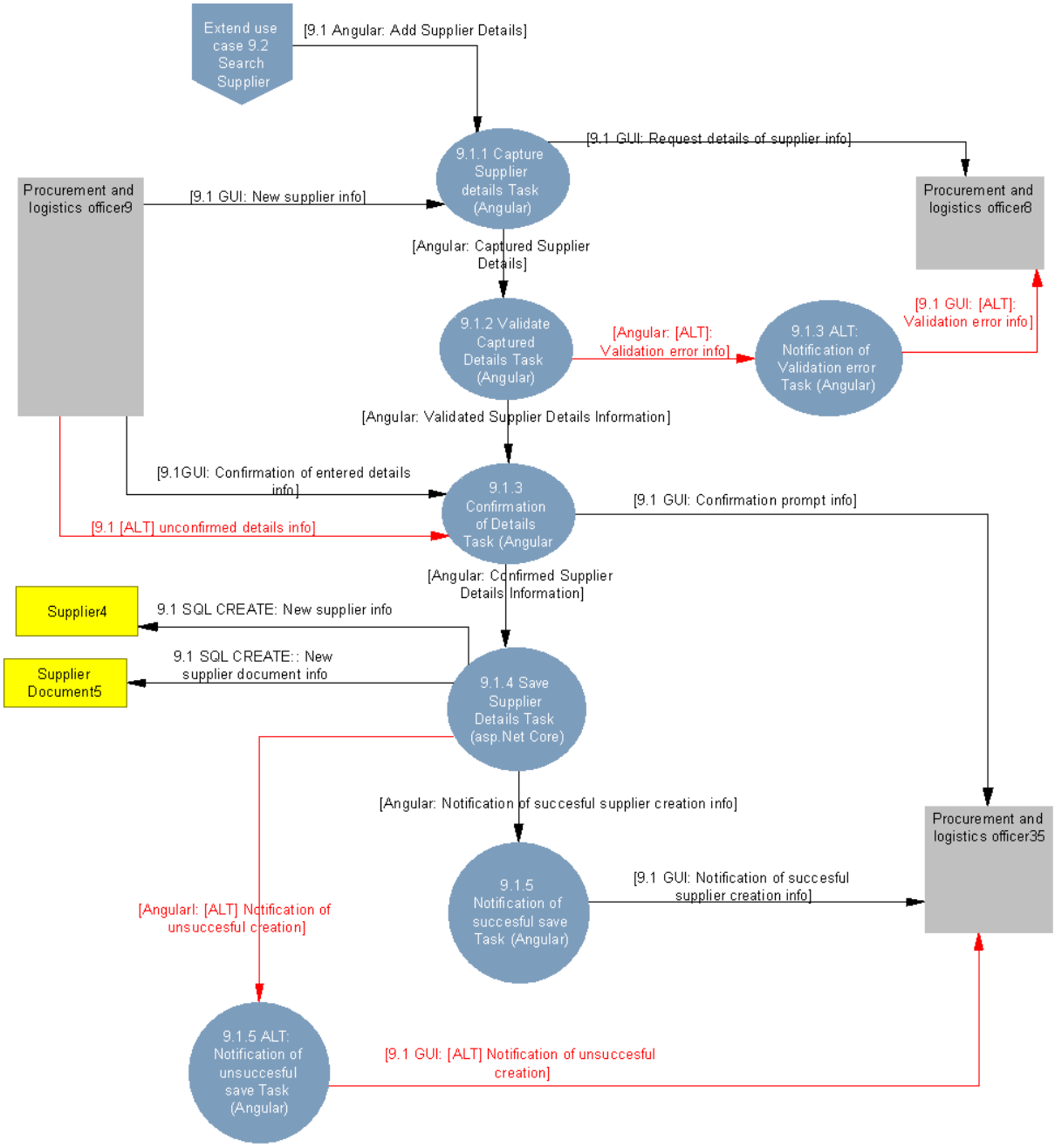


Figure 228: Primitive level data diagram9.1 Add Supplier

2.3.3.9.2 Search Supplier

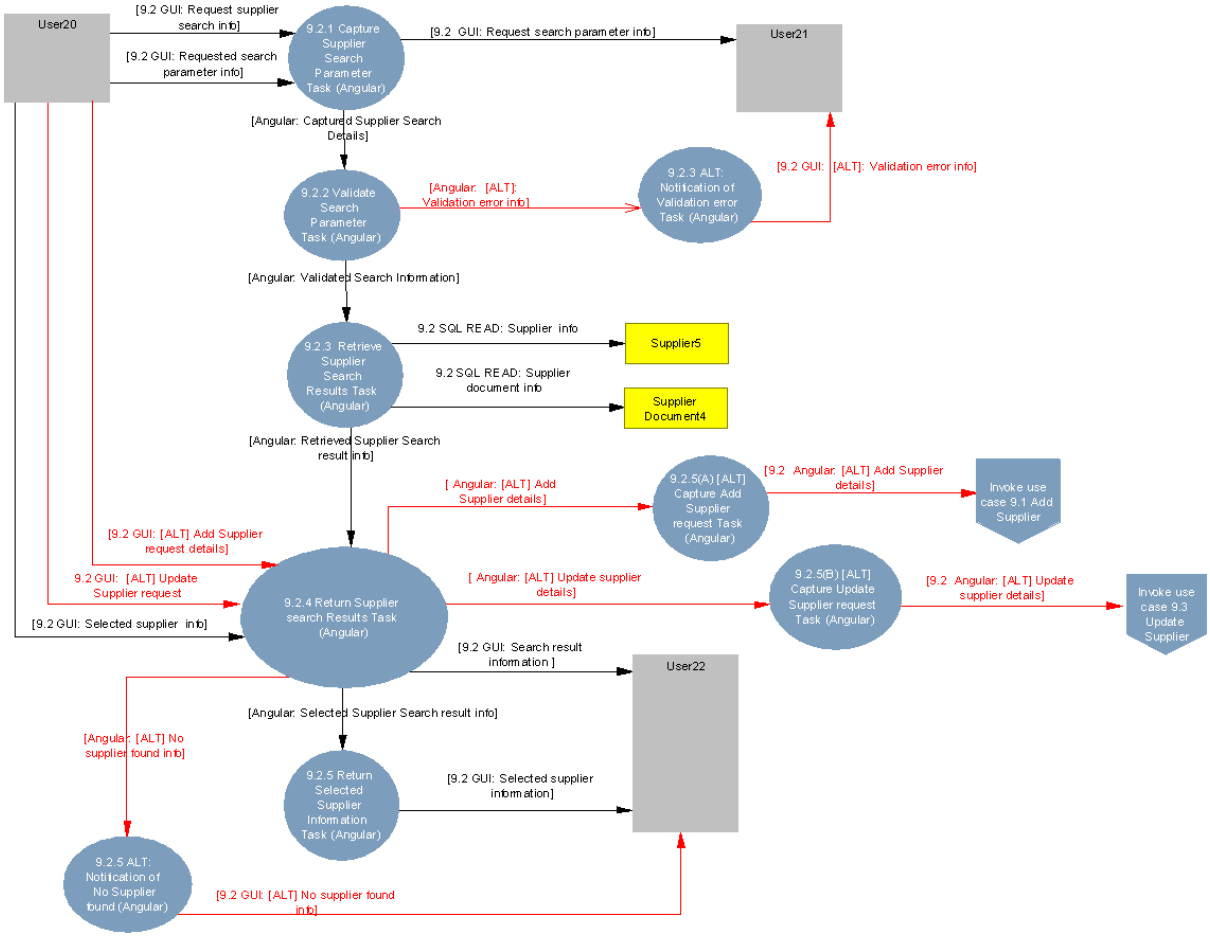


Figure 229: Primitive level data diagram9.2 Search Supplier

2.3.3.9.3 Edit Supplier

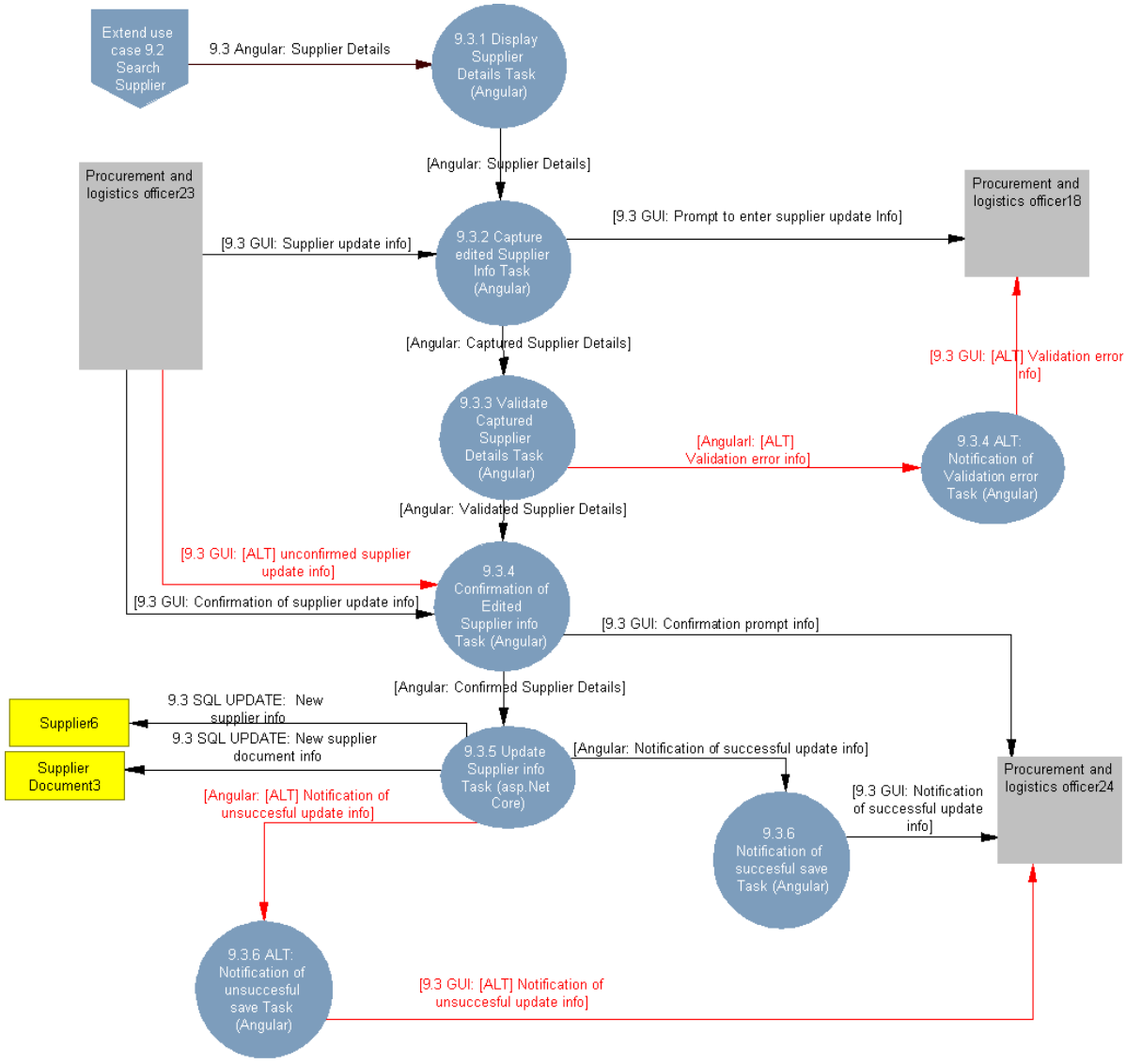


Figure 230: Primitive level data diagram9.3 Edit Supplier

2.3.3.10. Inventory Management Subsystem

2.3.3.10.1 Add-Line Item

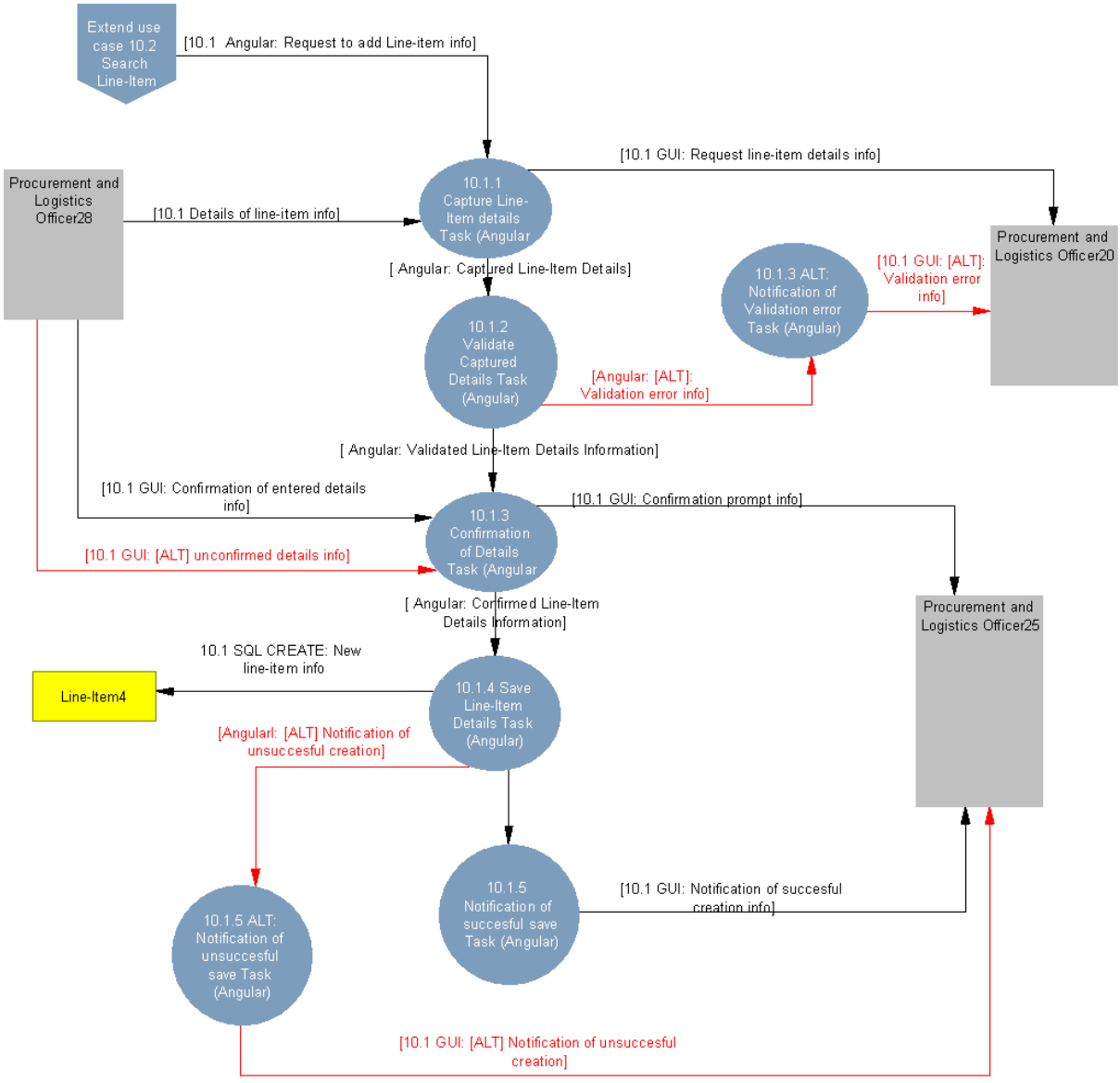


Figure 231: Primitive level data diagram10.1 Add-Line Item

2.3.3.10.2 Search-Line Item

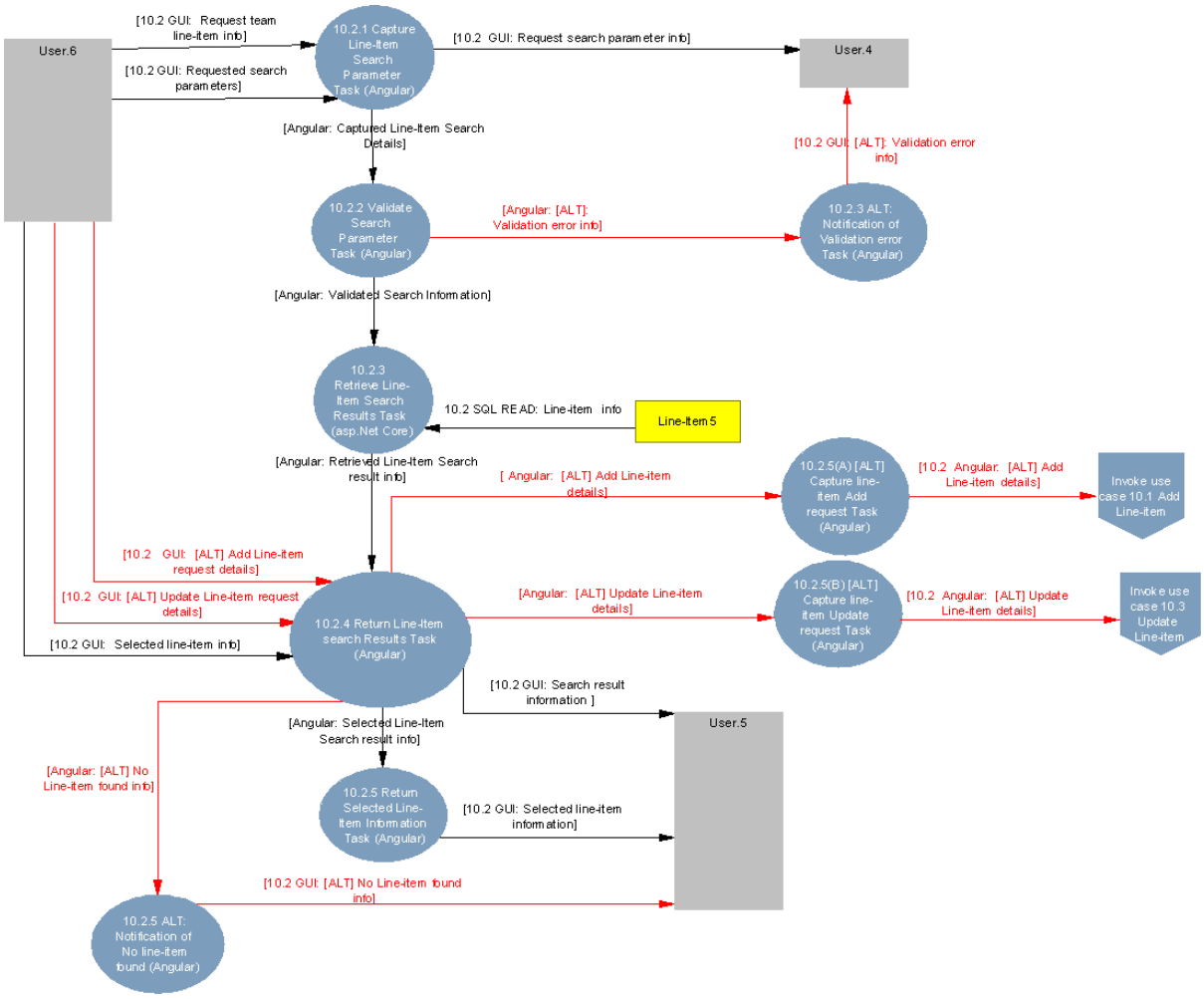


Figure 232: Primitive level data diagram10.2 Search-Line Item

2.3.3.10.3 Edit line-item

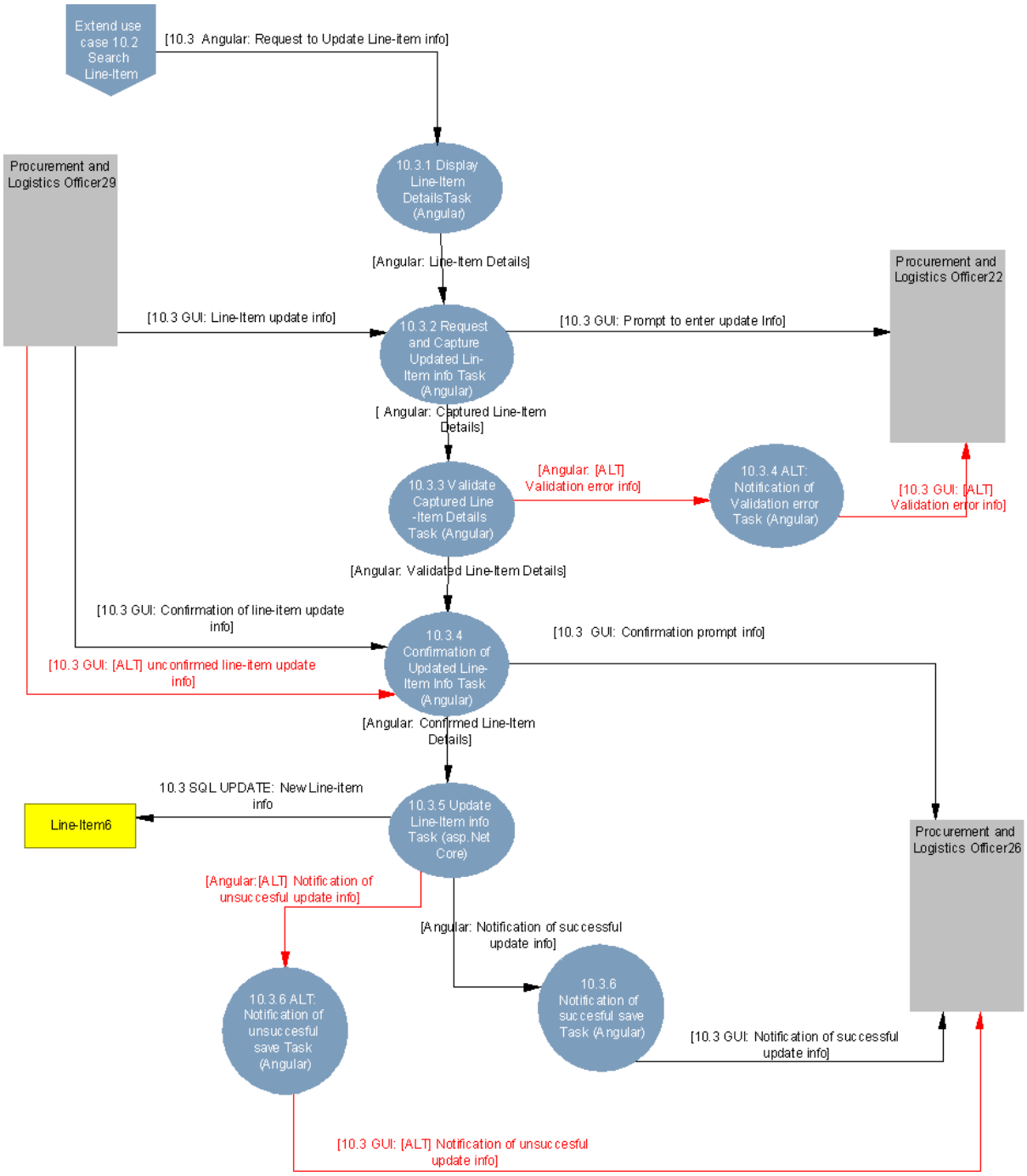


Figure 233: Primitive level data diagram10.3 Edit line-item

2.3.3.11. Reporting Subsystem

2.3.3.11.1 Generate Site Progress Report

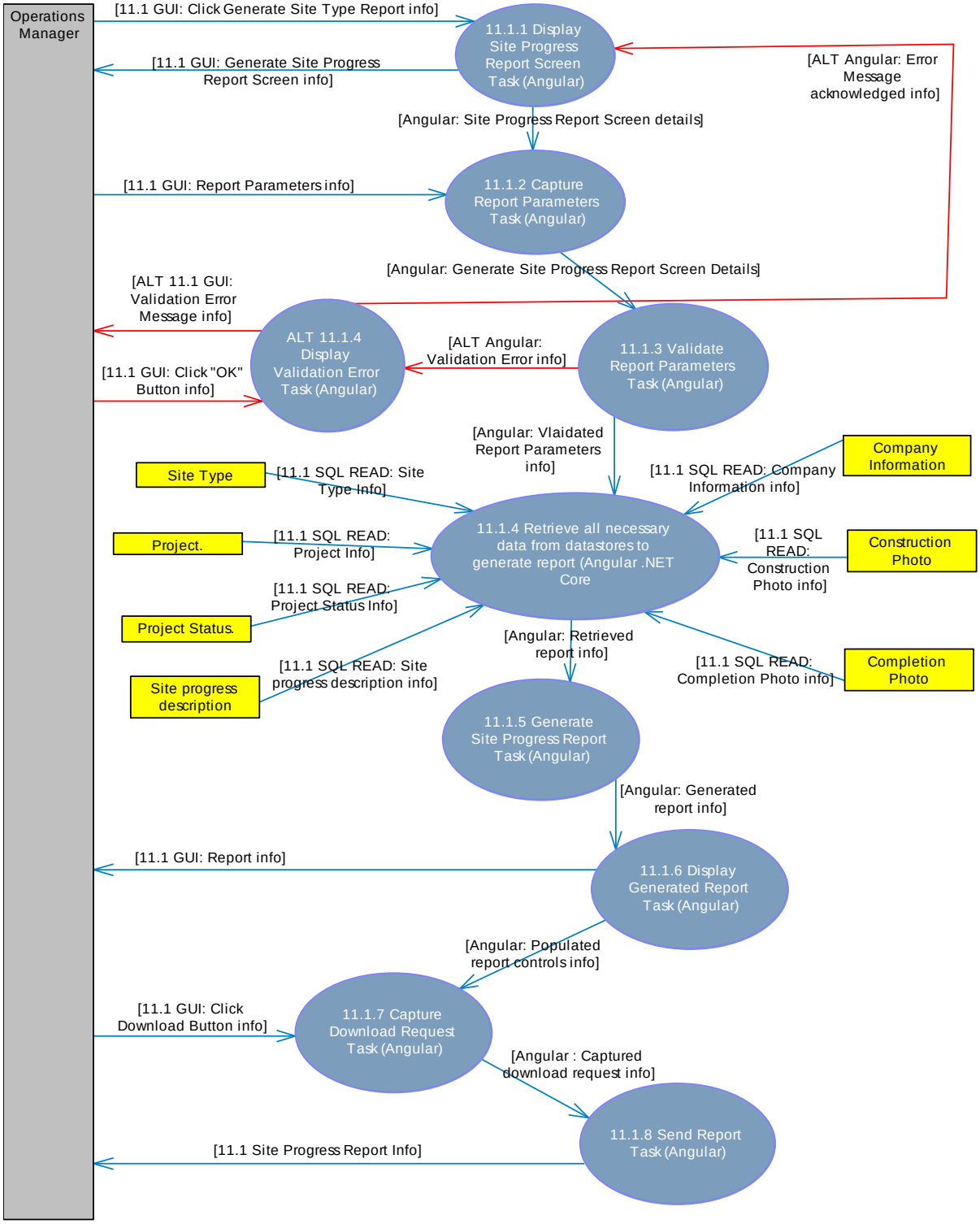


Figure 234: Primitive level data diagram Generate Site Progress Report

2.3.3.11.2 Generate Site Type Report

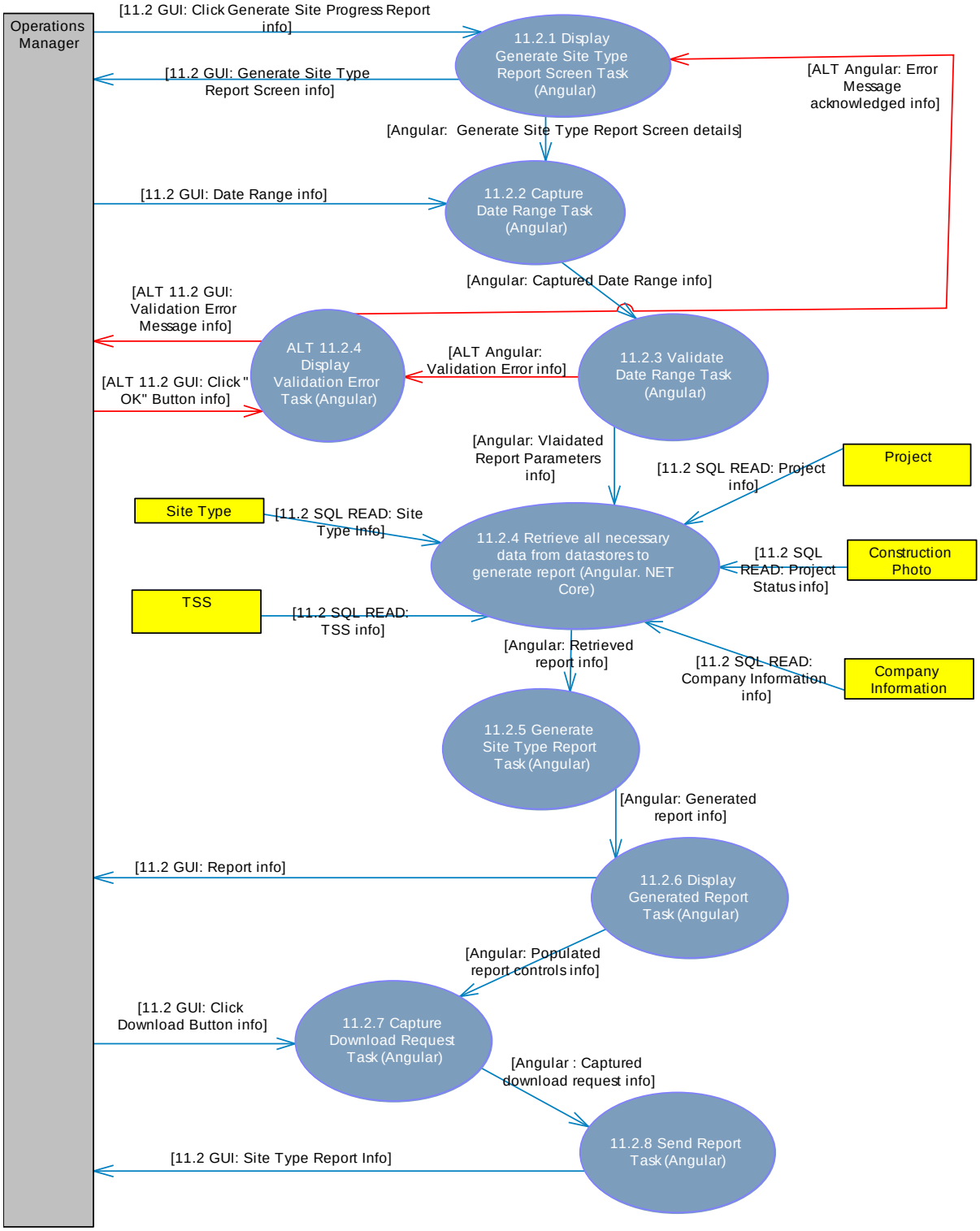


Figure 235: Primitive level data diagram Generate Site Type Report

2.3.3.11.3 Generate Team Site Completed Report

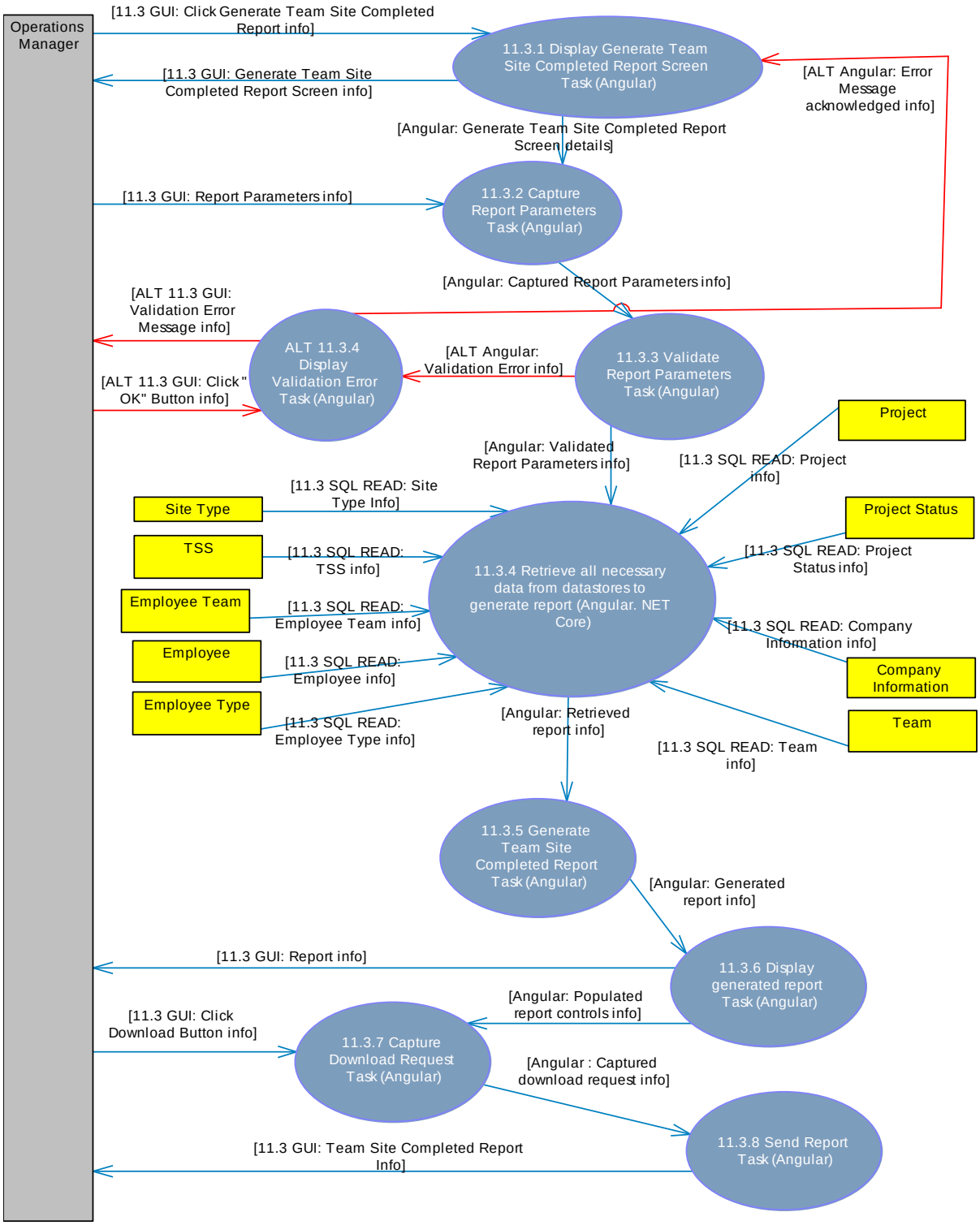


Figure 236: Primitive level data diagram Generate Team Site Completed Report

2.3.3.11.4 Generate Employee Document Expiry Report

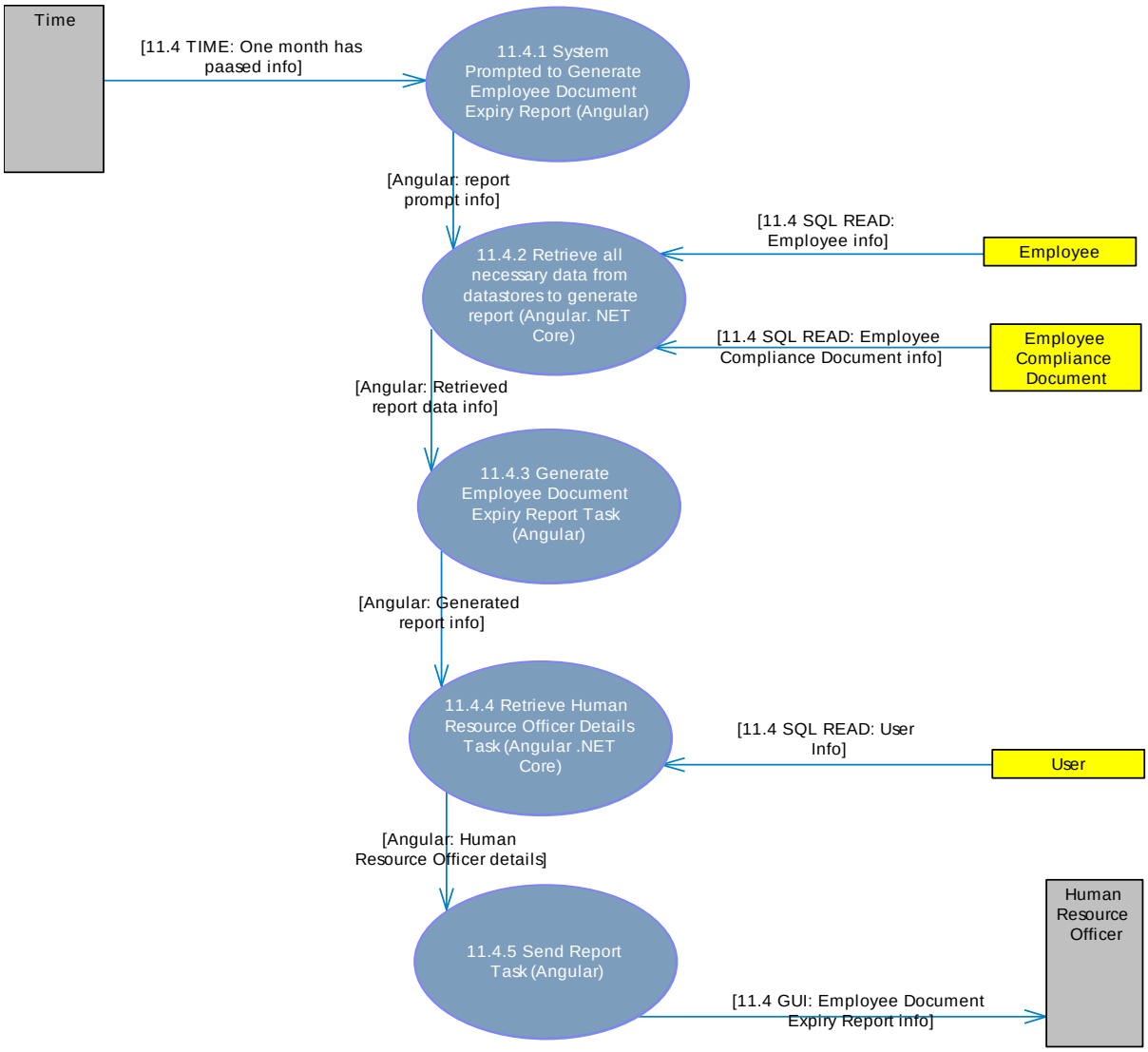


Figure 237: Primitive level data diagram Generate Employee Document Expiry Report

2.3.3.11.5 Generate Site Acceptance Document

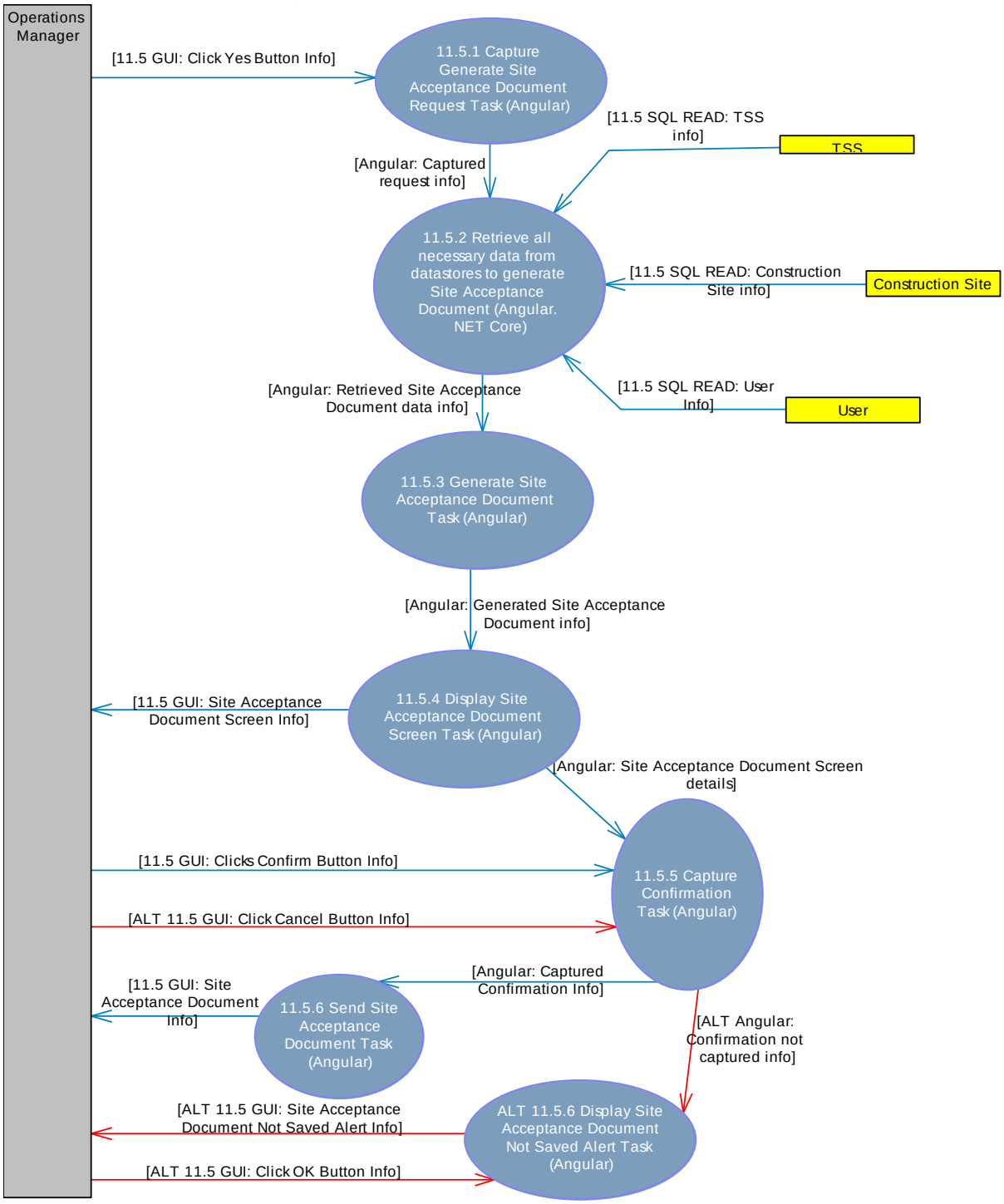


Figure 238: Primitive level data diagram Generate Site Acceptance Document

2.3.3.12. User Management Subsystem

2.3.3.12.1 Login

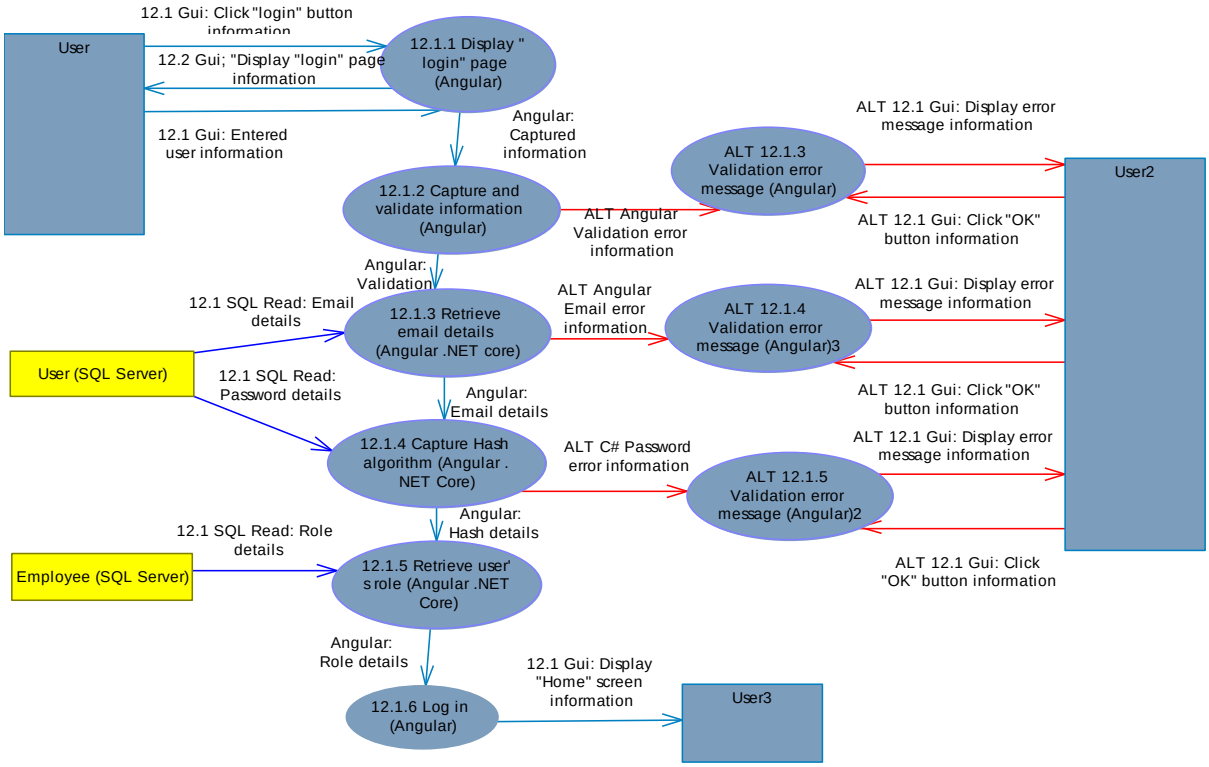


Figure 239: Primitive level data diagram12.1 Login

2.3.3.12.2 Logout

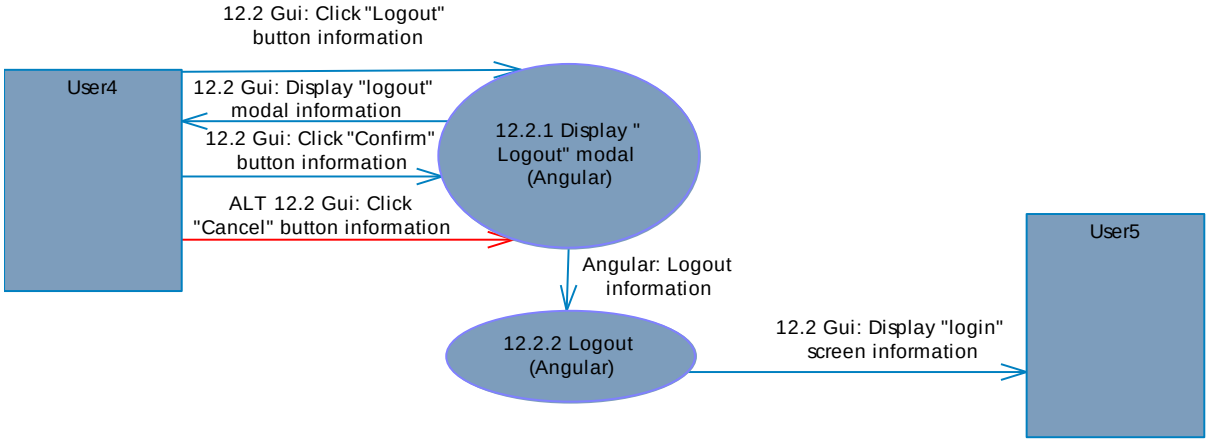


Figure 240: Primitive level data diagram12.2 Logout

2.3.3.12.3 Reset Password

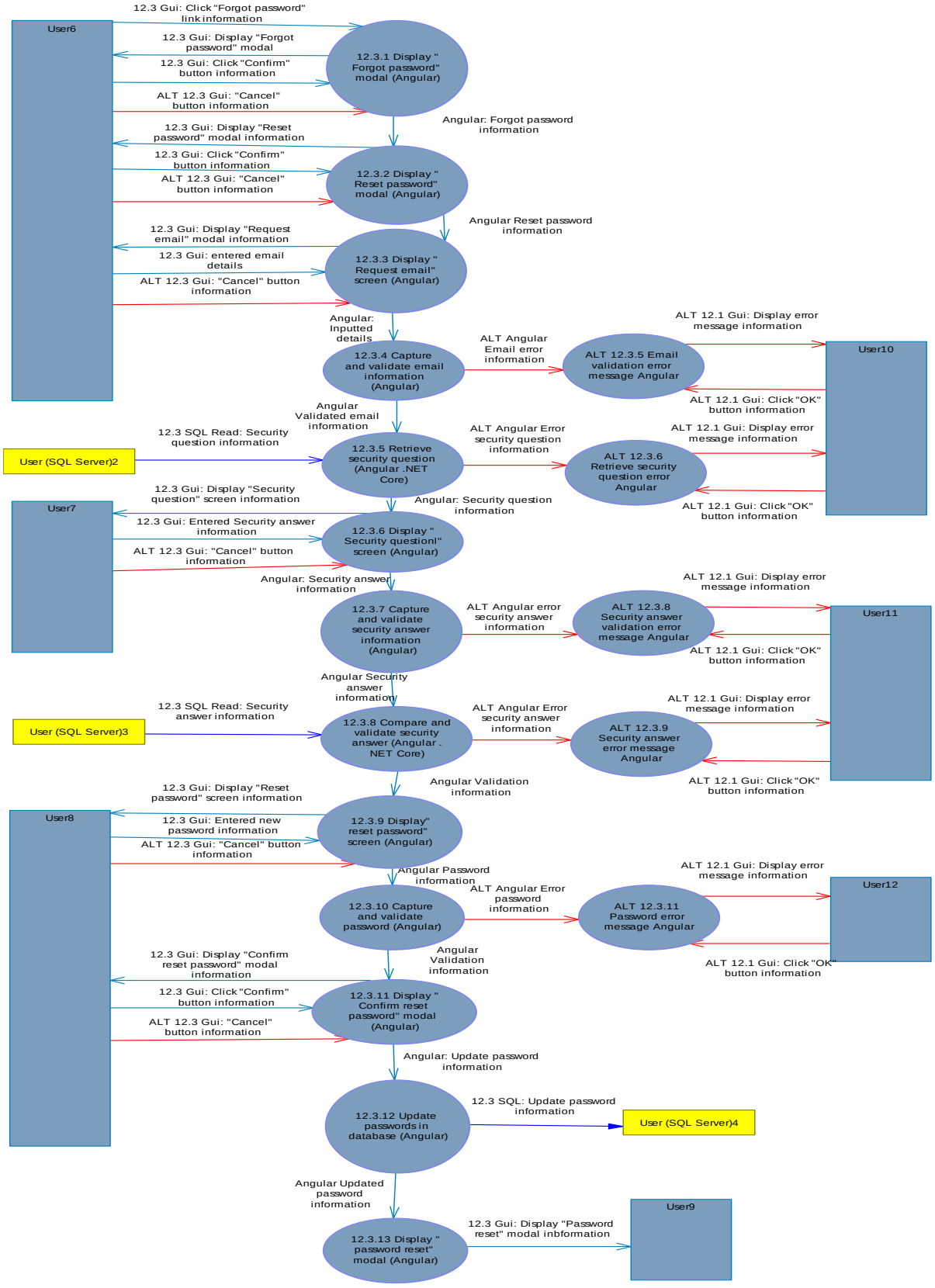


Figure 241: Primitive level data diagram12.3 Reset Password

2.3.3.13. Client Subsystem

2.3.3.13.1 Add Client

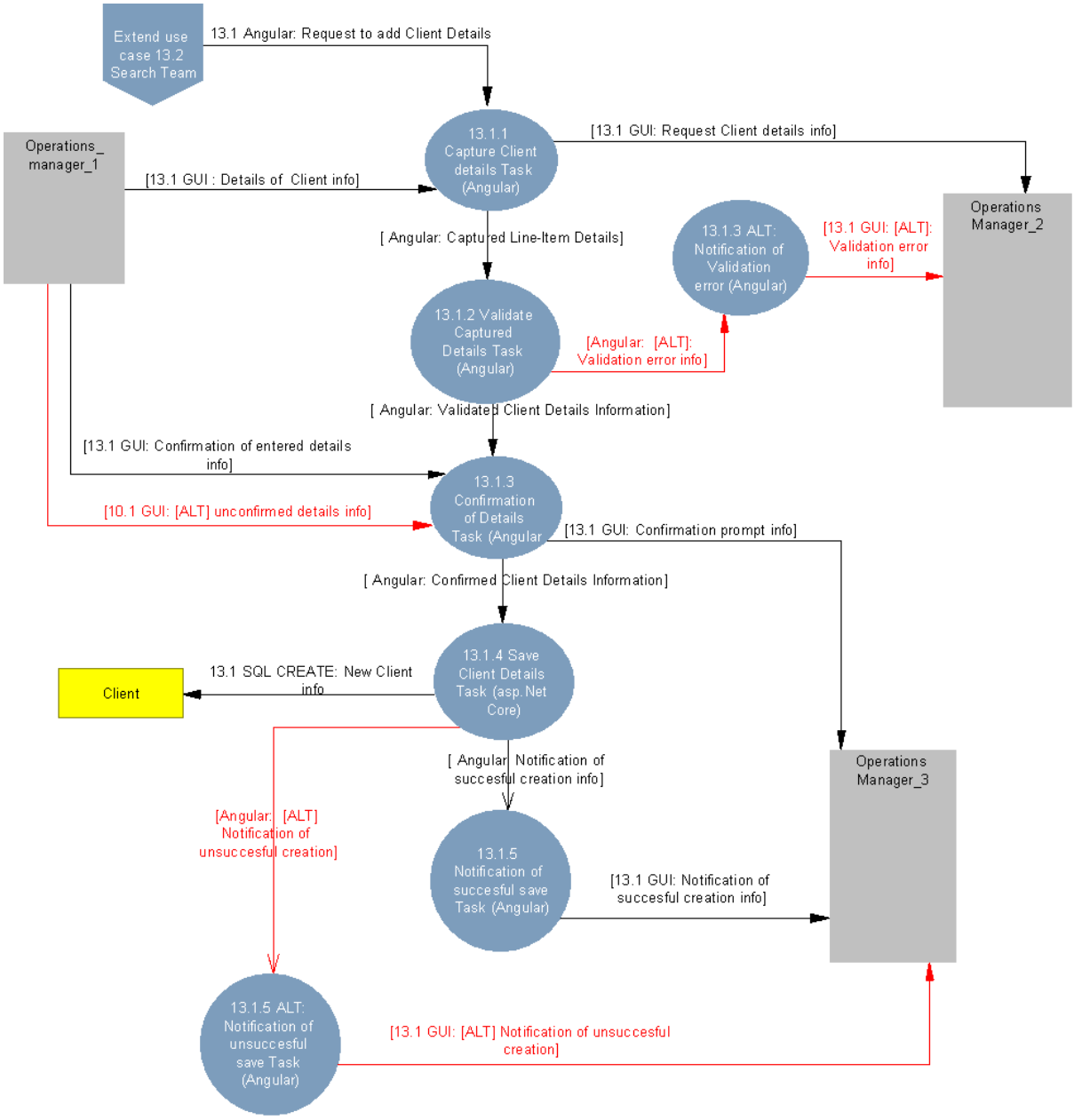


Figure 242: Primitive level data diagram13.1 Add Client

2.3.3.13.2 Search Client

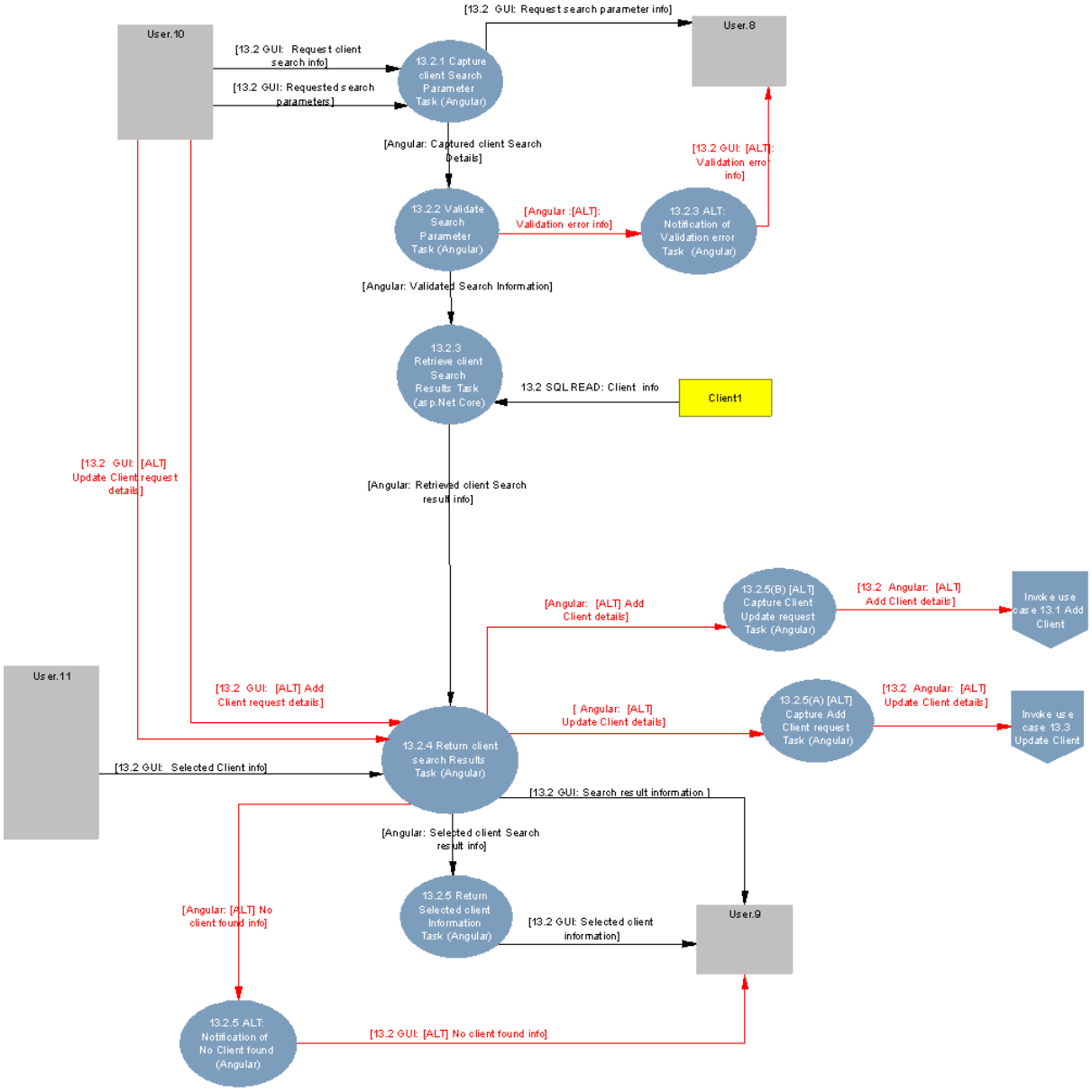


Figure 243: Primitive level data diagram13.2 Search Client

2.3.3.13.3 Edit Client

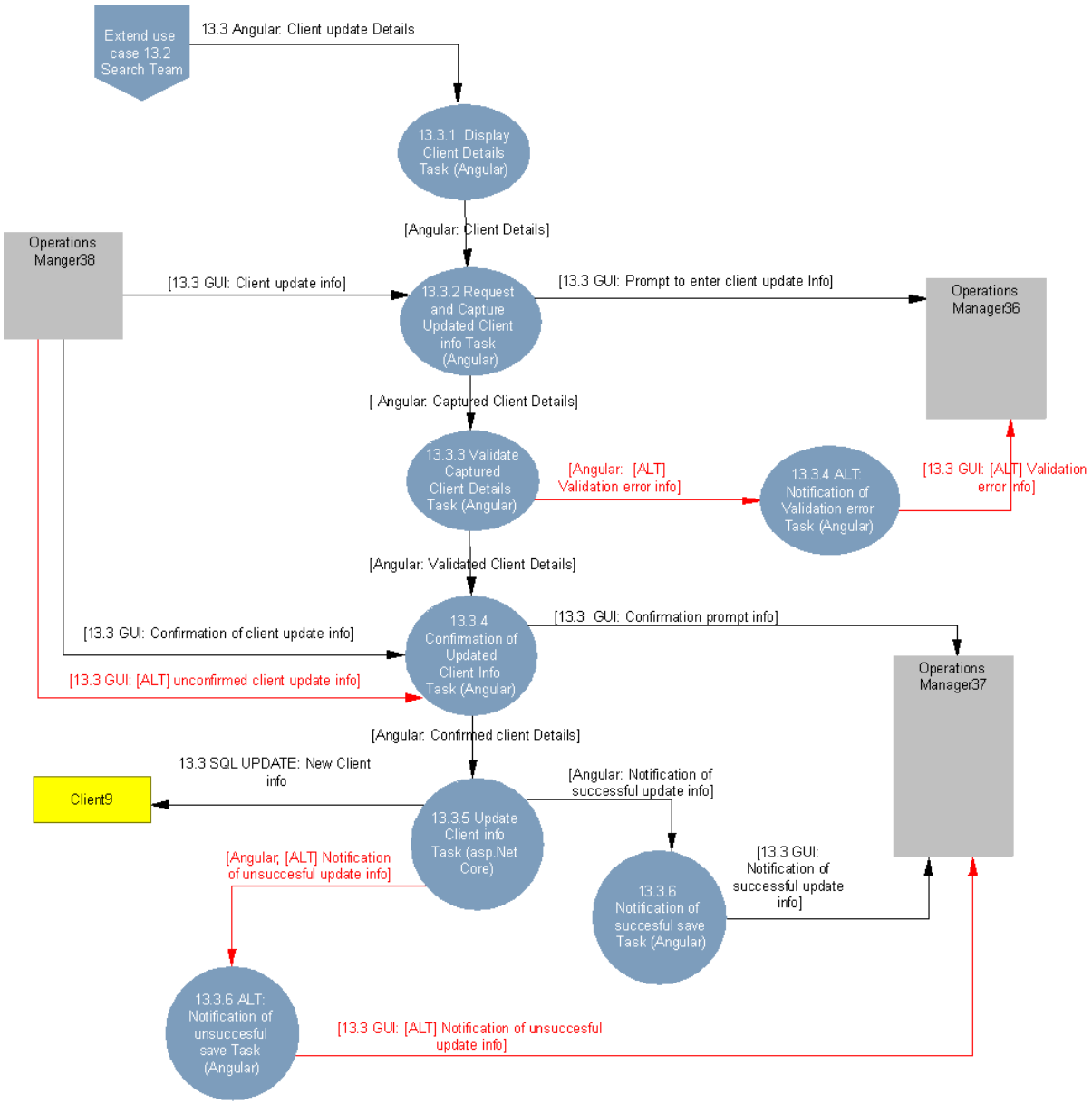


Figure 244: Primitive level data diagram13.3 Edit Client

Conclusion

This section uses various diagrams to describe the processes and layouts of the system for Gohvan Construction.

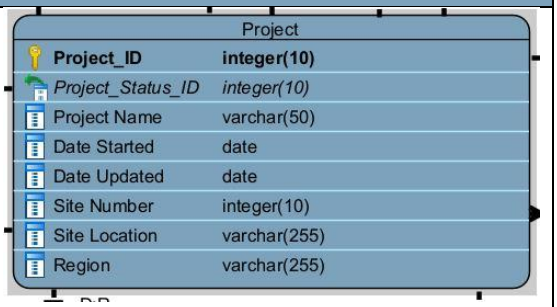
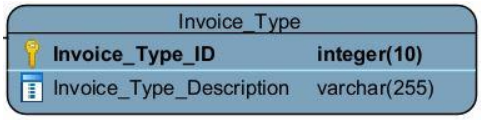
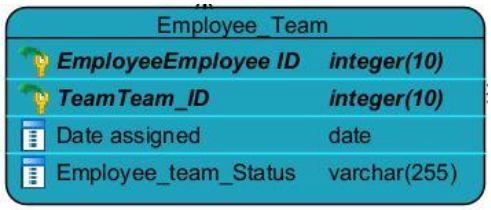
2.4 Pseudo Code

System Database Design

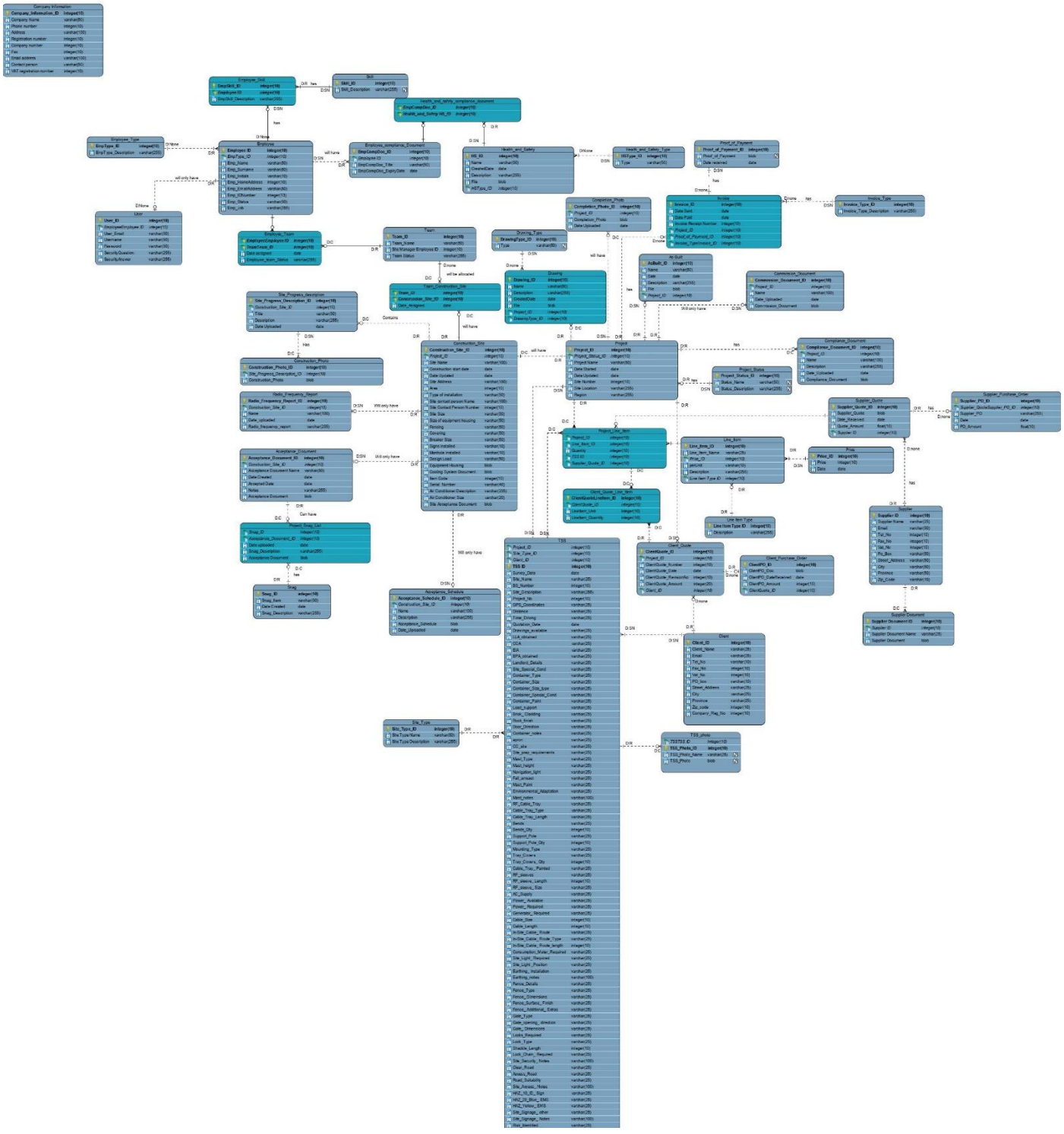
3. Entity Relationship Model

Introduction

This section of the document contains the Logical Entity Relationship Diagram. This contains symbols and connectors that help us to understand the major entities and their attributes within the system and the relationship between them. This diagram will also help us to understand Gohvan Construction's database design clearly.

Data Modelling Diagram Key		
Element	Description	Preview
Standalone Entity	Entity element, filled with a light blue and black text.	
Lookup Entity	Entity element, filled with light blue and black text.	
Associative Entity	Entity element filled with cyan and black text.	
Relationship	Element, with relevant relationship types, with red outline.	

3.1 Entity Relationship Model



Conclusion

In this section we have seen the Logical Entity Relationship Diagram. The major entities and their attributes within the system and the relationship between them have helped us to have a better understanding of the database. We have come to see that it is an important addition to the many diagrams that has been presented in this deliverable because it brings them all together.

4. Size Estimation of proposed database

Introduction

This section will contain the size estimation of the proposed system for Gohvan construction database. The size estimation contains all the database's attributes for its tables and calculates the estimated byte size of all the attributes to propose an estimated size of the database, as well as a 5-year projection of how big the database will grow in the next 5 years.

4.1 Size Estimation of proposed Database

Entity Name	Expected Rows	Entity Size (In Bytes)	Total Size (In Bytes)
EmpType_ID	10	4	40
EmpType_Description	10	257	2570
User_ID	25	4	100
User_Email	25	52	1300
Username	25	52	1300
Password	25	52	1300
SecurityQuestion	25	257	6425
SecurityAnswer	25	257	6425
EmpSkill_ID	200	4	800
Employee ID	200	4	800
EmpSkill_Description	200	257	51400
Company_Information_ID	1	4	4
Company Name	1	52	52
Phone number	1	4	4
Address	1	102	102
Registration number	1	4	4
Company number	1	4	4
Fax	1	4	4
Email address	1	102	102
Contact person	1	52	52
VAT registration number	1	4	4
Employee ID	100	4	400
Emp_Name	100	52	5200
Emp_Surname	100	52	5200
Emp_Initials	100	12	1200
Emp_HomeAddress	100	4	400
Emp_EmailAddress	100	52	5200
Emp_IDNumber	100	4	400
Emp_Status	100	52	5200
Emp_Job	100	257	25700
EmployeeEmployee ID	50	4	200
TeamTeam_ID	50	4	200
Date assigned	50	3	150
Employee_team_Status	50	257	12850
Site_Progress_Description_ID	20	4	80
Title	20	52	1040
Description	20	257	5140
Date Uploaded	20	3	60
Construction_Photo_ID	500	4	2000
Construction_Photo	500	700000	350000000
Radio_Frequency_Report_ID	20	4	80
Name	20	102	2040
Date_uploaded	20	3	60
Radio_frequency_report	20	257	5140
Acceptance_Document_ID	20	4	80

Acceptance Document Name	20	52	1040
Date Created	20	3	60
Accepted Date	20	3	60
Notes	20	257	5140
Acceptance Document	20	50000000	1000000000
Snag_ID	20	4	80
Acceptance_Document_ID	20	4	80
Date uploaded	20	3	60
Snag_Description	20	257	5140
Acceptance Document	20	50000000	1000000000
Snag_ID	100	4	400
Snag_Item	100	52	5200
Date Created	100	3	300
Snag_Description	100	257	25700
Skill_ID	20	4	80
Skill_Description	20	257	5140
EmpCompDoc_ID	100	4	400
EmpCompDoc_Title	100	52	5200
EmpCompDoc_ExpiryDate	100	3	300
EmpCompDoc_ID	100	4	400
Health_and_Safety HS_ID	20	4	80
HS_ID	20	4	80
Name	20	52	1040
CreatedDate	20	3	60
Description	20	257	5140
File	20	300000000	6000000000
Team_ID	10	4	40
Team_Name	10	52	520
Site Manager Employee ID	10	4	40
Team Status	10	257	2570
DrawingType_ID	10	4	40
Type	10	52	520
Team_ID	20	4	80
Construction_Site_ID	20	4	80
Date_Assigned	20	3	60
Drawing_ID	80	4	320
Name	80	52	4160
Description	80	257	20560
CreatedDate	80	3	240
File	80	200000000	16000000000
Construction_Site_ID	20	4	80
Site Name	20	102	2040
Construction start date	20	3	60
Date Updated	20	3	60
Site Address	20	152	3040
Area	20	4	80
Type of installation	20	52	1040

Site contact person Name	20	102	2040
Site Contact Person Number	20	4	80
Site Size	20	52	1040
Size of equipment housing	20	52	1040
Fencing	20	52	1040
Covering	20	52	1040
Breaker Size	20	52	1040
Signs installed	20	12	240
Manhole installed	20	12	240
Design Load	20	52	1040
Equipment Housing	20	200000000	4000000000
Cooling System Document	20	200000000	4000000000
Item Code	20	4	80
Serial Number	20	42	840
Air Conditioner Description	20	257	5140
Air Conditioner Size	20	22	440
Site Acceptance Document	20	50000000	1000000000
Acceptance_Schedule_ID	20	4	80
Name	20	102	2040
Description	20	257	5140
Acceptance_Schedule	20	50000000	1000000000
Date_Uploaded	20	3	60
TSS ID	20	4	80
Survey_Date	20	3	60
Site_Name	20	27	540
BS_Number	20	4	80
Site_Description	20	257	5140
Project_No	20	4	80
GPS_Coordinates	20	27	540
Distance	20	27	540
Time_Driving	20	27	540
Quotation_Date	20	3	60
Drawings_available	20	27	540
CCA	20	27	540
LLA_obtained	20	27	540
EIA	20	27	540
BPA_obtained	20	27	540
Landlord_Details	20	27	540
Site_Special_Cond	20	27	540
Container_Type	20	27	540
Container_Size	20	27	540
Container_Size_Type	20	27	540
Container_Special_Cond	20	27	540
Container_Paint	20	27	540
Load_Support	20	27	540
Brick_Cladding	20	27	540
Rock_finish	20	27	540

Door_Direction	20	27	540
Container_Notes	20	27	540
Apron	20	27	540
CC_site	20	27	540
Site_prep_requirements	20	27	540
Mast_type	20	27	540
Mast_Height	20	27	540
Navigation_light	20	27	540
Fall_arreast	20	27	540
Mast_Paint	20	27	540
Environmental_Adaptation	20	27	540
Mast_Notes	20	102	2040
RF_Cable_Tray	20	27	540
Cable_Tray_Type	20	27	540
Cable_Tray_Length	20	27	540
Bends	20	27	540
Bends_Qty	20	4	80
Support_Pole	20	27	540
Support_Pole_Qty	20	4	80
Mounting_Type	20	27	540
Tray_Covers	20	27	540
Tray_Covers_Qty	20	4	80
Cable_Tray_Painted	20	27	540
RF_sleeves	20	27	540
RF_sleeves_length	20	4	80
RF_sleeves_Size	20	27	540
AC_Supply	20	27	540
Power_Available	20	27	540
Power_Required	20	27	540
Generator_Required	20	27	540
Cable_Size	20	4	80
Cable_Length	20	4	80
In-Site_Cable_Route	20	27	540
In-Site_Cable_Route_Type	20	27	540
In-Site_Cable_Route_Length	20	4	80
Consumption_Meter_Required	20	27	540
Site_Light_Required	20	27	540
Site_Light_Position	20	27	540
Earthing_Installation	20	27	540
Earthing_Notes	20	102	2040
Fence_Details	20	27	540
Fence_Type	20	27	540
Fence_Dimensions	20	27	540
Fence_Surface_Finish	20	27	540
Fence_Additional_Extras	20	27	540
Gate_Type	20	27	540
Gate_opening_direction	20	27	540

Gate_Dimensions	20	27	540
Locks_Required	20	27	540
Lock_type	20	27	540
Shackle_Length	20	4	80
Lock_Chain_Required	20	27	540
Site_security_notes	20	102	2040
Clear_road	20	27	540
Access_road	20	27	540
Road_Suitability	20	27	540
Site_Access_Notes	20	102	2040
HAZ_10_ID_Sign	20	27	540
HAZ_29_Blue_EMS	20	27	540
Site_Signage_other	20	27	540
Site_Signage_Notes	20	102	2040
Risk_Identified	20	27	540
Site_Type_ID	4	4	16
Site Type Name	4	52	208
Site Type Description	4	257	1028
HS_ID	20	4	80
Name	20	52	1040
CreatedDate	20	3	60
Description	20	257	5140
File	20	300000000	6000000000
HSType_ID	25	4	100
Type	25	52	1300
Completion_Photo_ID	200	4	800
Completion_Photo	200	700000	140000000
Date Uploaded	200	3	600
DrawingType_ID	5	4	20
Type	5	52	260
Project_ID	20	4	80
Project Name	20	52	1040
Date Started	20	3	60
Date Updated	20	3	60
Site Number	20	4	80
Site Location	20	257	5140
Region	20	257	5140
Quantity	1200	4	4800
ClientQuoteLineItem_ID	1500	4	6000
ClientQuoteLineItem_Unit	1500	4	6000
ClientQuoteLineItem_Quantity	1500	4	6000
ClientQuote_ID	25	4	100
ClientQuote_Number	25	12	300
ClientQuote_Date	25	3	75
ClientQuote_RevisionNo	25	12	300
ClientQuote_Amount	25	22	550
Proof_of_Payment_ID	40	4	160

Proof_of_Payment	40	1000000	40000000
Date received	40	3	120
Invoice_ID	50	4	200
Date Sent	50	3	150
Date Paid	50	3	150
Invoice Receipt Number	50	4	200
AsBuilt_ID	20	4	80
Name	20	52	1040
Date	20	3	60
Description	20	257	5140
File	20	300000000	6000000000
Commission_Document_ID	20	4	80
Name	20	102	2040
Date_Uploaded	20	3	60
Commission_Document_ID	20	4	80
Project_Status_ID	15	4	60
Status_Name	15	52	780
Status_Description	15	257	3855
Line_Item_ID	1500	4	6000
Line_Item_Name	1500	27	40500
perUnit	1500	12	18000
Description	1500	257	385500
Line Item Type ID	12	4	48
Description	12	257	3084
ClientPO_ID	25	4	100
ClientPO_Doc	25	30000000	750000000
ClientPO_DateReceived	25	3	75
ClientPO_Amount	25	4	100
Client_ID	2	4	8
Client_Name	2	27	54
Email	2	27	54
Tel_No	2	4	8
Fax_No	2	4	8
Vat_No	2	4	8
PO_box	2	4	8
Street_Address	2	27	54
City	2	27	54
Province	2	27	54
Zip_code	2	4	8
Company_Reg_No	2	4	8
TSS_Photo_ID	500	4	2000
TSS_Photo_Name	500	27	13500
TSS_Photo	500	700000	350000000
Invoice_Type_ID	3	4	12
Invoice_Type_Description	3	257	771
Compliance_Document_ID	20	4	80
Name	20	102	2040

PARALINEAR

Description	20	257	5140
Date_Uploaded	20	3	60
Compliance_Document	20	30000000	600000000
Supplier_Quote_ID	80	4	320
Supplier_Quote	80	30000000	2400000000
Date_Received	80	3	240
Quote_Amount	80	4	320
Price_ID	1500	4	6000
Price	1500	4	6000
Date	1500	3	4500
Supplier ID	40	4	160
Supplier Name	40	27	1080
Email	40	52	2080
Tel_No	40	4	160
Fax_No	40	4	160
Vat_No	40	4	160
Po_Box	40	52	2080
Street_Address	40	52	2080
City	40	52	2080
Province	40	52	2080
Zip_Code	40	4	160
Supplier Document ID	40	4	160
Supplier Document Name	40	27	1080
Supplier Document	40	30000000	1200000000
Supplier_PO_ID	30	4	120
Supplier_PO	30	257	7710
Date	30	3	90
PO_Amount	30	4	120
Total Size in Bytes (Excluding Overhead factor)			51830900250
30% overhead Factor			15549270075
Total Size in Bytes (Including Overhead factor)			67380170325

4.2 Five Year Projection

Year	Expected Growth	Expected Size (in Bytes)
Current 2021	N/A	67380170325
2022	50%	101070255488
2023	100%	134760340650
2024	150%	168450425813
2025	200%	202140510975
Overall Total in Bytes		673801703250,00
Overall Total in Kilobytes (1KB = 1000bytes)		673801703,25
Overall Total in Megabytes (1MB = 1000KB)		673801,70
Overall Total in Gigabytes (1GB=1000MB)		673,80
Overall Total in Terabytes (1TB = 1000GB)		0,67

Conclusion

This section contained the size estimation of the proposed system for Gohvan construction database. The size estimation contained all the database's attributes for its tables and calculated the estimated byte size of all the attributes to propose an estimated size of the database, as well as a 5-year projection of how big the database will grow in the next 5 years.

System Interface Design

5. Input Design

Introduction

This section contains a detailed description of the screens, dialogs and modals for the Website application as well as the Mobile application. It describes the views controls and functionalities.

5.1 Complete Input design description

Please see File Deliverable 4: Appendix A for Input Design

Conclusion

This section contained a detailed description of the screens, dialogs and modals for the Website application as well as the Mobile application. It described the views controls and functionalities.

6. Output Design

Introduction

This section contains descriptions of each report and its layout. It will also indicate when it will be generated, and possible entities involved.

6.1 Complete Output design

6.1.1.11 Report Subsystem

6.1.1.11.1 Generate Site Progress Report



Report Number:	11.1
Report Name:	Site Progress Report
Short Description:	This output is created when the Operations Manager click the Generate Site Progress button.
Frequency:	Every Quarter
Special Features:	Report based on start and end dates provided.

Field Name	Format	Max length	Justified	Leading Zero Suppression?	Overflow?	Source
Date Generated	DateTime	dd/mm/yyyy	Right	No	No	System.DateTime.Now on generation of the report.
Generated By	varchar	50	Right	N/A	No	Employee Table EmpName EmpSurname
Company Name	varchar	(50	Center	No	No	Company Information Table
Company Phone Number	integer	10	Center	No	No	Company Information Table Phone number
Company Email	varchar	100	Center	No	No	Company Information Table Email address
Company Address	varchar		Center	No	No	Company Information Table Address
Start Date	DateTime	dd/mm/yyyy	Left	No	No	Captured form the generate site progress report screen
End Date	DateTime	dd/mm/yyyy	Left	No	No	Captured form the generate site progress report screen
Site type	varchar	50	Right	No	No	Site_Type Table Site Type Name
Site Progress Bar Chart	Chart	-	Center	No	NO	Generated from the aggregated data collected.
Total Number of Sites (In this duration):	Numeric	10	Left	No	No	Calculated Total Number Of Sites.
Number of completed Sites	Numeric	10	Left	No	No	Calculated Number of completed Sites.
Number of on-going Sites	Numeric	10	Left	No	No	Calculated Number of completed Sites.

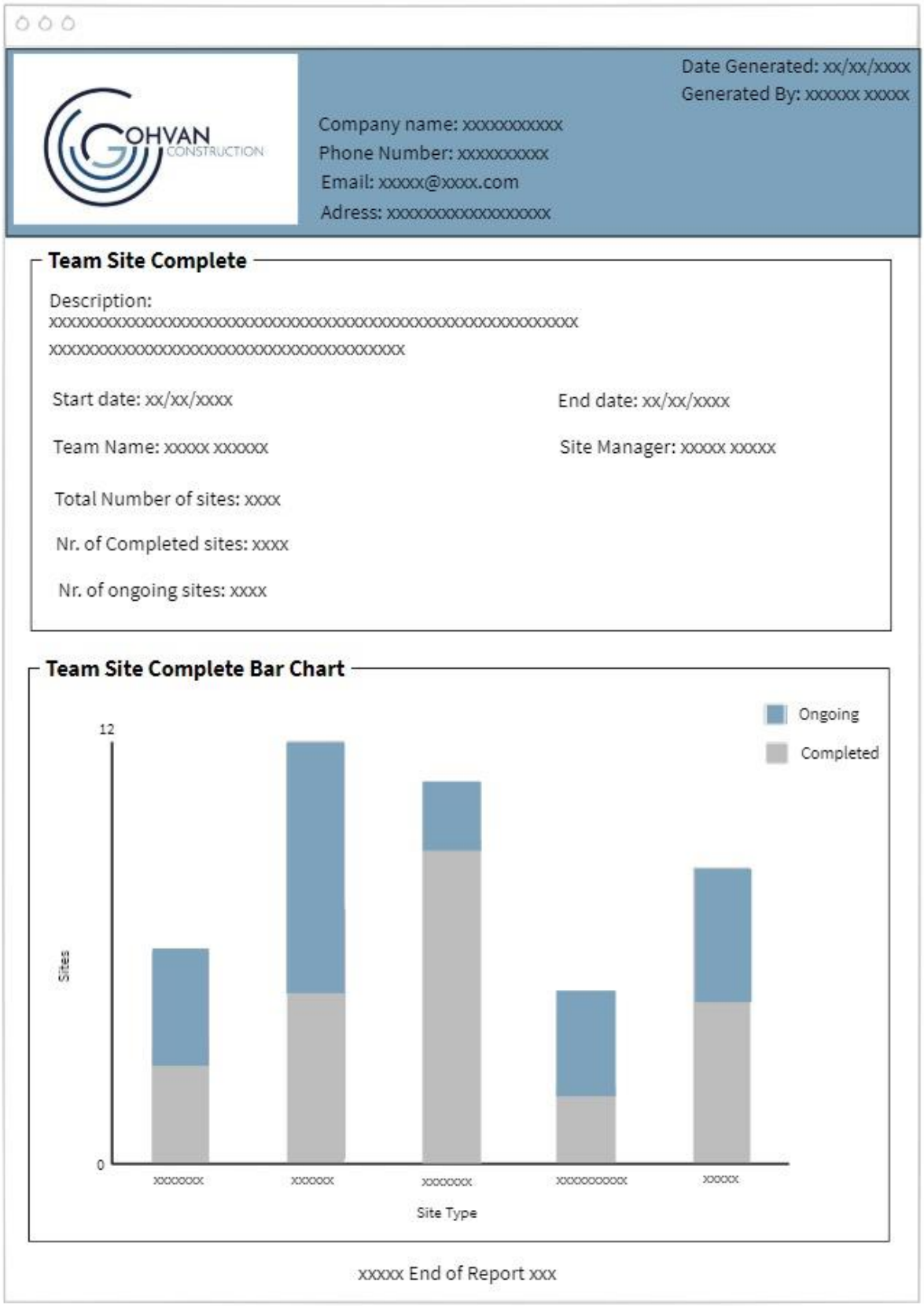
Number of completed Sites	Numeric	10	Left	No	No	Calculated Number of on-going Sites.
Site Name	varchar	50	Left	No	No	Construction_Site table Site Name
Date started	DateTime	Dd/mm/yyyy	Left	No	No	Project Table Date Started
Address	varchar	150	Left	No	No	Construction_Site Table Site Address
Status	varchar	50	Right	No	No	Project_Status Table Status_Name
Area	integer	10	Right	No	No	Construction_Site Table Area
Site Size	varchar	50	Right	No	No	Construction_Site Table Site Size
Title	varchar	50	Left	No	No	Site_Progress_description Table Title
Description	varchar	50	Left	No	No	Site_Progress_description Table Description
Date Uploaded	DateTime	Dd/mm/yyyy	Left	No	No	Site_Progress_description Table Date Uploaded
Completion Photo	Blob	-	Left	No	No	Construction_Photo Table Construction_Photo

6.1.1.11.2 Generate Site Type Report

Report Number:	11.2
Report Name:	Site Type Report
Short Description:	This output is created when the Operations Manager click the Generate Site Type button.
Frequency:	Every Quarter
Special Features:	Report based on start and end dates provided.

Field Name	Format	Max length	Justified	Leading Zero Suppression?	Overflow?	Source
Date Generated	DateTime	dd/mm/yyyy	Right	No	No	System.DateTime.Now on generation of the report.
Generated By	varchar	50	Right	N/A	No	Employee Table EmpName EmpSurname
Company Name	varchar	50	Center	No	No	Company Information Table
Company Phone Number	integer	10	Center	No	No	Company Information Table Phone number
Company Email	varchar	100	Center	No	No	Company Information Table Email address
Company Address	varchar	150	Center	No	No	Company Information Table Address
Start Date	DateTime	dd/mm/yyyy	Left	No	No	Captured form the generate site progress report screen
End Date	DateTime	dd/mm/yyyy	Left	No	No	Captured form the generate site progress report screen
Description	varchar	50	Left	No	No	Hard coded
Total Number of Sites (In this duration):	Numeric	10	Left	No	No	Calculated Total Number Of Sites.
Number of completed Sites:	Numeric	10	Left	No	No	Calculated Number of completed Sites.
Number of on-going Sites:	Numeric	10	Left	No	No	Calculated Number of completed Sites.
Site Type Bar Chart	Chart	-	Center	No	NO	Generated from the aggregated data collected.

6.1.1.11.2 Generate Site Type Report



Report Number:	11.3
Report Name:	Team Site Completed Report

Short Description:	This output is created when the Operations Manager click the Generate Team Site Completed Report button.
Frequency:	Every Quarter
Special Features:	Report based on start and end dates provided.

Field Name	Format	Max length	Justified	Leading Zero Suppression?	Overflow?	Source
Date Generated	DateTime	dd/mm/yyyy	Right	No	No	System.DateTime.Now on generation of the report.
Generated By	varchar	50	Right	N/A	No	Employee Table EmpName EmpSurname
Company Name	varchar	50	Center	No	No	Company Information Table
Company Phone Number	integer	10	Center	No	No	Company Information Table Phone number
Company Email	varchar	100	Center	No	No	Company Information Table Email address
Company Address	varchar		Center	No	No	Company Information Table Address
Start Date	DateTime	dd/mm/yyyy	Left	No	No	Captured form the generate site progress report screen
End Date	DateTime	dd/mm/yyyy	Left	No	No	Captured form the generate site progress report screen
Site type	varchar	50	Right	No	No	Site_Type Table Site Type Name
Team Name	varchar	50	Left	No	No	Team Table Team Name
Description	varchar	50	Left	No	No	Hard coded
Site Manager	varchar	50	Right	No	No	Employee Table EmpName EmpSurname
Total Number of Sites (In this duration):	Numeric	10	Left	No	No	Calculated Total Number Of Sites.
Number of completed Sites:	Numeric	10	Left	No	No	Calculated Number of completed Sites.
Number of on-going Sites:	Numeric	10	Left	No	No	Calculated Number of completed Sites.

Team Site Complete Bar Chart	Chart	-	Center	No	NO	Generated from the aggregated data collected.

6.2 Message Outputs

6.2.1 Establishment Subsystem

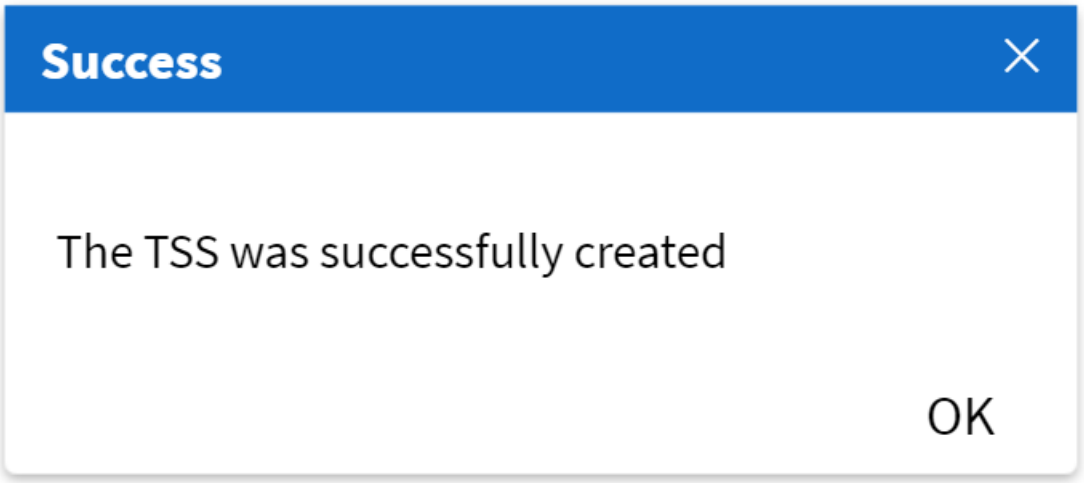


Figure 245: 1. Success modal

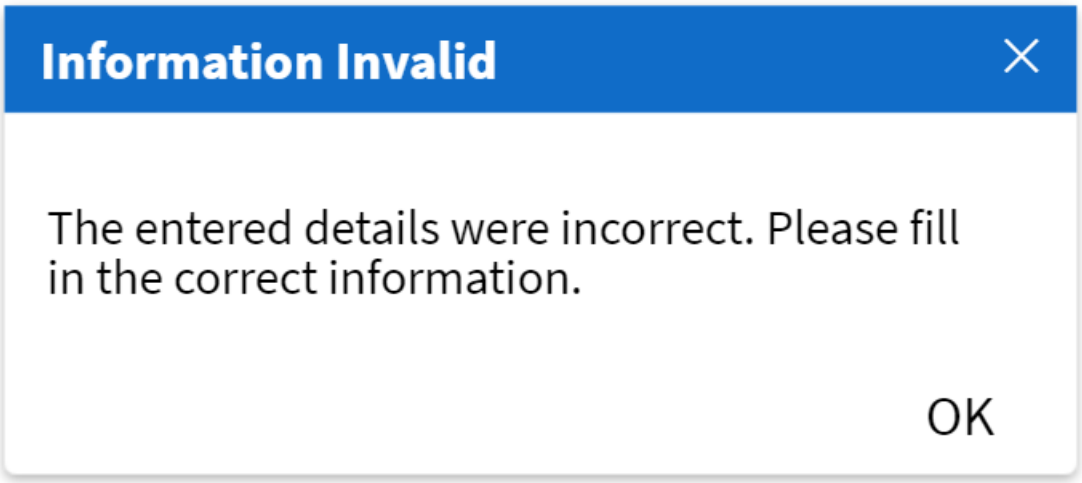


Figure 246: 1. Invalid modal

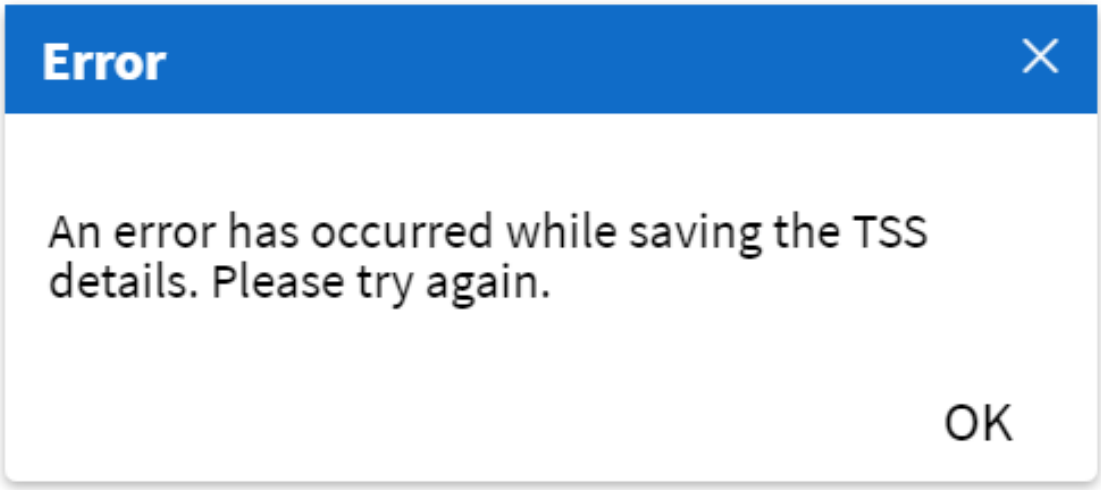


Figure 247: 1. Error modal

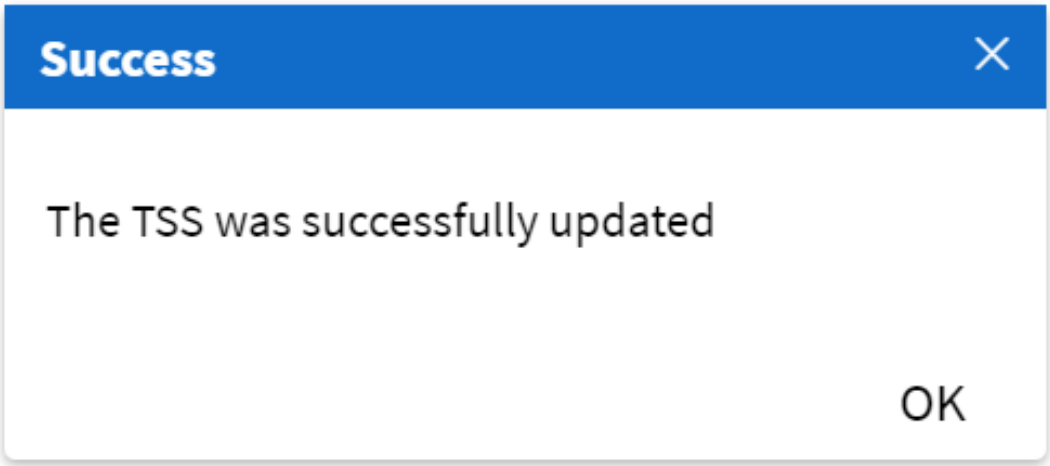


Figure 248: 1. Success modal

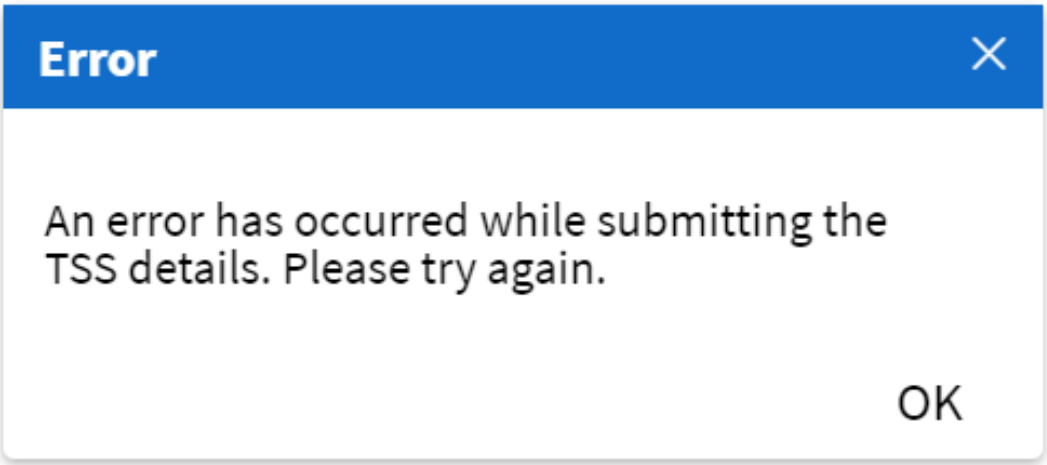


Figure 249: 1. Error modal

6.2.2 Quotation Subsystem

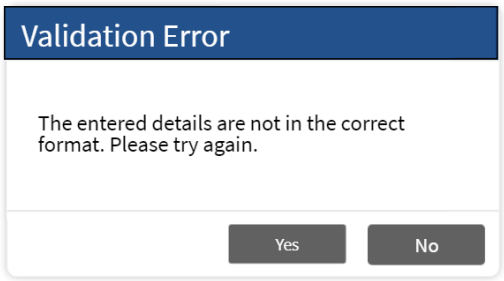


Figure 250: 2. Validation Error Dialog

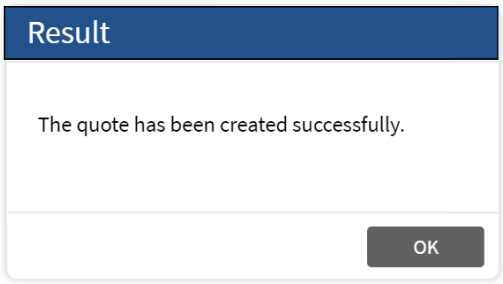


Figure 251: 2. Create Quote Successful Dialog

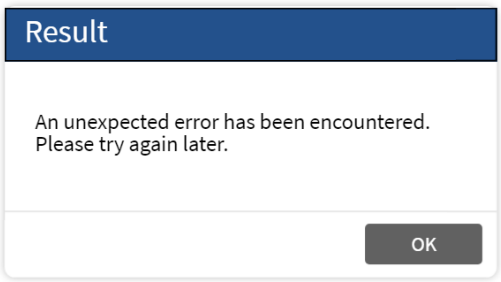


Figure 252: 2. Unexpected Error Dialog

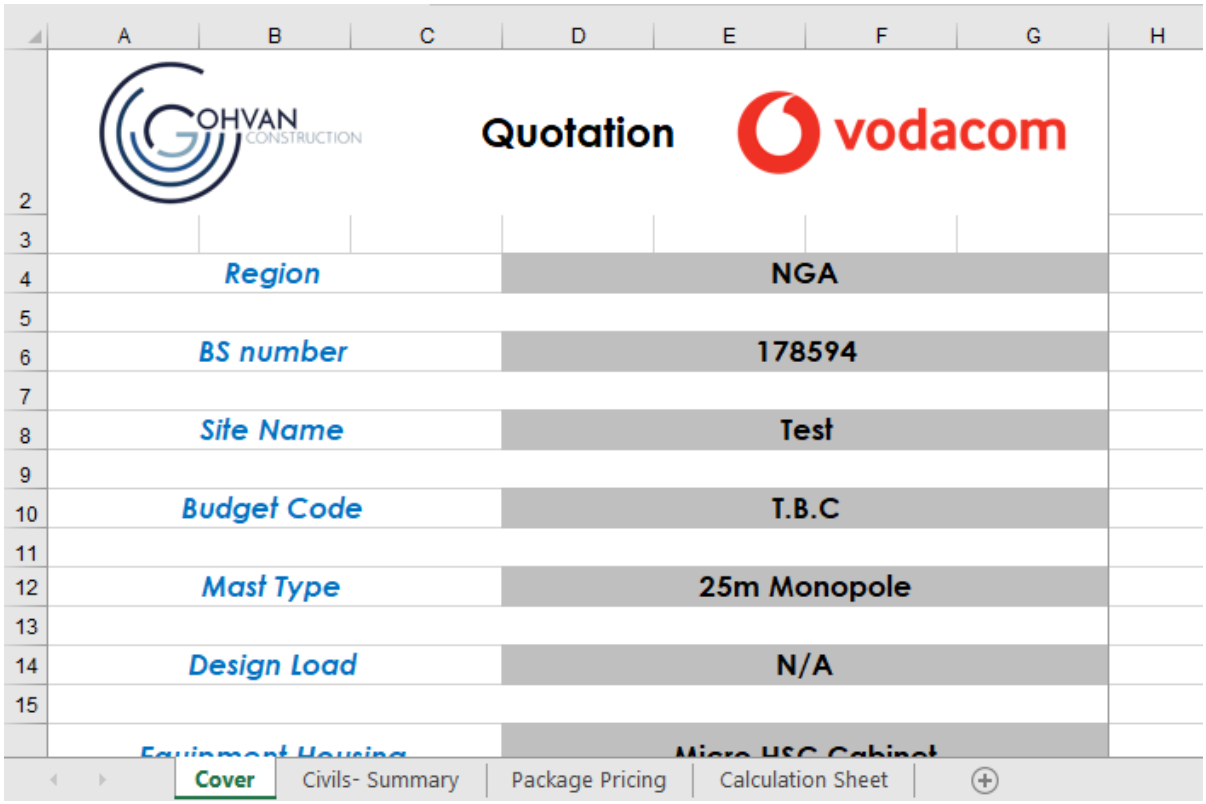


Figure 253: 2. Cover Sheet – 1

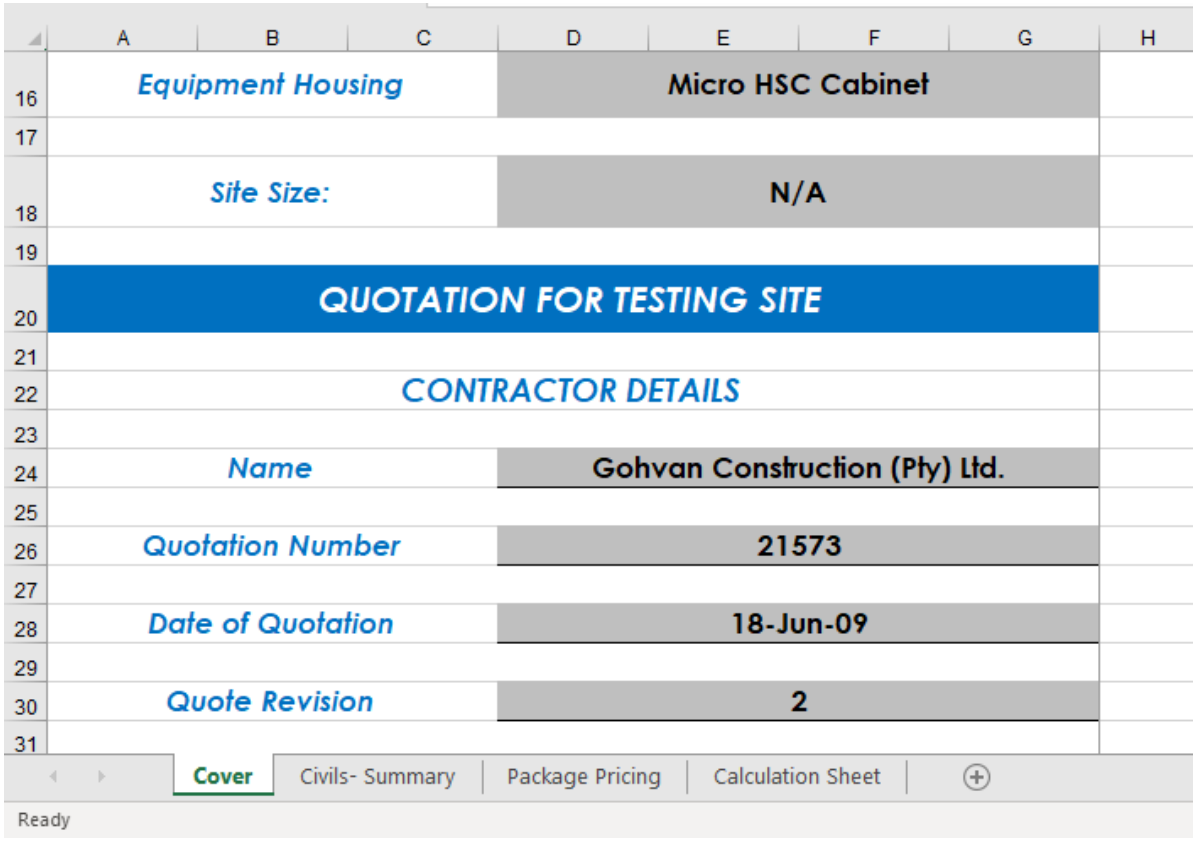
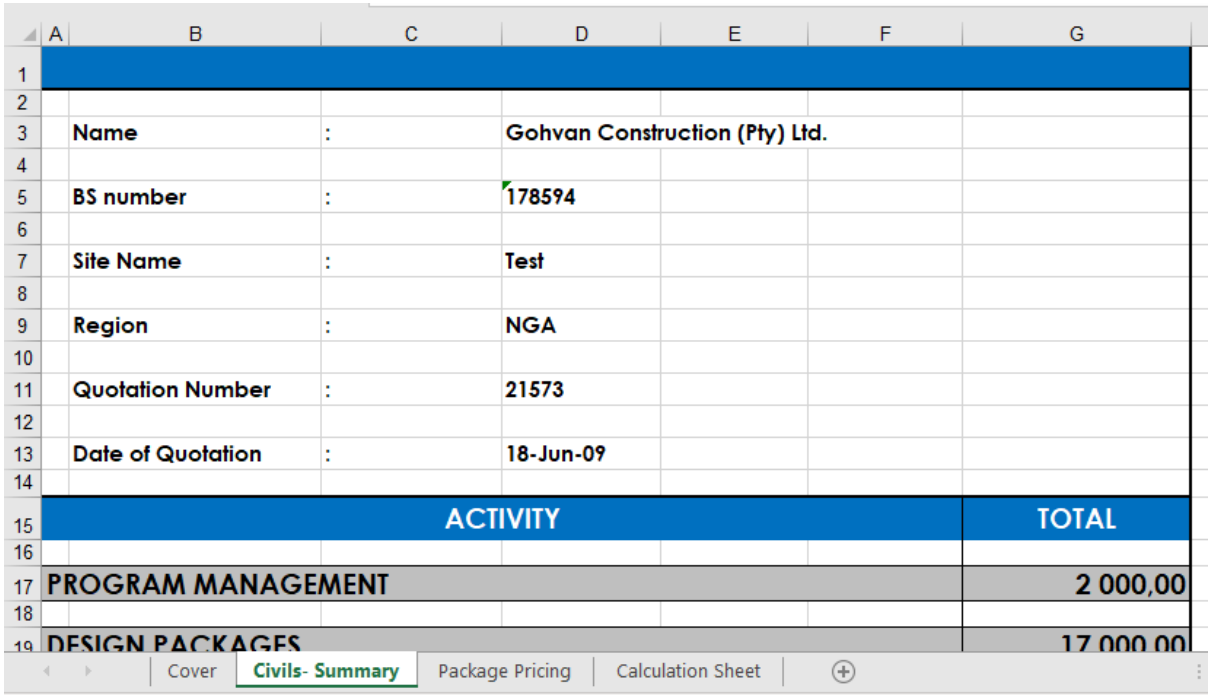


Figure 254: 2. Cover Sheet – 2

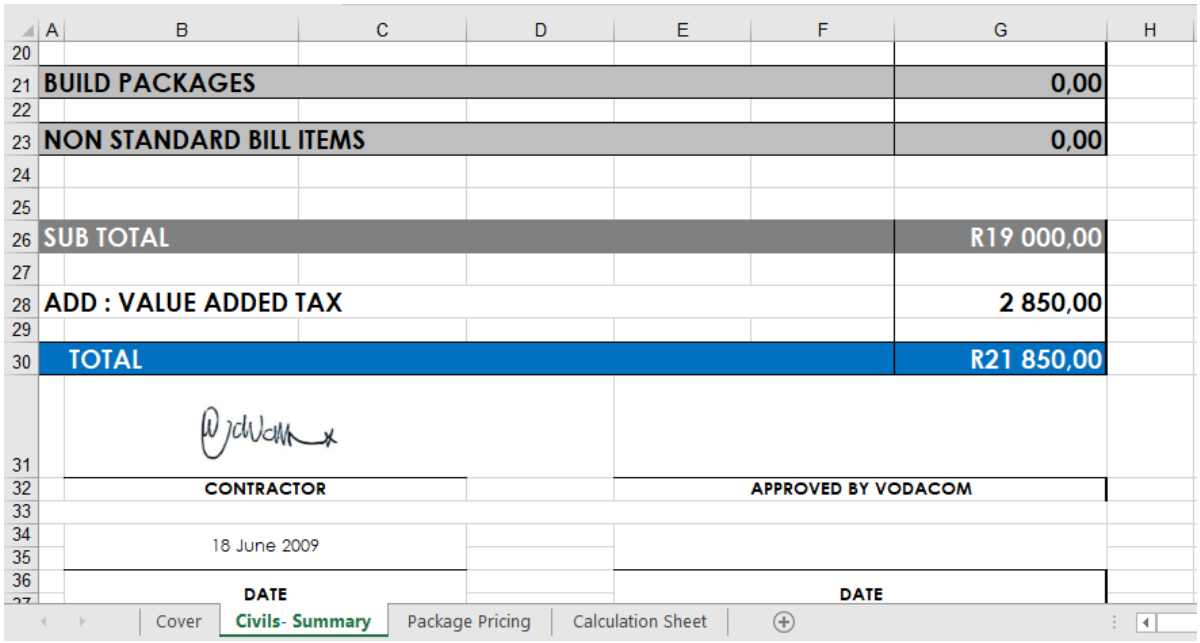


Name	:	Gohvan Construction (Pty) Ltd.	
BS number	:	178594	
Site Name	:	Test	
Region	:	NGA	
Quotation Number	:	21573	
Date of Quotation	:	18-Jun-09	

ACTIVITY	TOTAL
PROGRAM MANAGEMENT	2 000,00
DESIGN PACKAGES	17 000,00

Cover | Civils- Summary | Package Pricing | Calculation Sheet

Figure 255: 2. Civils Summary Sheet – 1



BUILD PACKAGES	0,00
NON STANDARD BILL ITEMS	0,00
SUB TOTAL	R19 000,00
ADD : VALUE ADDED TAX	2 850,00
TOTAL	R21 850,00

CONTRACTOR	APPROVED BY VODACOM
18 June 2009	
DATE	DATE

Cover | Civils- Summary | Package Pricing | Calculation Sheet

Figure 256: 2. Civils Summary Sheet – 2

	A	B	C	D	E	F	G
1			Site Name:		Test		
2			BS number:		178594		
3							
4	SAP Material Id	Material Number Used by Vendor	Packages	Unit	Unit Price	Quantity	Total
5			PROGRAM MANAGEMENT				R 3 000,00
6	304360789	GOHVAN_MLA_657	SH PM Shared Sites Metro	per site	R1 000,00	1	R 1 000,00
7	304360789	GOHVAN_MLA_678	SH PM Shared Rural	per site	R2 000,00	1	R 2 000,00
8			DESIGN PACKAGES				R 27 000,00
9	304366789	GOHVAN_MLA_780	SH Design Requirements Metro	per site	R10 000,00	1	R 10 000,00
10	304366789	GOHVAN_MLA_781	SH Design Requirements Rural	per site	R17 000,00	1	R 17 000,00
11			BUILD PACKAGES				R 19 100,00
12	304365678	GOHVAN_MLA_657	GF High Security Day shift - Non Std	shifts	R900,00	1	R 900,00
13	304365679	GOHVAN_MLA_658	GF High Security Night shift - Non Std	shifts	R1 000,00	1	R 1 000,00
14	304365680	GOHVAN_MLA_659	GF Std Security day shift - Non Standard	shifts	R200,00	1	R 200,00

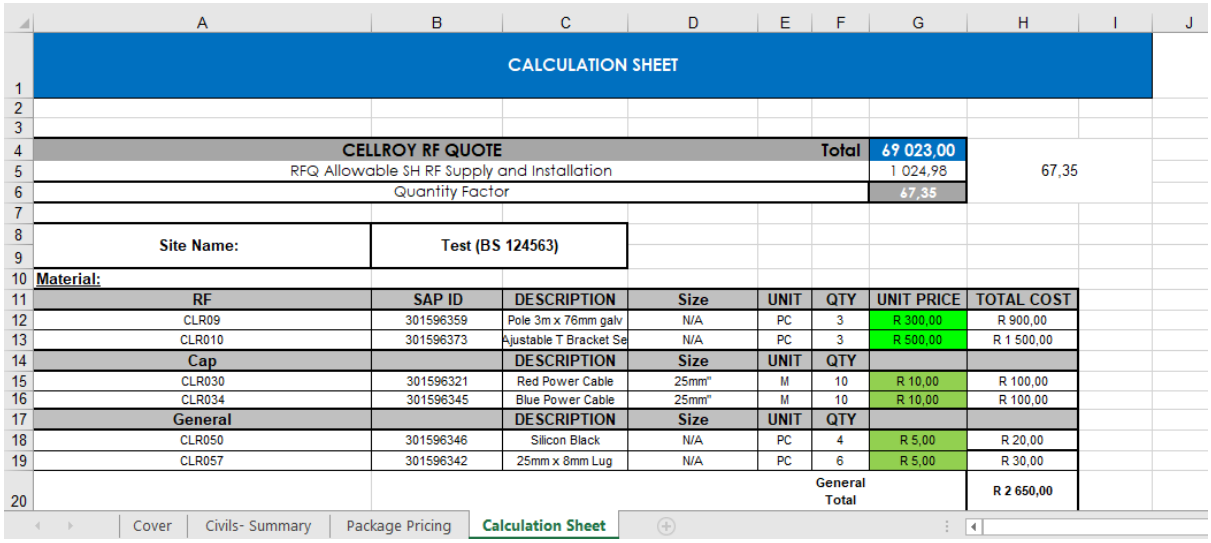
Cover | Civils- Summary | **Package Pricing** | Calculation Sheet

Figure 257: 2. Package Pricing Sheet – 1

	A	B	C	D	E	F	G
5			PROGRAM MANAGEMENT				R 3 000,00
6	304360789	GOHVAN_MLA_657	SH PM Shared Sites Metro	per site	R1 000,00	1	R 1 000,00
7	304360789	GOHVAN_MLA_678	SH PM Shared Rural	per site	R2 000,00	1	R 2 000,00
8			DESIGN PACKAGES				R 27 000,00
9	304366789	GOHVAN_MLA_780	SH Design Requirements Metro	per site	R10 000,00	1	R 10 000,00
10	304366789	GOHVAN_MLA_781	SH Design Requirements Rural	per site	R17 000,00	1	R 17 000,00
11			BUILD PACKAGES				R 19 100,00
12	304365678	GOHVAN_MLA_657	GF High Security Day shift - Non Std	shifts	R900,00	1	R 900,00
13	304365679	GOHVAN_MLA_658	GF High Security Night shift - Non Std	shifts	R1 000,00	1	R 1 000,00
14	304365680	GOHVAN_MLA_659	GF Std Security day shift - Non Standard	shifts	R200,00	1	R 200,00
15	304365681	GOHVAN_MLA_660	GF Std Security Night shift - Non Std	shifts	R3 000,00	1	R 3 000,00
16	304365682	GOHVAN_MLA_661	SH Site Prep Clear&Filling Site Ext	per site	R8 000,00	1	R 8 000,00
17	304365683	GOHVAN_MLA_662	SH Site Prep Clear & Fill Site Ext	m²	R6 000,00	1	R 6 000,00
18							

Cover | Civils- Summary | **Package Pricing** | Calculation Sheet

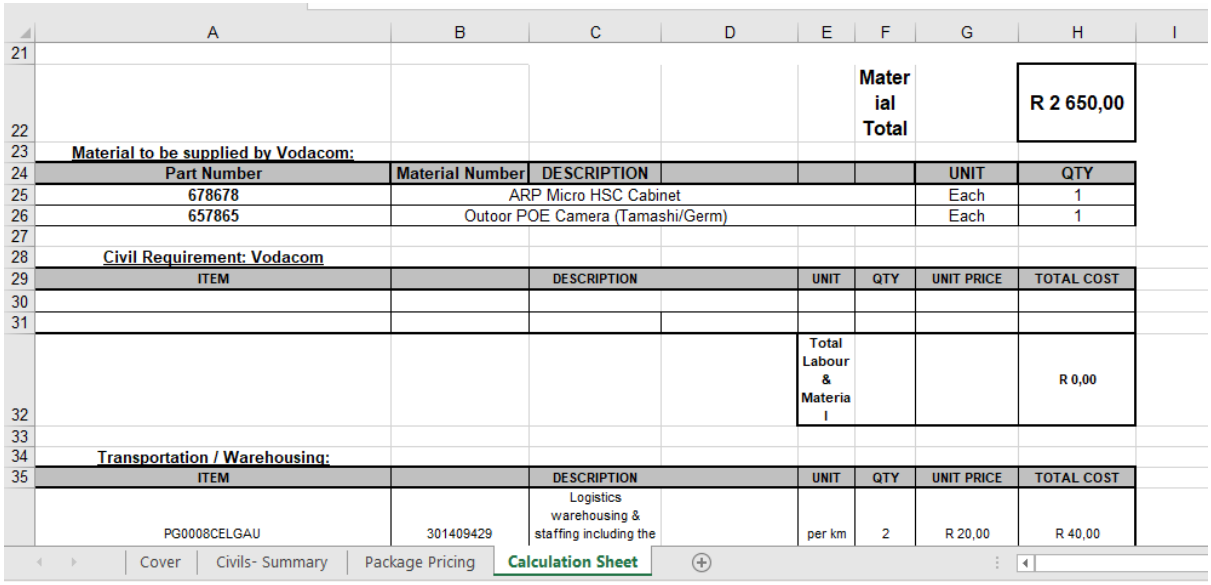
Figure 258: 2. Package Pricing Sheet – 2



	CALCULATION SHEET							
	CELLROY RF QUOTE					Total	69 023,00	
	RFQ Allowable SH RF Supply and Installation						1 024,98	67,35
	Quantity Factor						67,35	
	Site Name:	Test (BS 124563)						
	Material:							
	RF	SAP ID	DESCRIPTION	Size	UNIT	QTY	UNIT PRICE	TOTAL COST
	CLR09	301596359	Pole 3m x 76mm galv	N/A	PC	3	R 300,00	R 900,00
	CLR010	301596373	Ajustable T Bracket Se	N/A	PC	3	R 500,00	R 1 500,00
	Cap		DESCRIPTION	Size	UNIT	QTY		
	CLR030	301596321	Red Power Cable	25mm"	M	10	R 10,00	R 100,00
	CLR034	301596345	Blue Power Cable	25mm"	M	10	R 10,00	R 100,00
	General		DESCRIPTION	Size	UNIT	QTY		
	CLR050	301596346	Silicon Black	N/A	PC	4	R 5,00	R 20,00
	CLR057	301596342	25mm x 8mm Lug	N/A	PC	6	R 5,00	R 30,00
						General Total		R 2 650,00

Cover | Civils- Summary | Package Pricing | Calculation Sheet

Figure 259: 2. Calculation Sheet Sheet – 1



					Material Total		R 2 650,00
Material to be supplied by Vodacom:							
Part Number	Material Number	DESCRIPTION				UNIT	QTY
678678	ARP Micro HSC Cabinet					Each	1
657865	Outoor POE Camera (Tamashi/Germ)					Each	1
Civil Requirement: Vodacom							
ITEM		DESCRIPTION		UNIT	QTY	UNIT PRICE	TOTAL COST
				Total Labour & Material			R 0,00
Transportation / Warehousing:							
ITEM		DESCRIPTION		UNIT	QTY	UNIT PRICE	TOTAL COST
PG0008CELGAU	301409429	Logistics warehousing & staffing including the		per km	2	R 20,00	R 40,00

Cover | Civils- Summary | Package Pricing | Calculation Sheet

Figure 260: 2. Calculation Sheet Sheet – 2

	A	B	C	D	E	F	G	H	I
33									
34	**Transportation / Warehousing:**								
35	**ITEM**		**DESCRIPTION**		**UNIT**	**QTY**	**UNIT PRICE**	**TOTAL COST**	
36	PG0008CELGAU	301409429	Logistics warehousing & staffing including the collection & delivery of materials to site		per km	2	R 20,00	R 40,00	
37	Non Catalogue Item	N/A	Abbott to Collect ARP 1 Door Micro H&S		PC	1	R 4 000,00	R 4 000,00	
38	CLR114	304290986	Warehousing & NSS		PC	1	R 300,00	R 330,00	
39				**Total Transport**				**R 4 370,00**	
40									
41	**Labour:**								
42	**ITEM**		**DESCRIPTION**		**UNIT**	**QTY**	**UNIT PRICE**	**TOTAL COST**	
43	BW0069LCELGAU	304290832	Site establishment		Item	1	R 3 000,00	R 3 000,00	
44	CLR115	304290994	Site survey		Item	1	R 50 000,00	R 50 000,00	
45				**Total Labour**				**R 53 000,00**	
46									
47									
48	**Summary of cost**							**Amount**	
49	1. Material							R 2 650,00	
50	2. Civil Requirement							R 0,00	
51	3. Transportation / Warehousing							R 4 370,00	
52	4. Labour							R 53 000,00	

Cover | Civils- Summary | Package Pricing | **Calculation Sheet**

Figure 261: 2. Calculation Sheet Sheet – 3

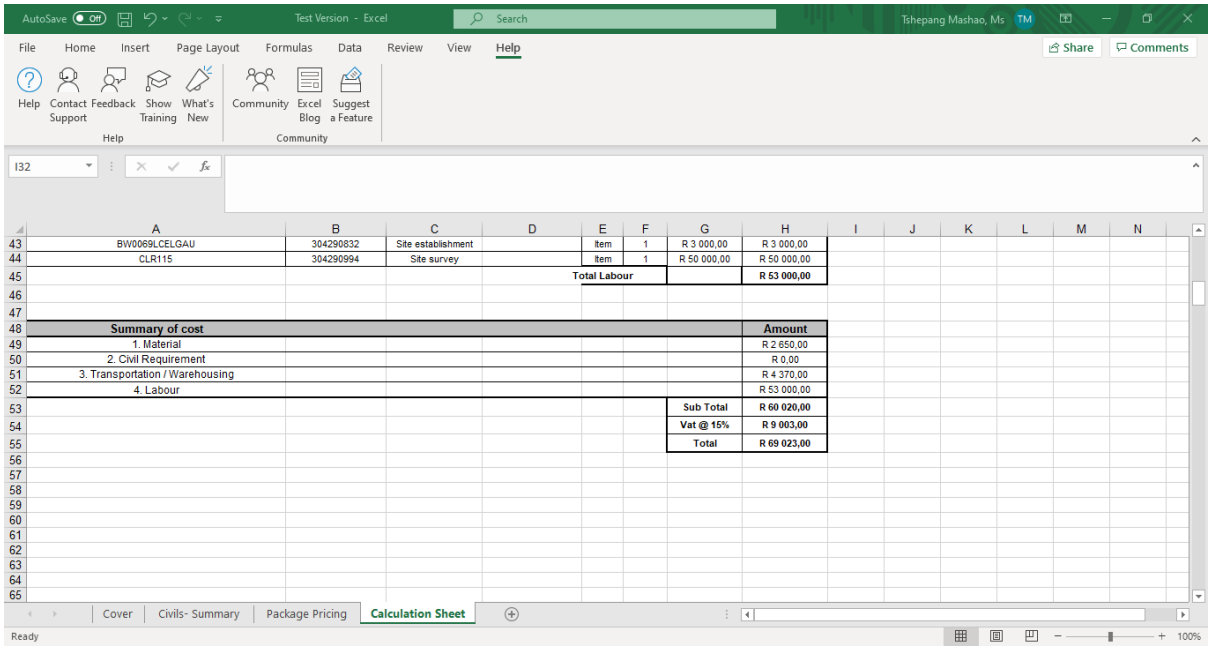


Figure 262: 2. Calculation Sheet Sheet – 4

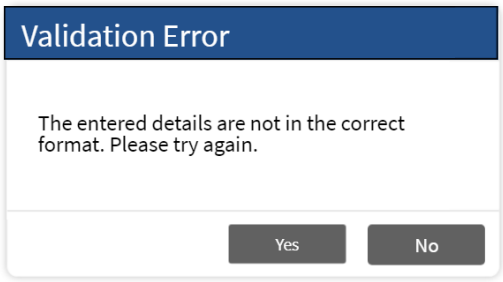


Figure 263: 2. Validation Error Dialog

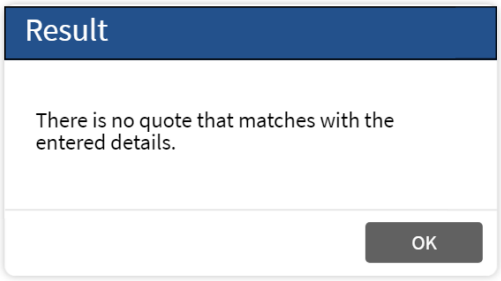


Figure 264: 2. No quote found dialog

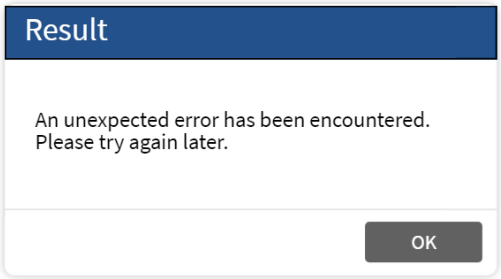


Figure 265: 2. Unexpected Error Dialog

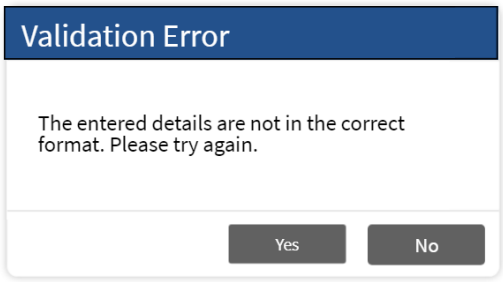


Figure 266: 2. Successful Purchase Order Upload Dialog

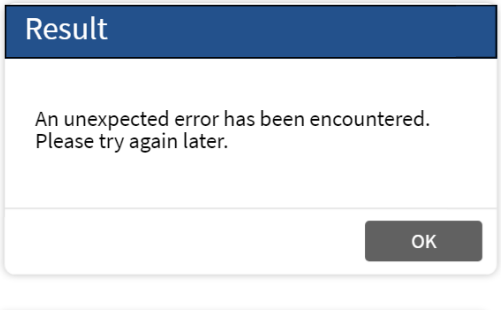


Figure 267: 2. Unexpected Error Dialog

6.2.3 Administration Subsystem

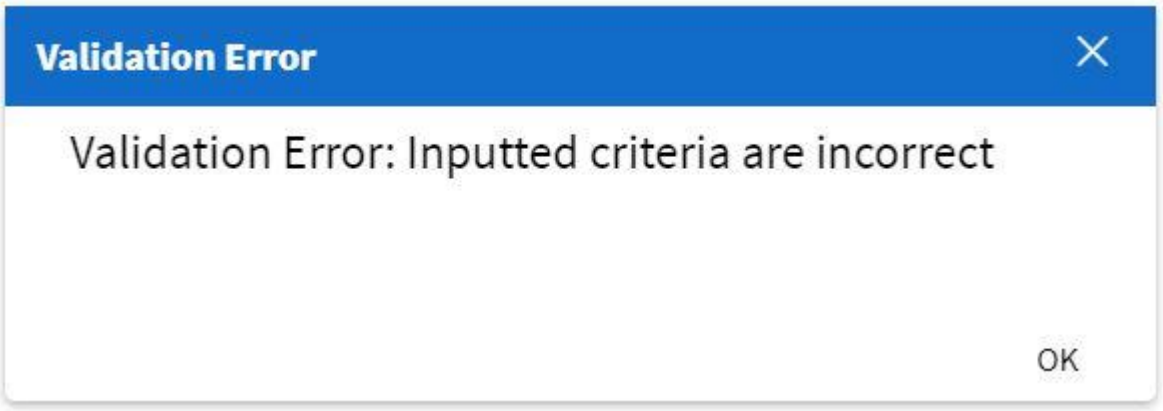


Figure 268: 3. Validation error message modal

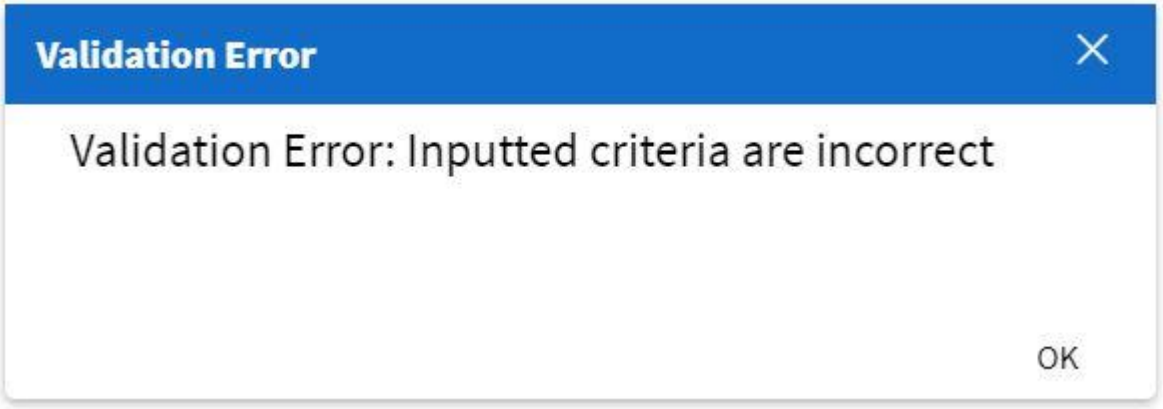


Figure 269: 3. Validation error message modal



Figure 270: 3. Validation error message modal



Figure 271: 3. Validation error message modal



Figure 272: 3. Validation error message modal



Figure 273: 3. Validation error message modal



Figure 274: 3. Validation error message modal



Figure 275: 3. Validation error message modal

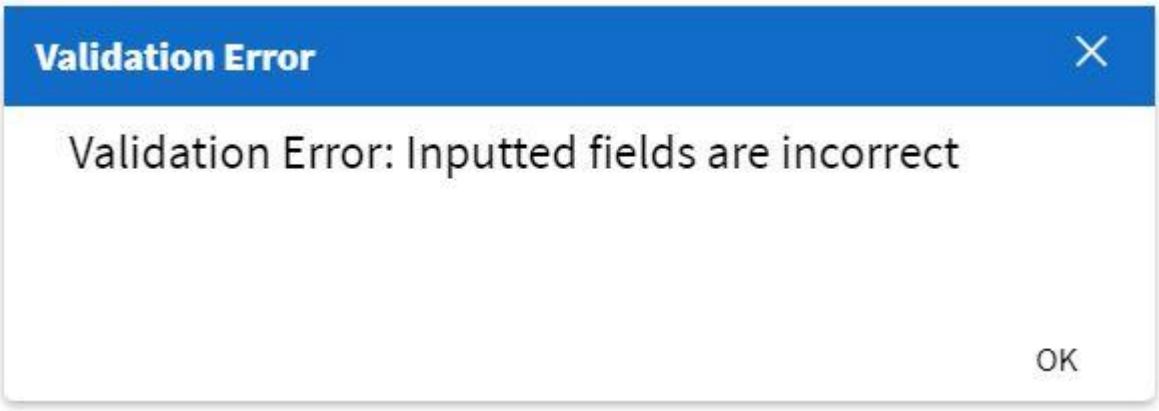


Figure 276: 3. Validation error message modal

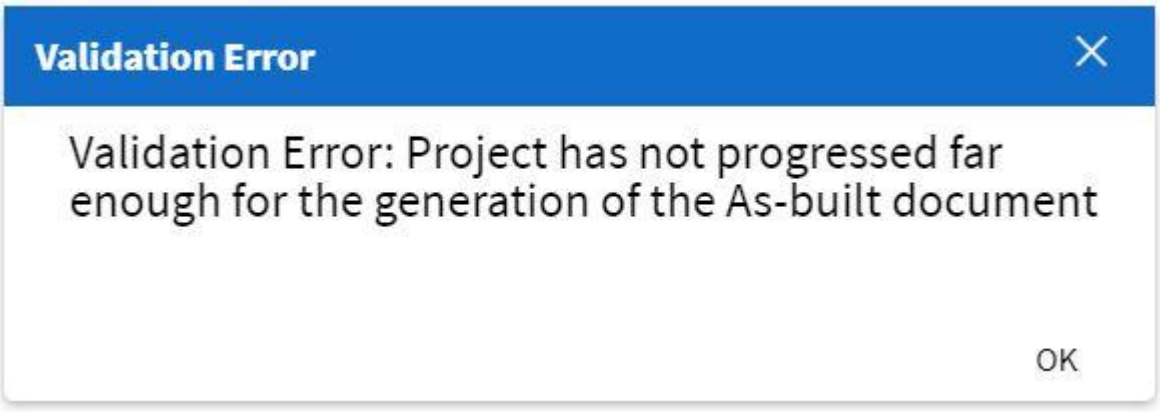


Figure 277: 3. Validation error project status modal



Figure 278: 3. Validation error message modal

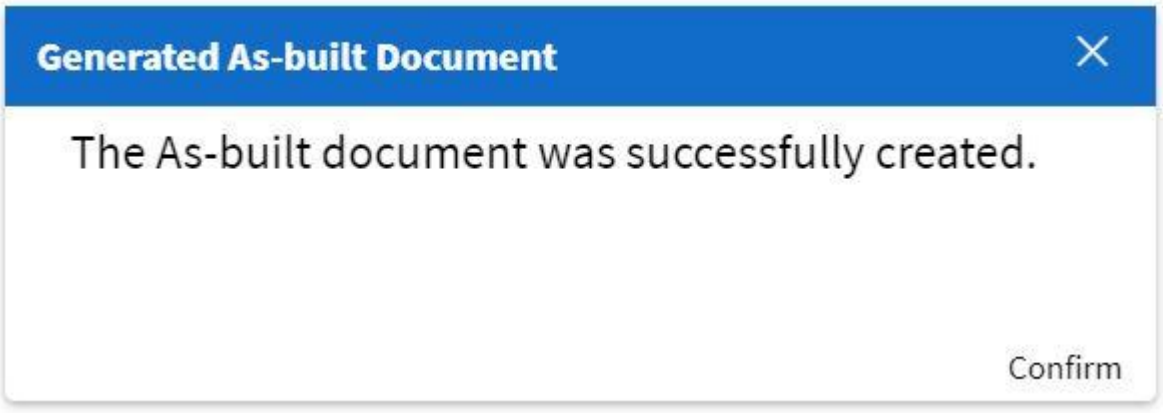


Figure 279: 3. Generated As-built document modal

6.2.4 Construction Subsystem

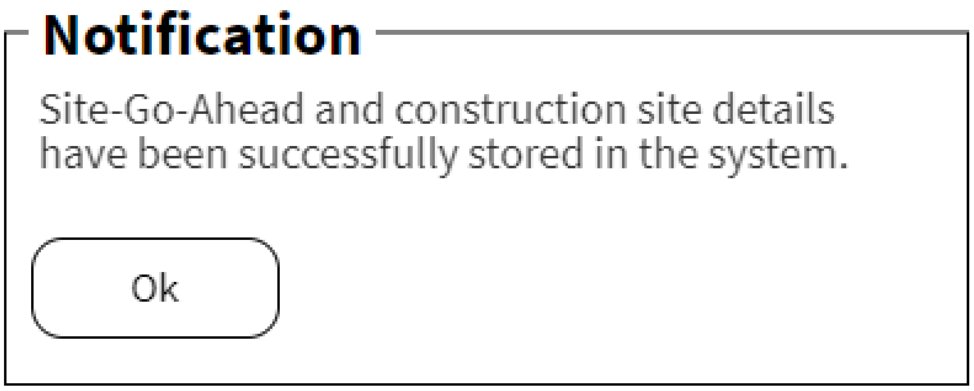


Figure 280: 4. Success Notification

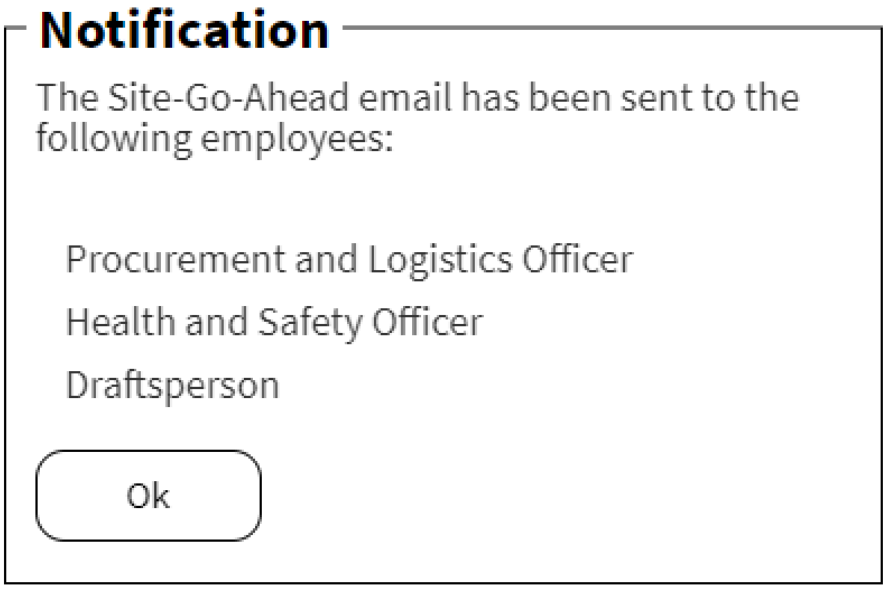


Figure 281: 4. Emailed Notification

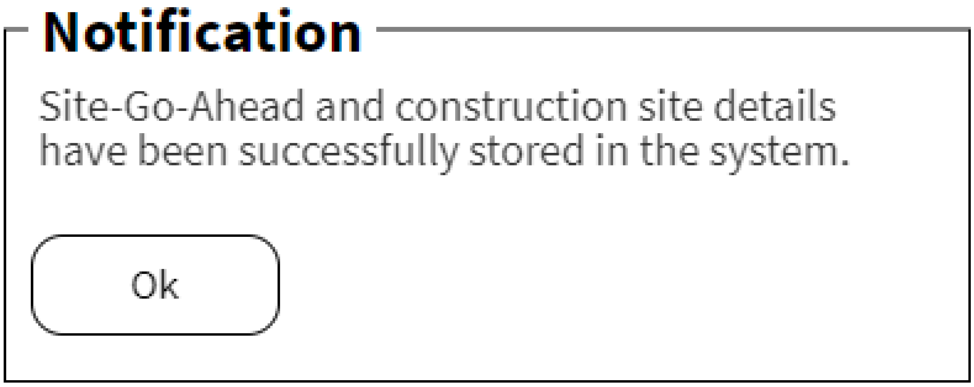


Figure 282: 4. Updated Construction site notification

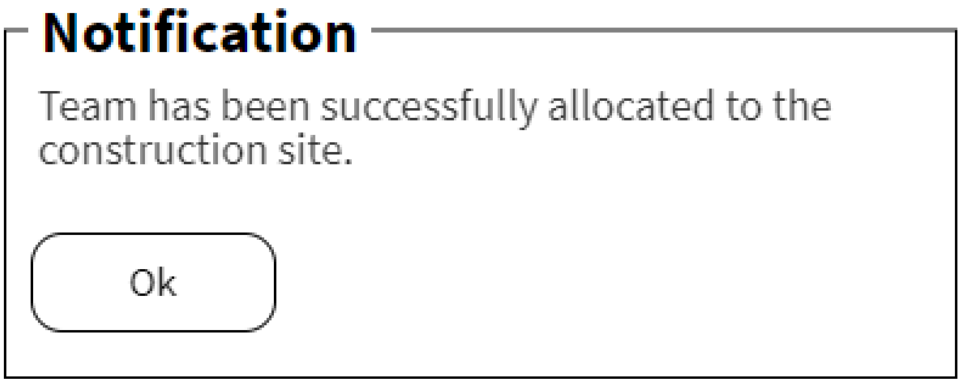


Figure 283: 4. Successfully allocated team notification

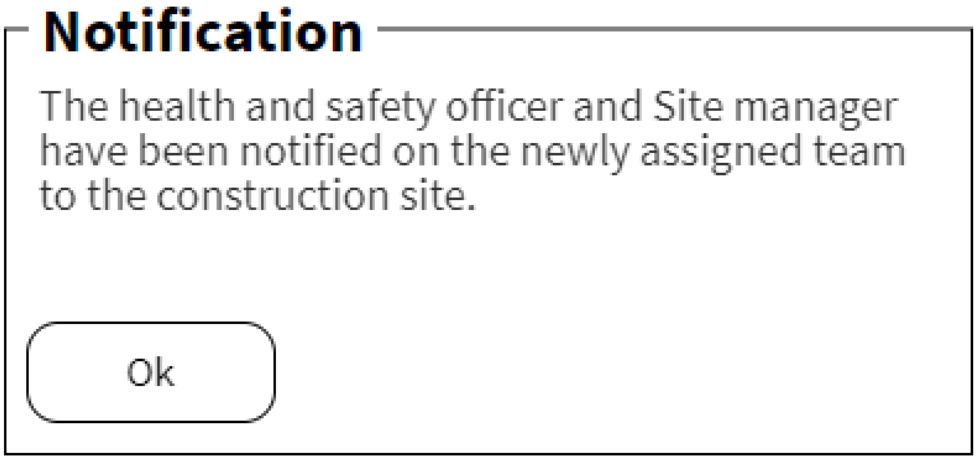


Figure 284: 4. Successfully updated team notification

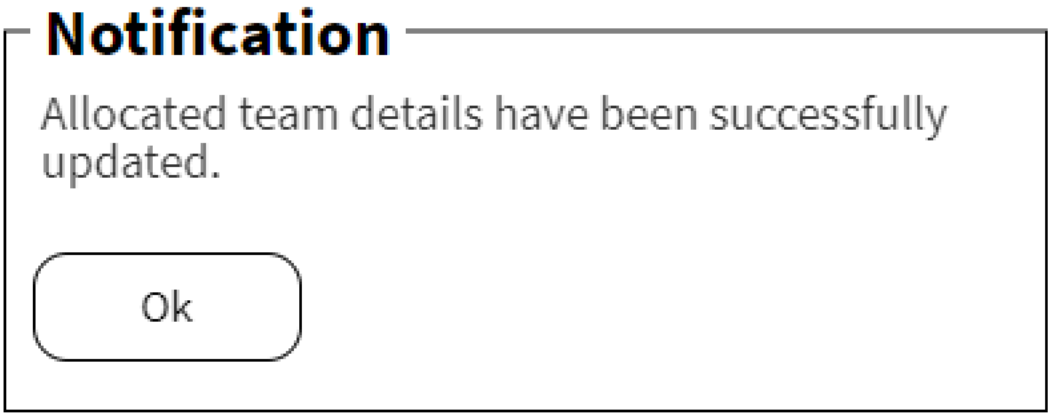


Figure 285: 4. Successfully updated team notification

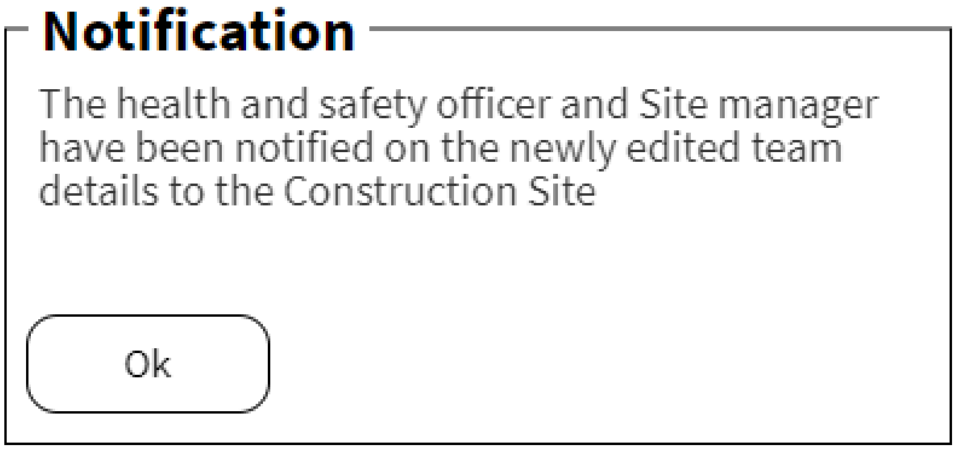


Figure 286: 4. Successfully notified employees notification

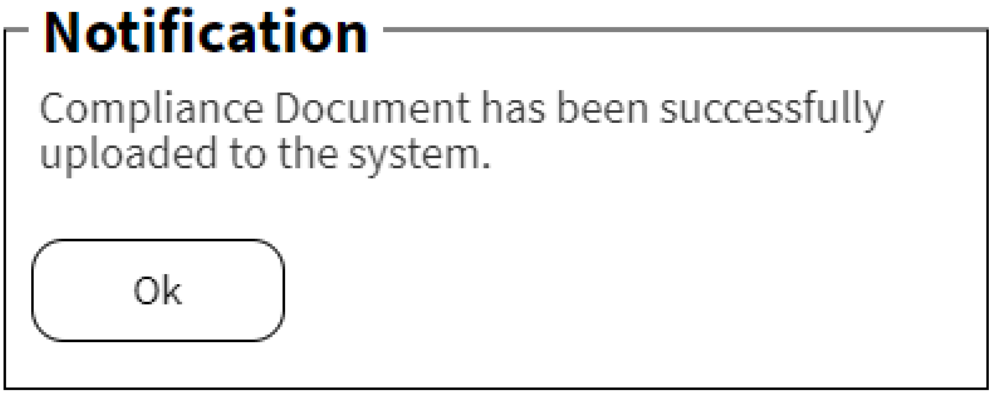


Figure 287: 4. Successfully upload compliance document notification

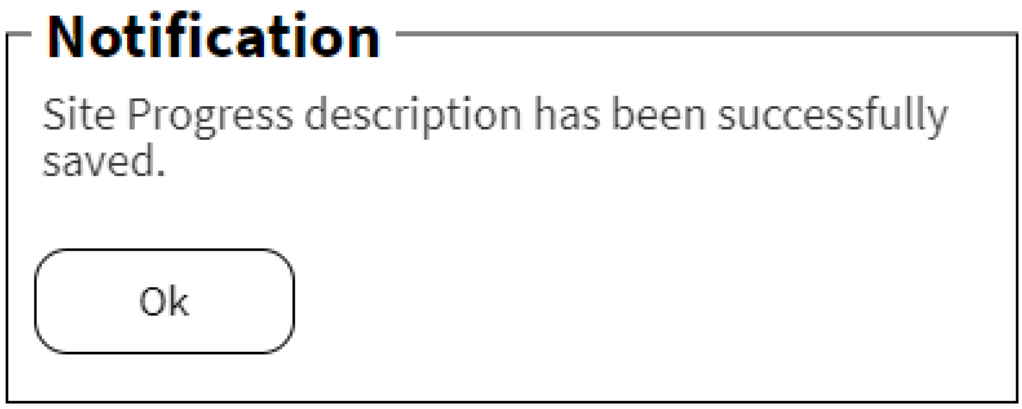


Figure 288: 4. Successfully captured notification

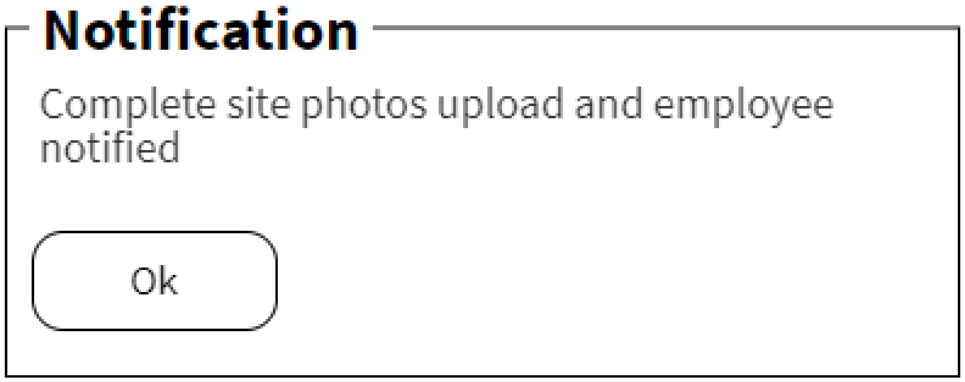


Figure 289: 4. Successfully upload complete site photos notification



Figure 290: 4. Successfully uploaded Radio Frequency report notification

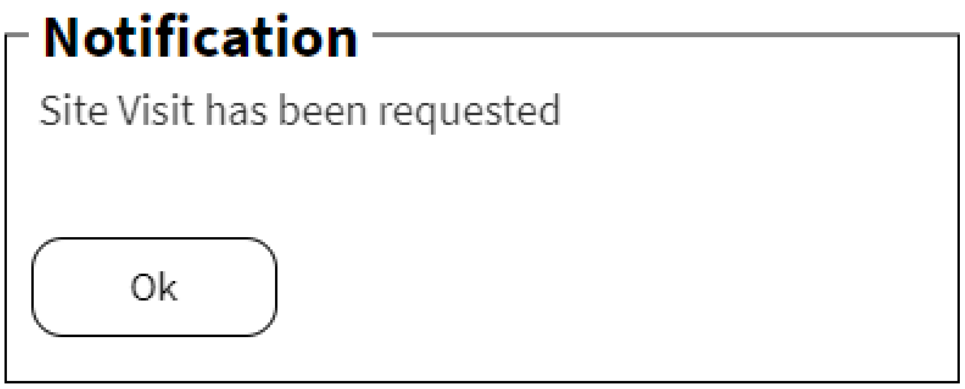


Figure 291: 4. Successfully requested site visit notification

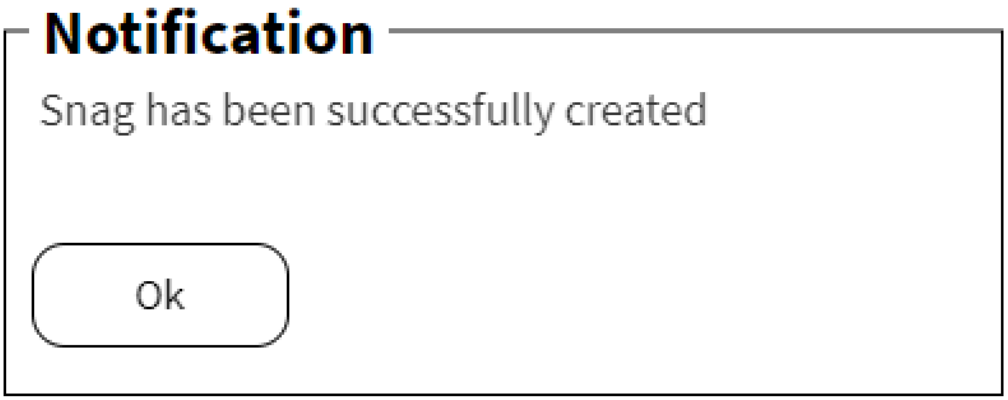


Figure 292: 4. Successfully created snag notification

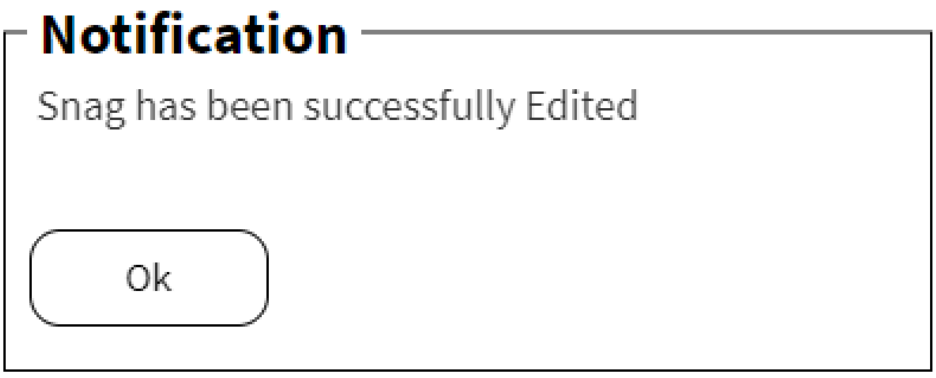


Figure 293: 4. Successfully edit snag notification

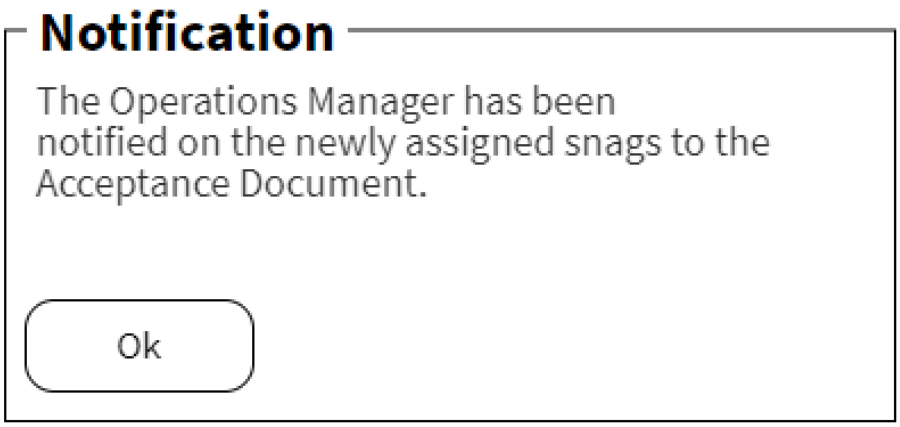


Figure 294: 4. Successfully updated acceptance document notification

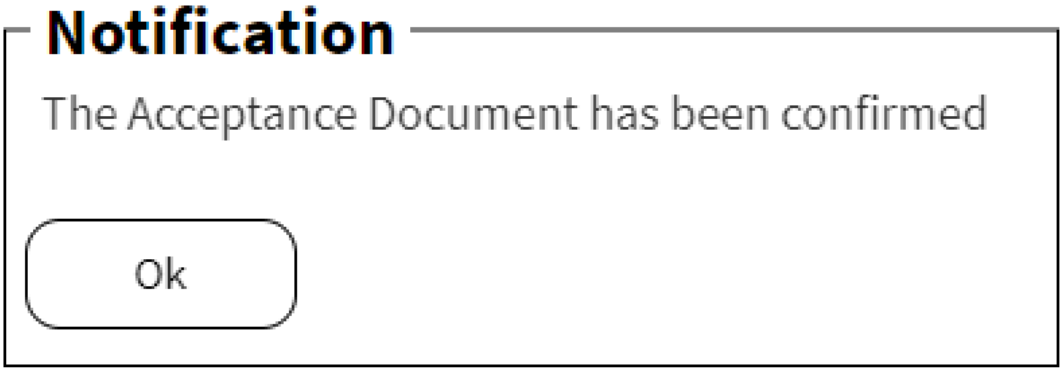


Figure 295: 4. Confirmed Acceptance document notification

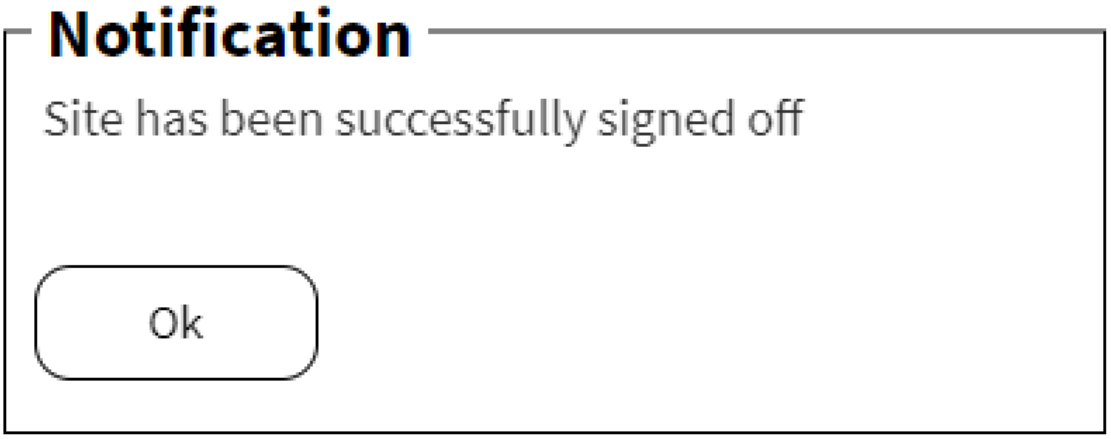


Figure 296: 4. Successfully emailed employees notification

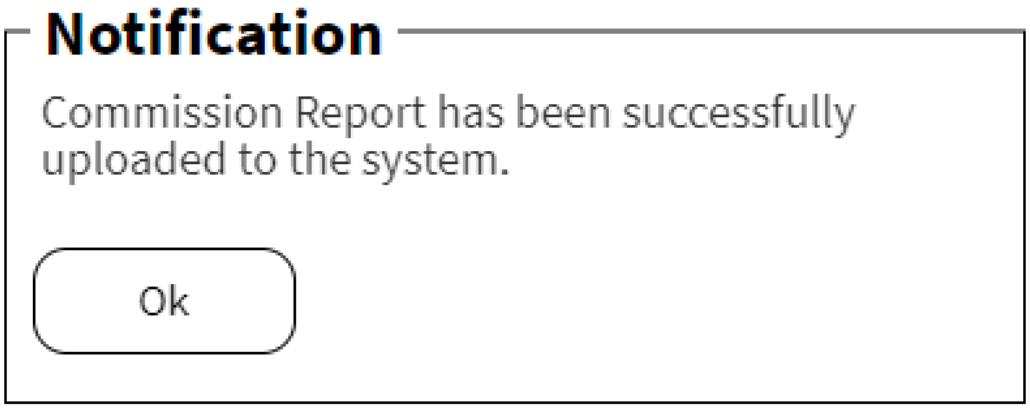


Figure 297: 4. Successfully uploaded Commission report page

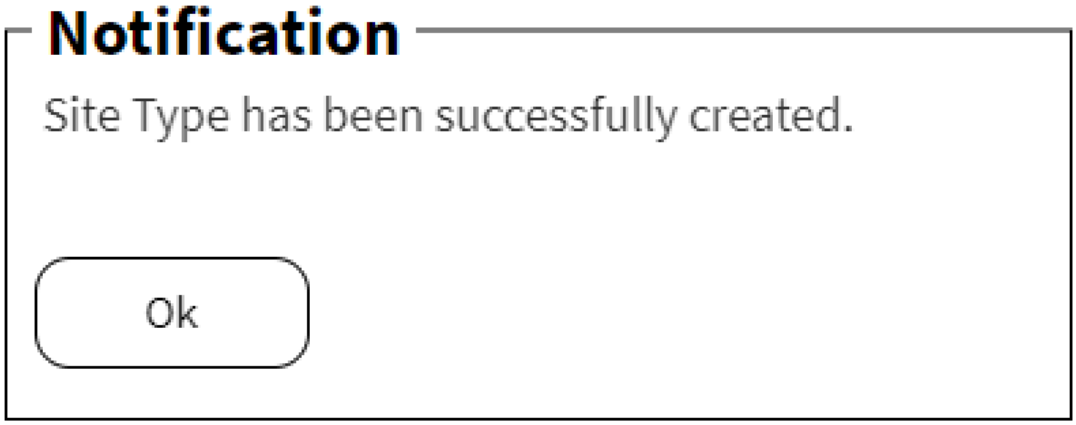


Figure 298: 4. Successfully create site type notification

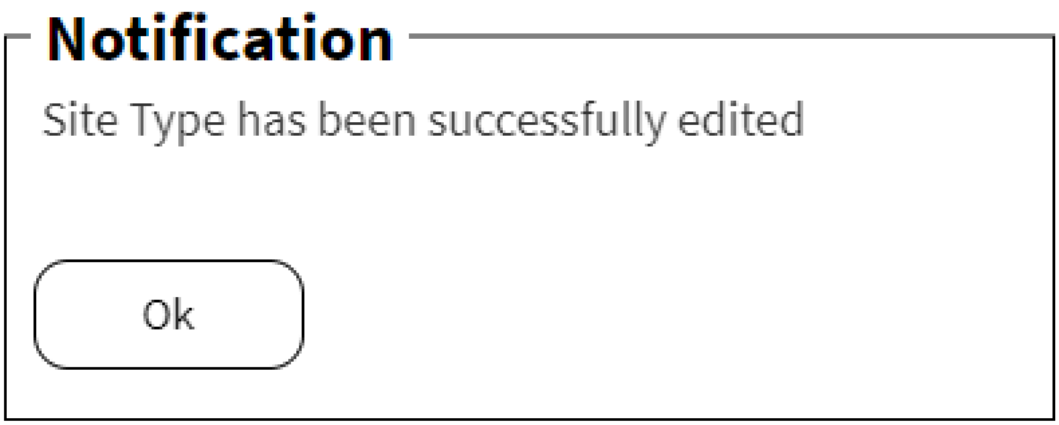


Figure 299: 4. Successfully edited site type notification

6.2.5 Invoicing Subsystem

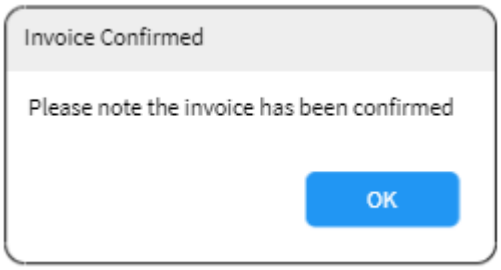


Figure 300: 5. Invoice Confirmed Alert

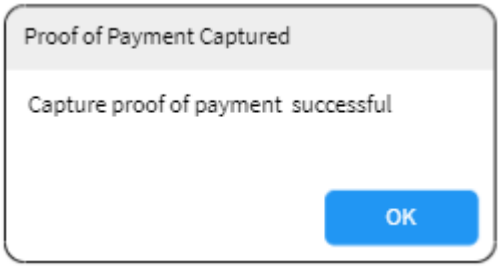


Figure 301: 5. Proof of Payment captured alert

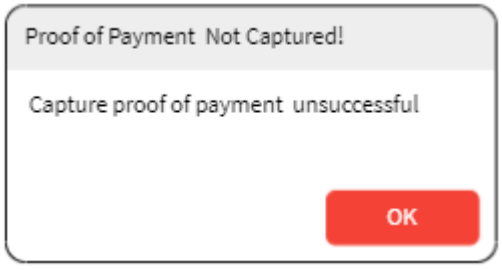


Figure 302: 5. Validation Error Message Alert

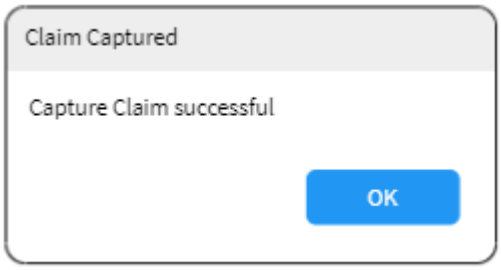


Figure 303: 5. Claim Captured Alert

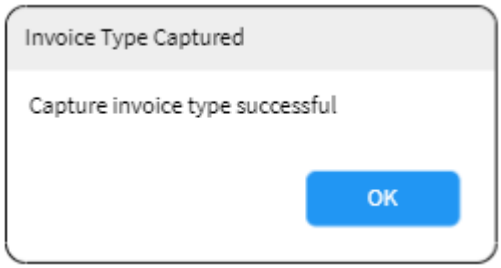


Figure 304: 5. Claim Captured Alert

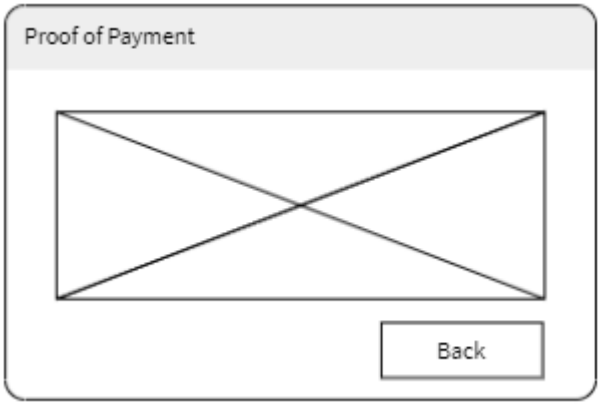


Figure 305: 5. View Proof of Payment Modal

6.2.6 Human Resource Subsystem

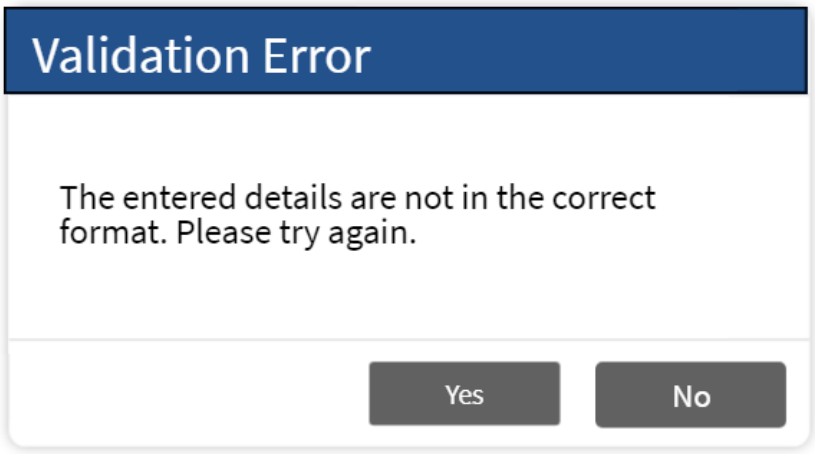


Figure 306: 6. Validation Error Dialog

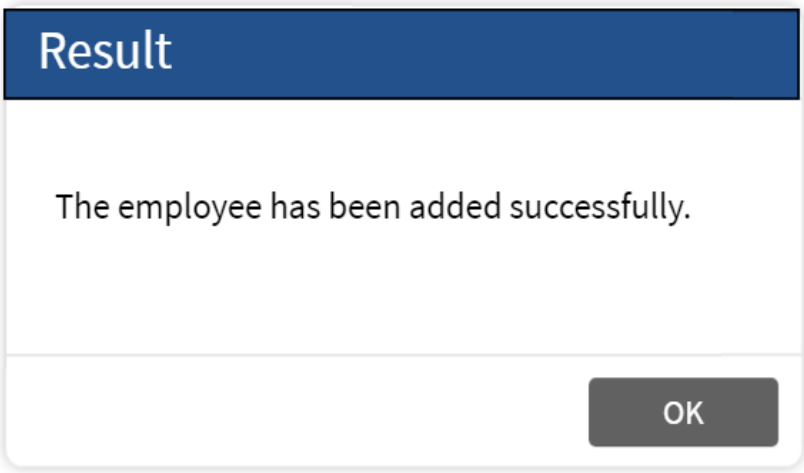


Figure 307: 6. Successful Employee Addition Modal

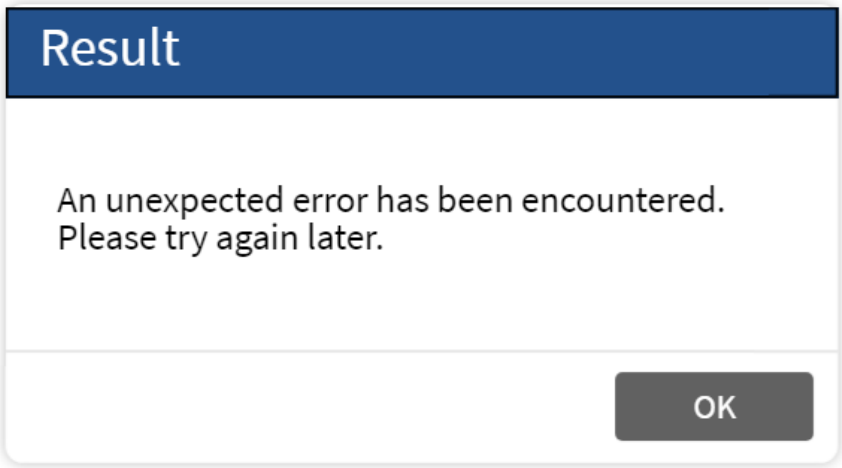


Figure 308: 6. Unexpected Error Dialog

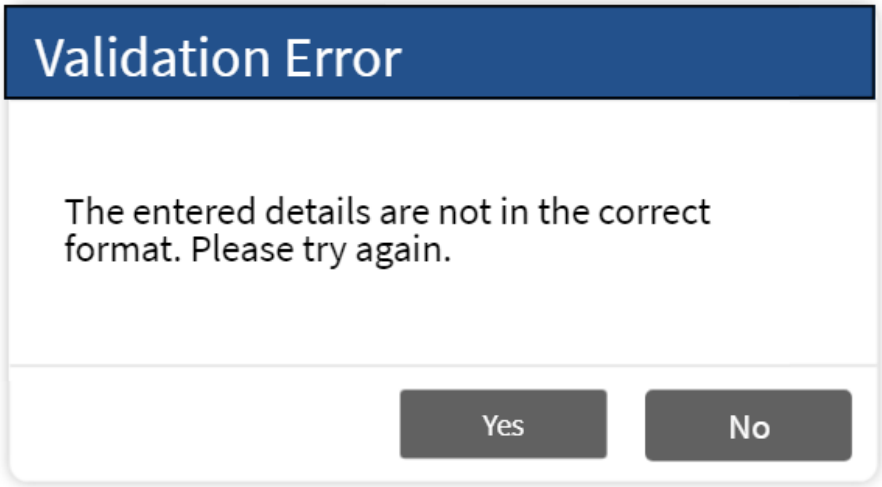


Figure 309: 6. Validation Error Dialog

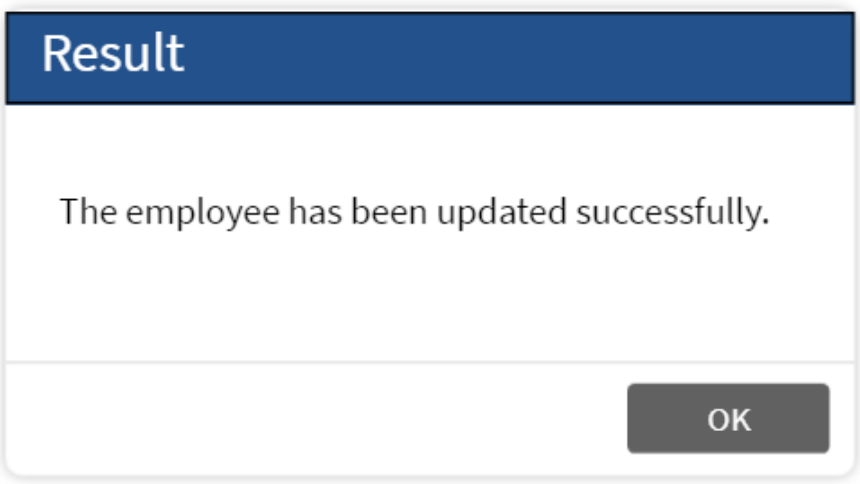


Figure 310: 6. Successful Employee Update Modal

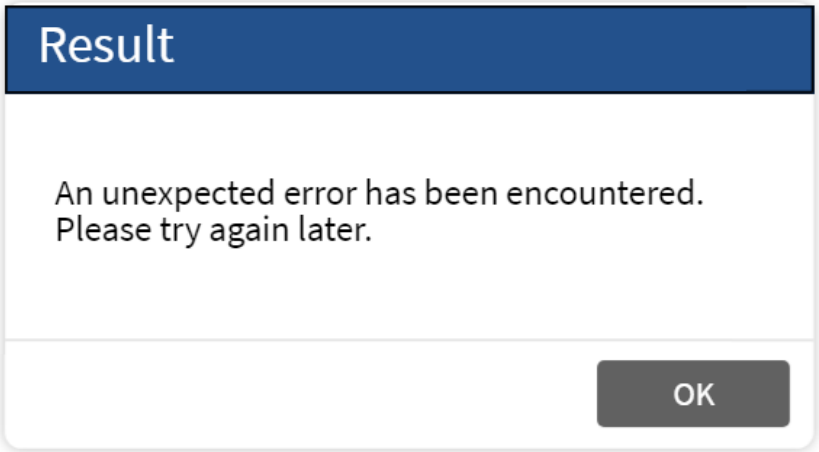


Figure 311: 6. Unexpected Error Dialog

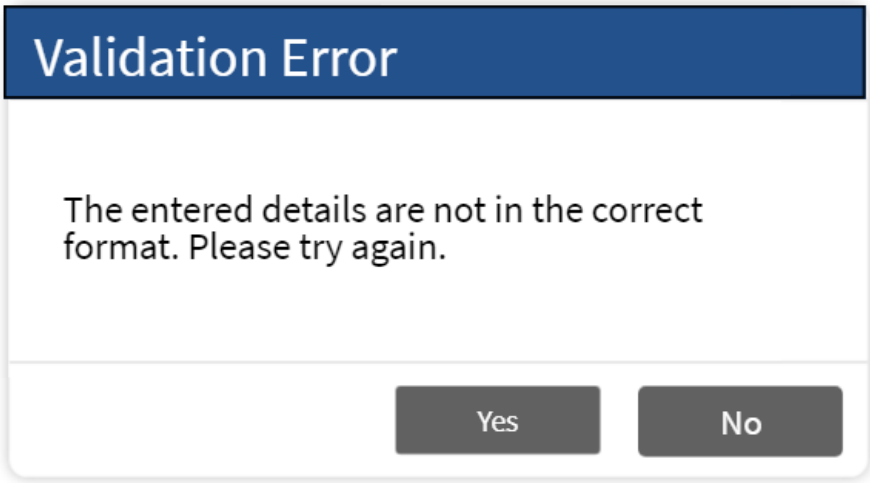


Figure 312: 6. Validation Error Dialog

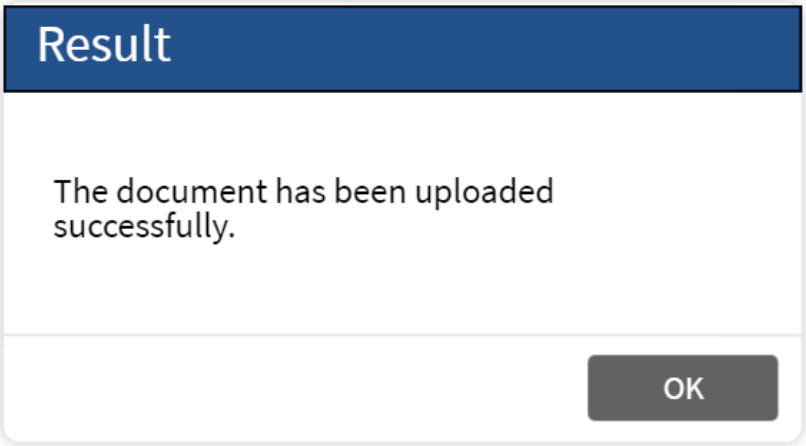


Figure 313: 6. Successful Document Upload Dialog

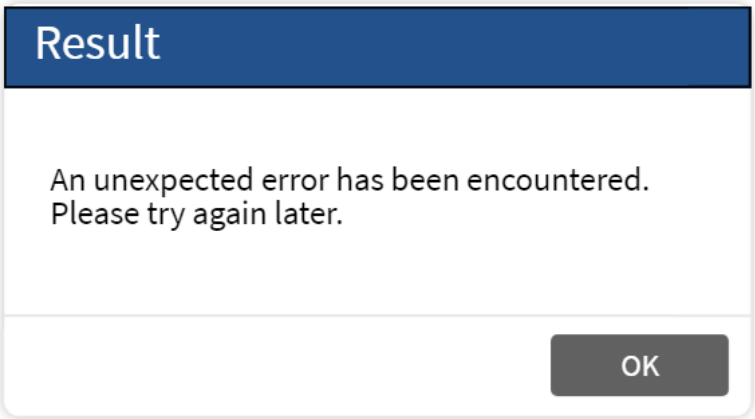


Figure 314: 6. Unexpected Error Dialog

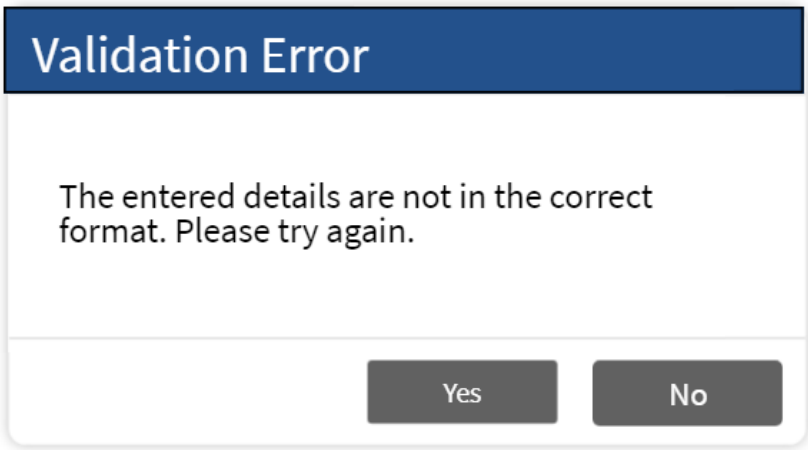


Figure 315: 6. Validation Error Dialog

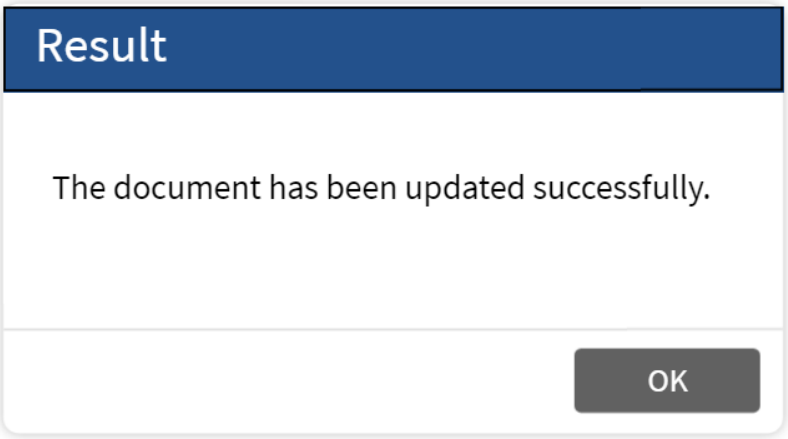


Figure 316: 6. Successful Document Update Dialog

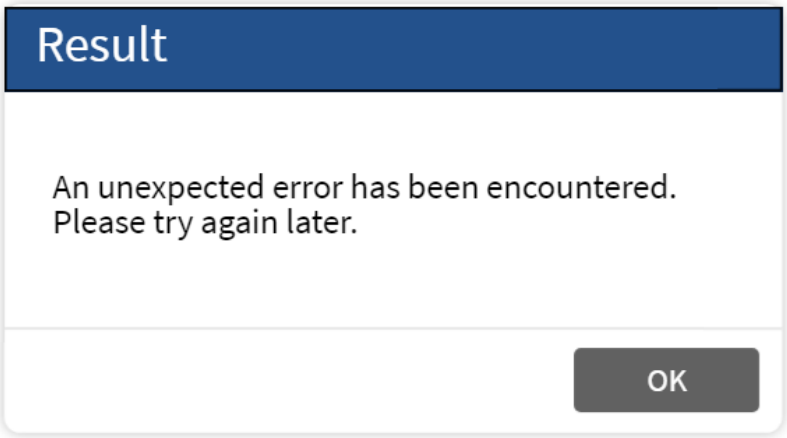


Figure 317: 6. Unexpected Error Dialog

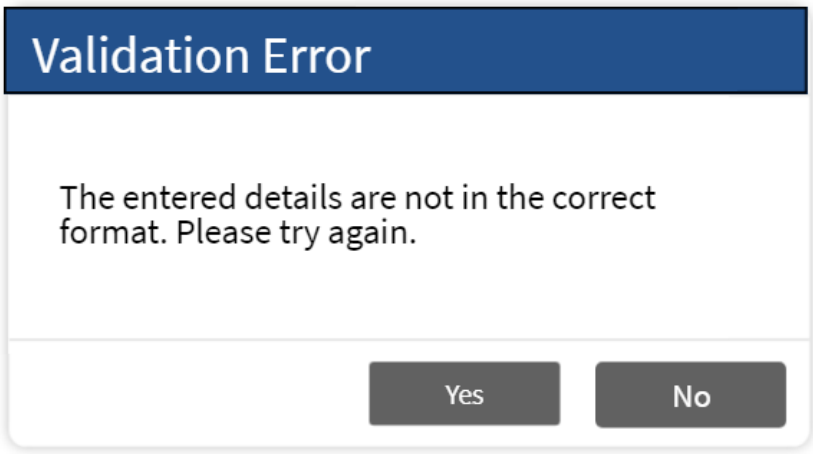


Figure 318: 6. Validation Error Dialog

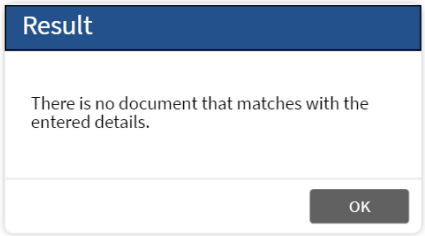


Figure 319: 6. No Document Found Dialog

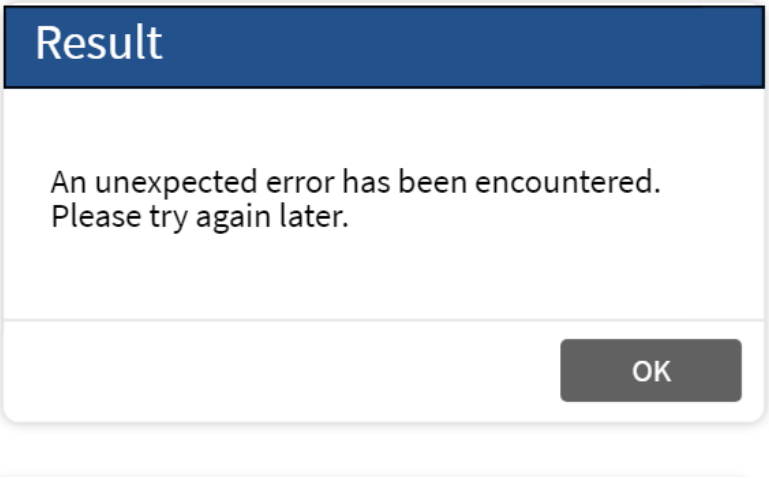


Figure 320: 6. Unexpected Error Dialog

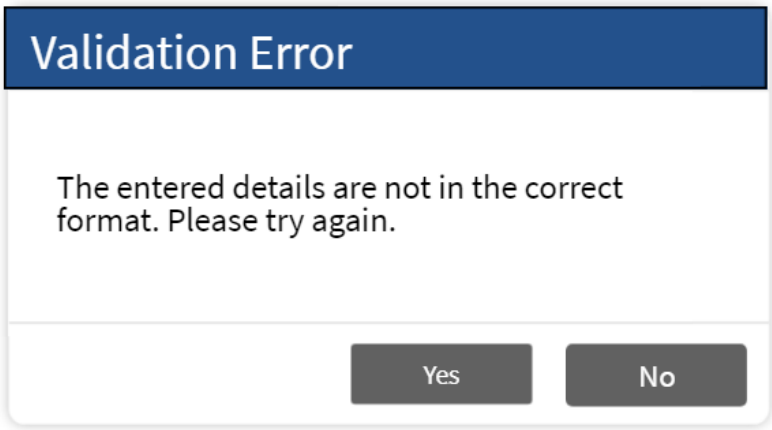


Figure 321: 6. Validation Error Dialog

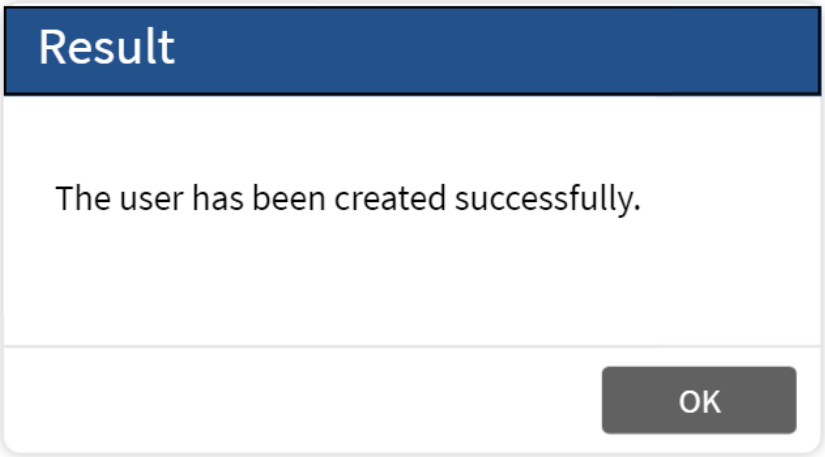


Figure 322: 6. Successful User Addition Dialog

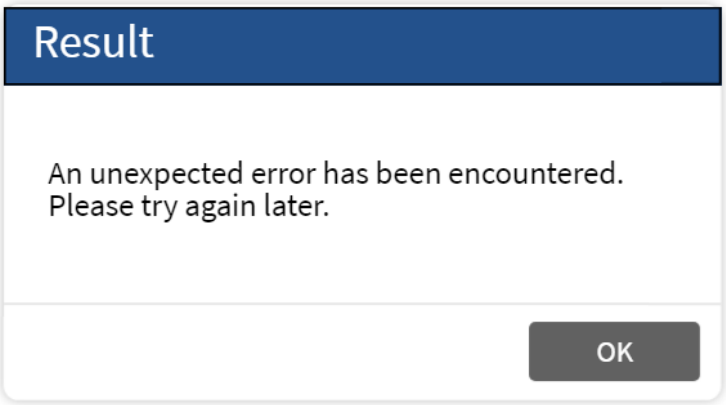


Figure 323: 6. Unexpected Error Dialog

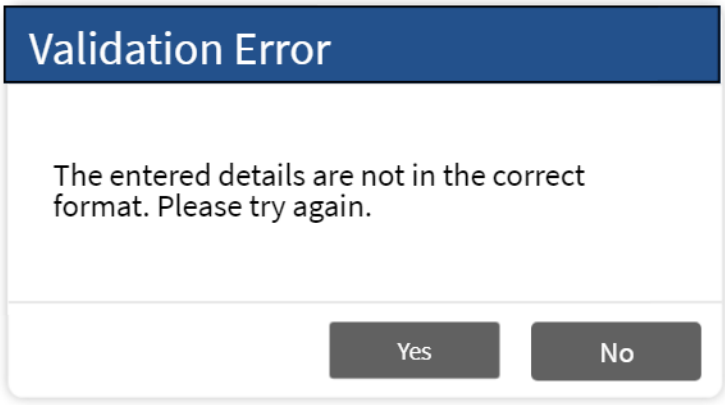


Figure 324: 6. Unexpected Error Dialog

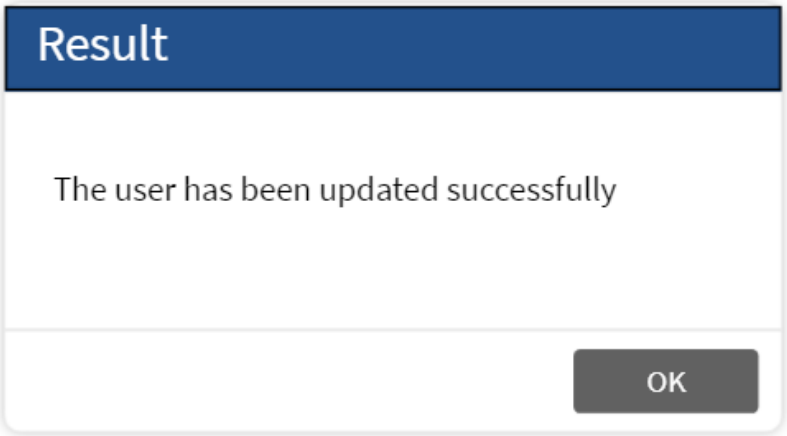


Figure 325: 6. Successful User Update Dialog

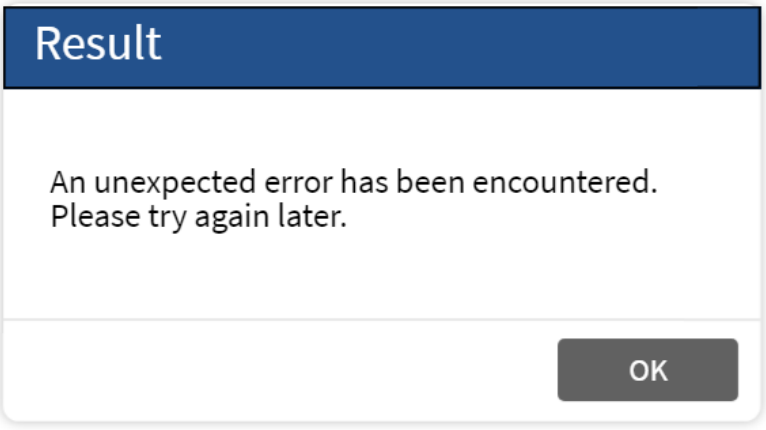


Figure 326: 6. Unexpected Error Dialog

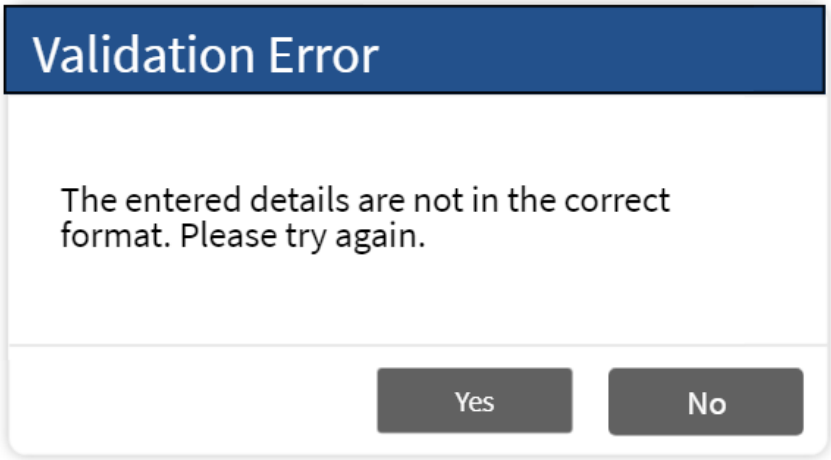


Figure 327: 6. Validation Error Dialog

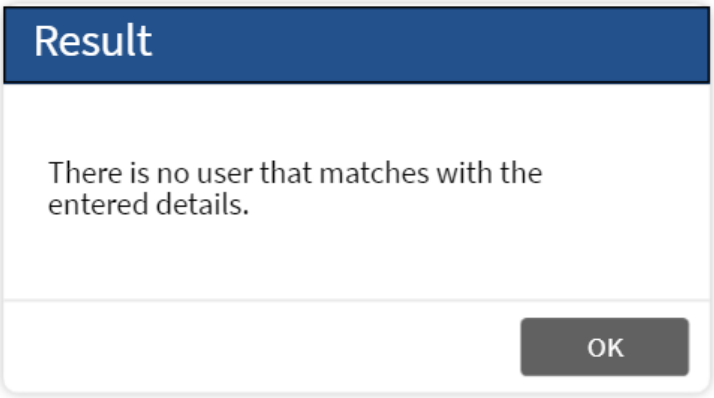


Figure 328: 6. No user found dialog

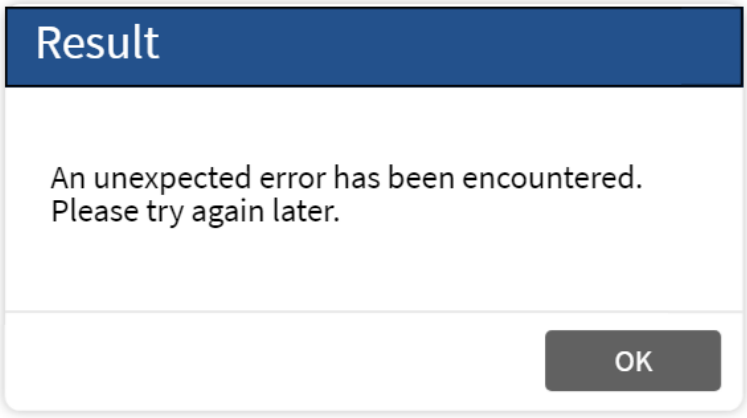


Figure 329: 6. Unexpected Error Dialog

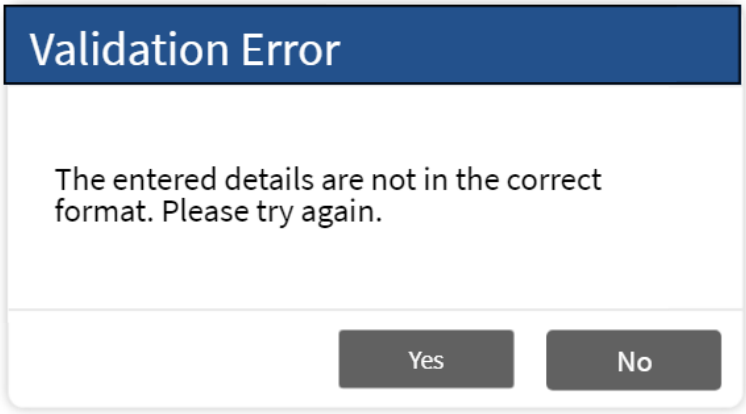


Figure 330: 6. Validation Error Dialog

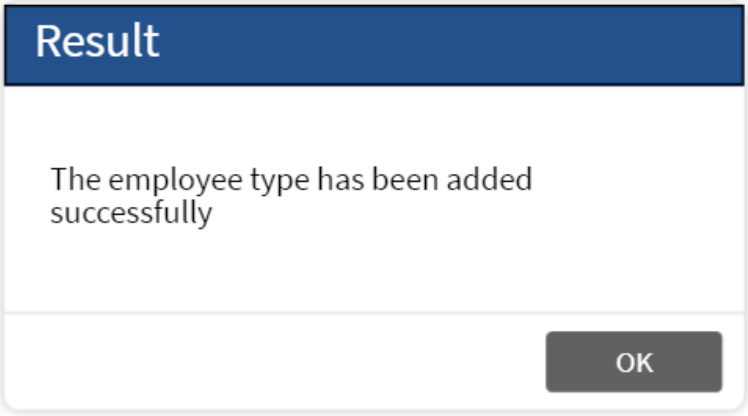


Figure 331: 6. Employee Type Creation Successful Modal

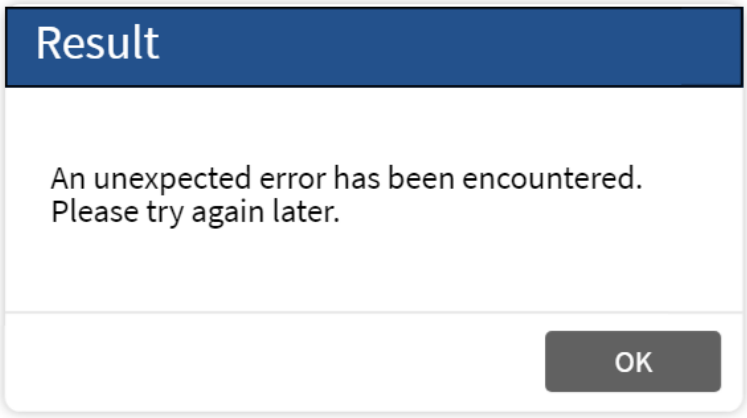


Figure 332: 6. Unexpected Error Dialog

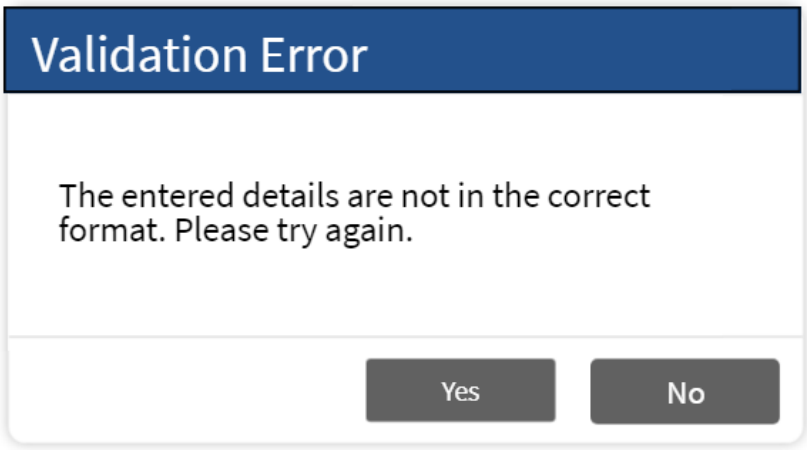


Figure 333: 6. Validation Error Dialog

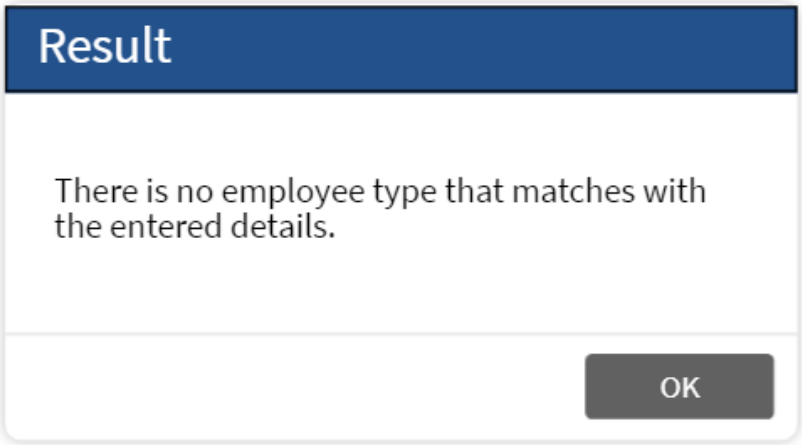


Figure 334: 6. No Employee Type Found Dialog

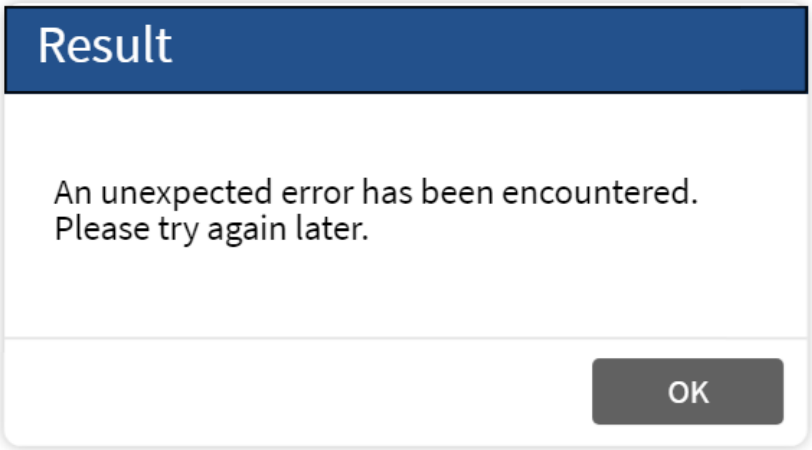


Figure 335: 6. Unexpected Error Dialog

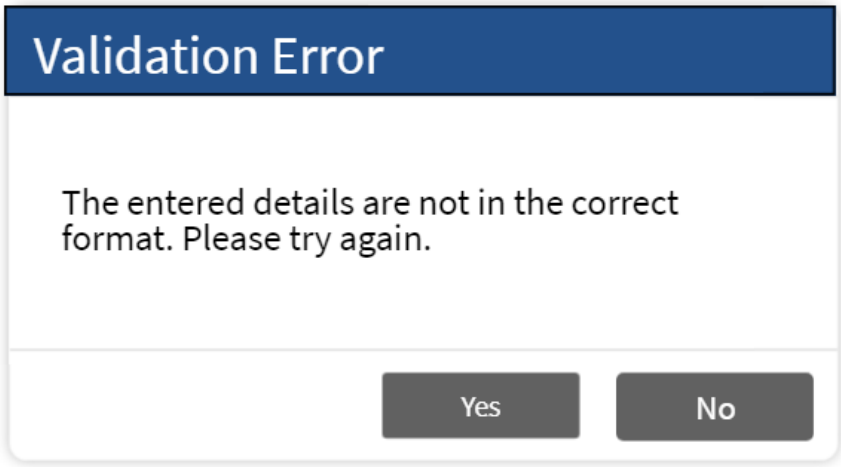


Figure 336: 6. Validation Error Dialog

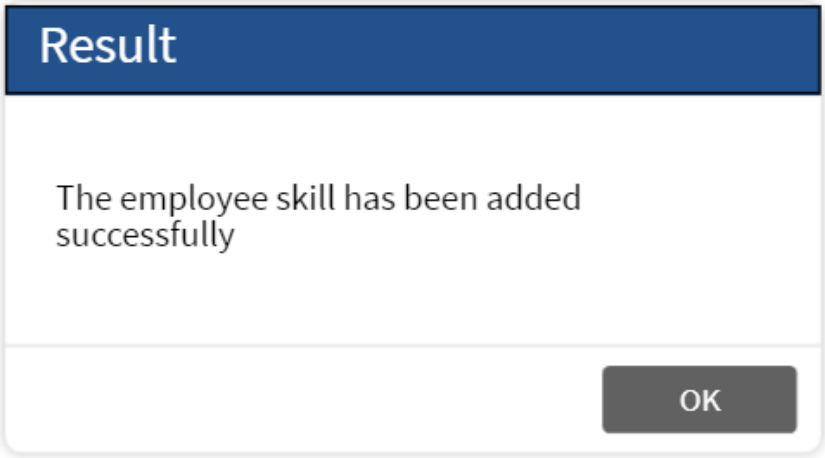


Figure 337: 6. Successful Employee Skill Creation

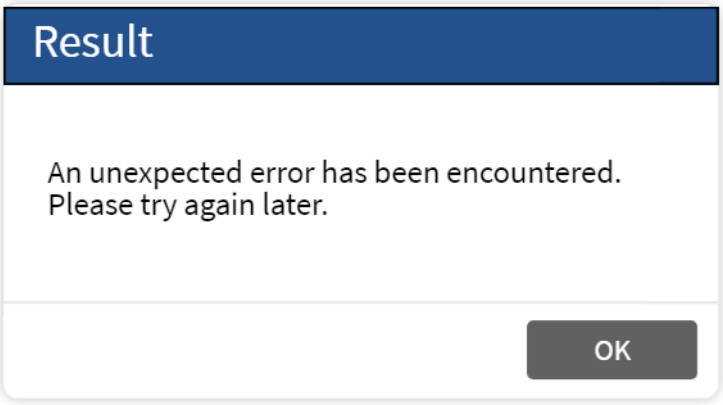


Figure 338: 6. Unexpected Error Dialog

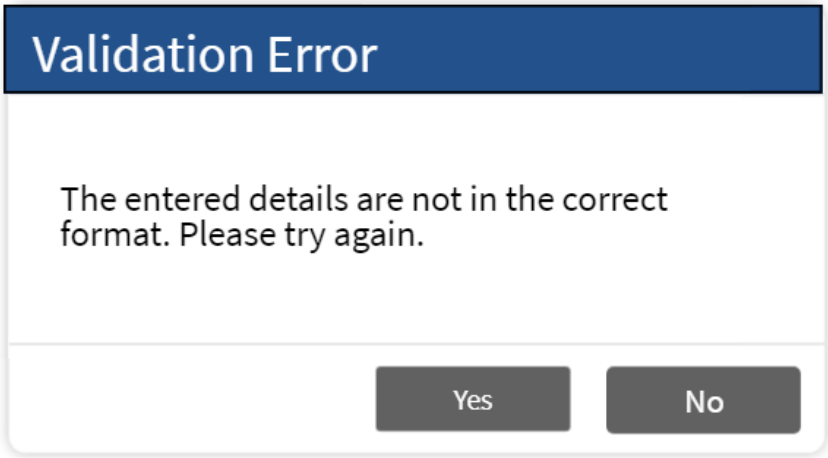


Figure 339: 6. Validation Error Dialog

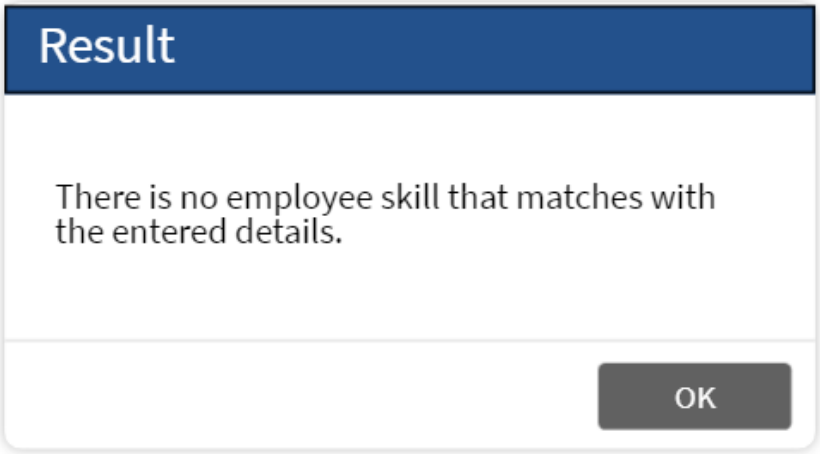


Figure 340: 6. No Employee Skill Found Dialog

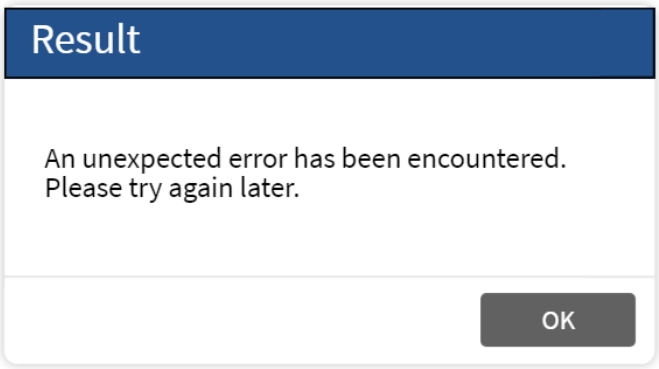


Figure 341: 6. Unexpected Error Dialog

6.2.7 Teams Subsystem

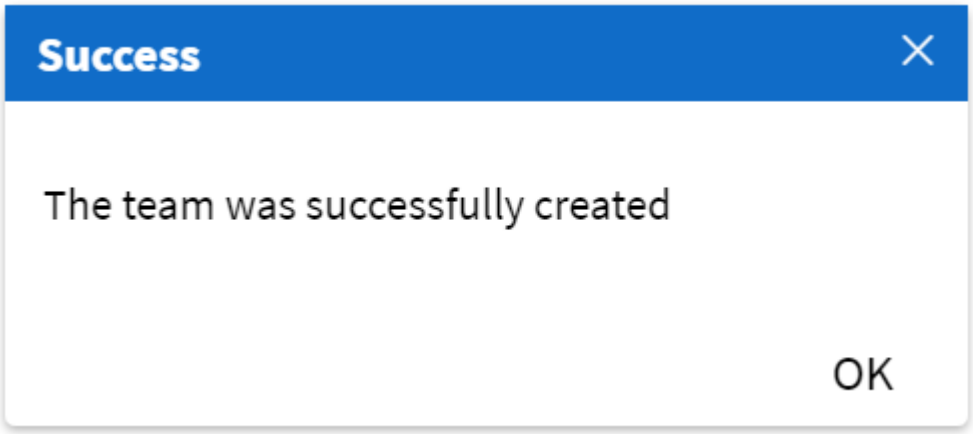


Figure 342: 7. Success modal

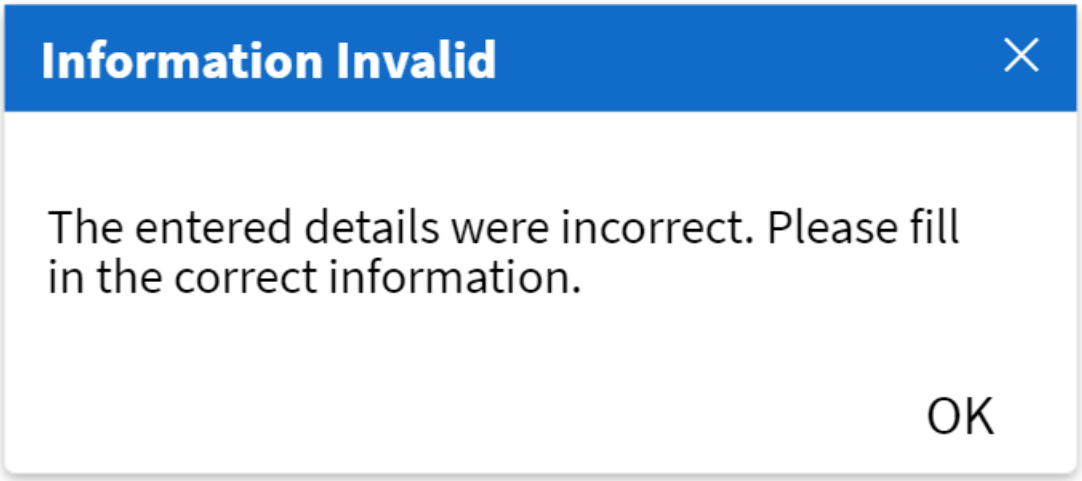


Figure 343: 7 Invalid Modal

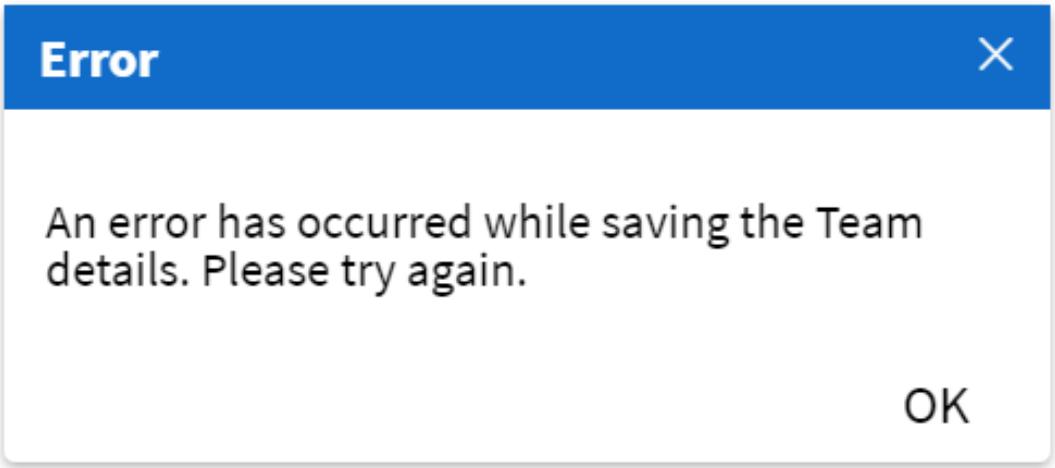


Figure 344: 7. Error Modal

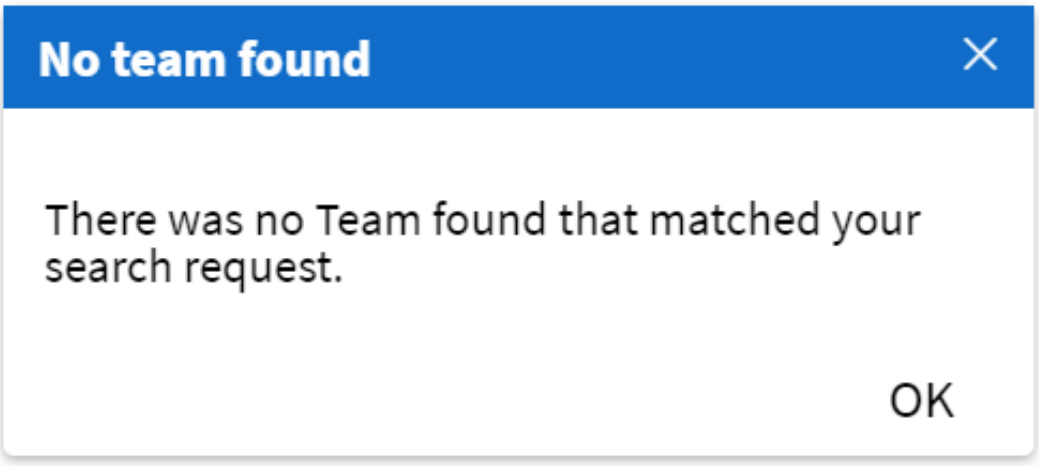


Figure 345: 7. No team found modal

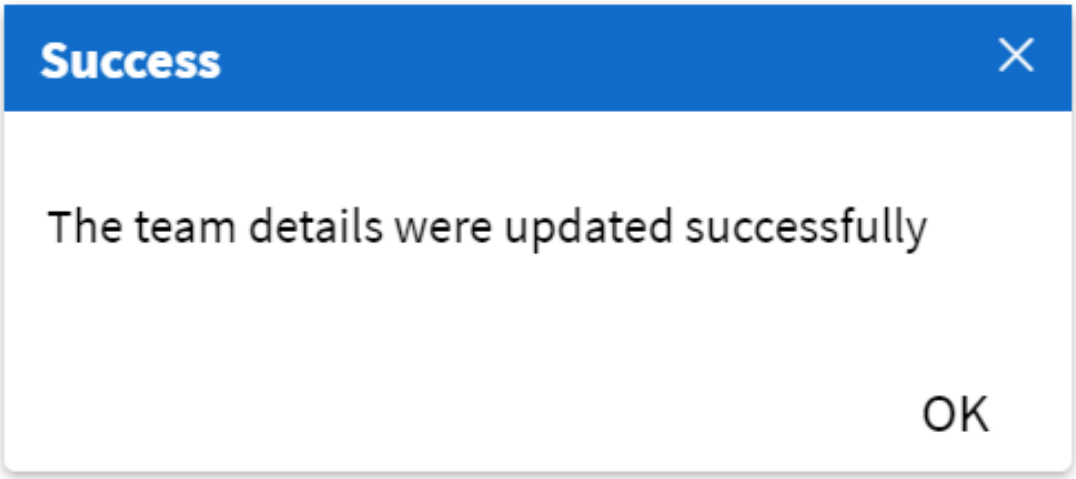


Figure 346: 7. Success modal

6.2.8 Supplier Order Subsystem

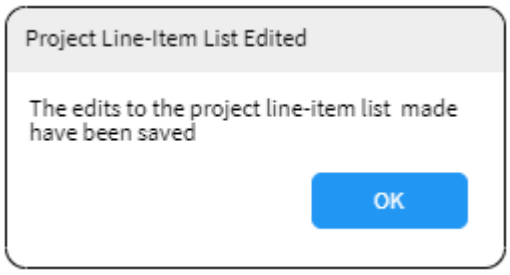


Figure 347: 8. Supplier Purchase Orders created Alert

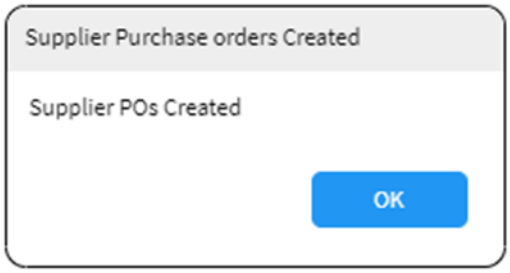


Figure 348: 8. Supplier Purchase Orders created Alert

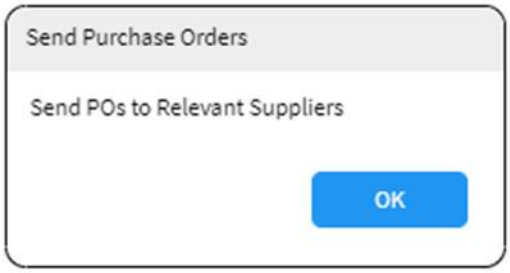


Figure 349: 8. Supplier Purchase Orders created Alert

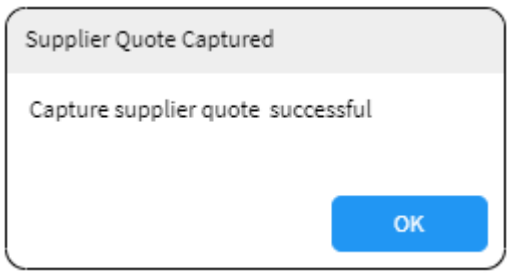


Figure 350: 8. Supplier Quote Captured Alert

6.2.9 Supplier Subsystem

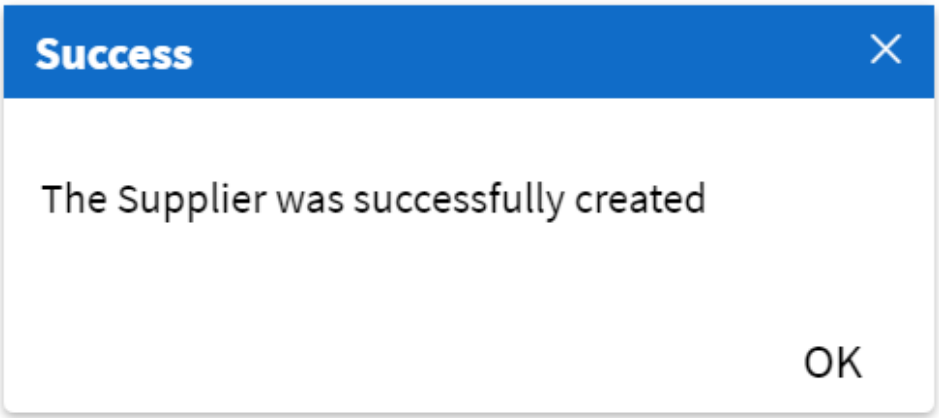


Figure 351: 9. Success modal

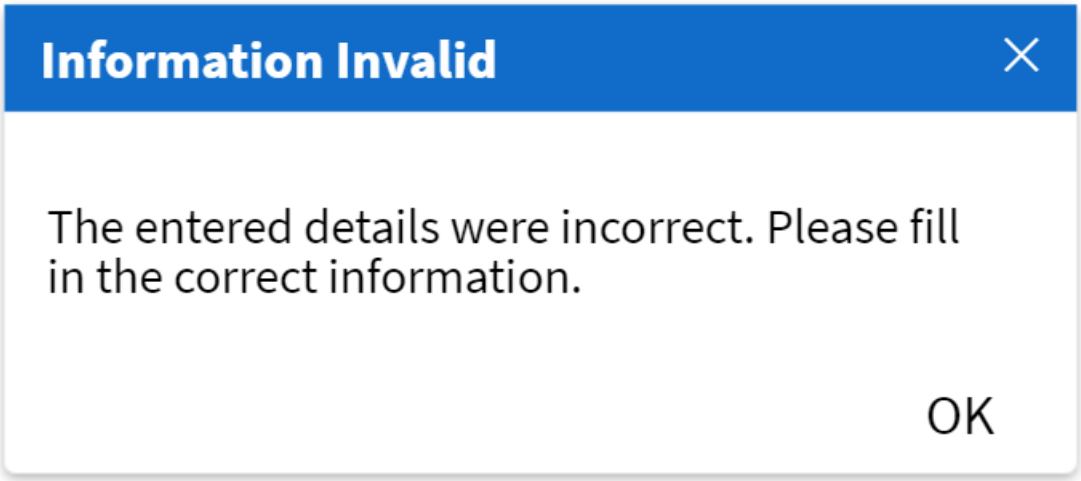


Figure 352: 9. Invalid modal

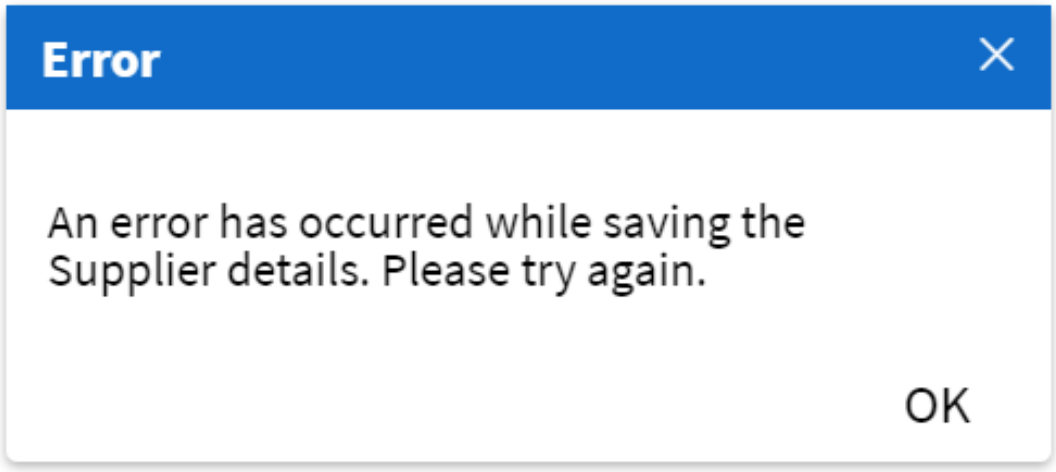


Figure 353: 9. Error modal

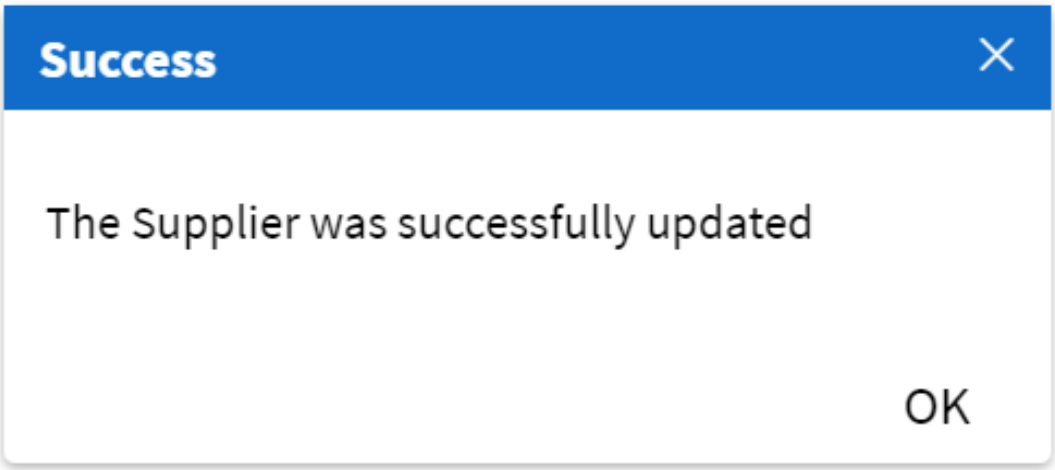


Figure 354: 9 Success modal

6.2.10 Inventory Management Subsystem

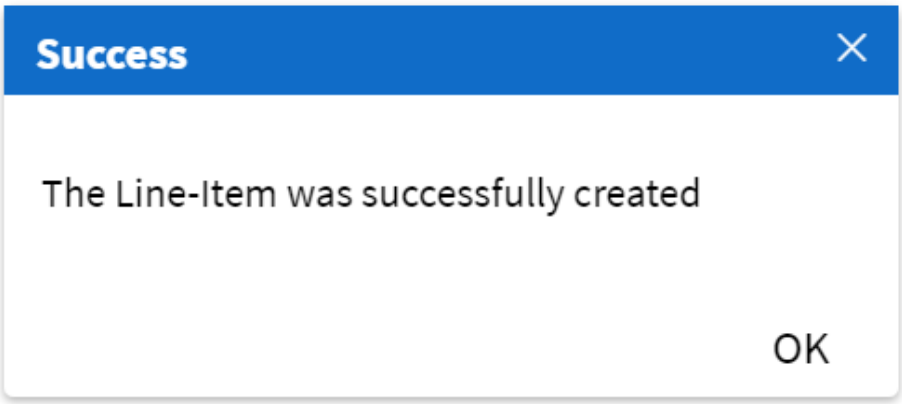


Figure 355: 10. Success modal

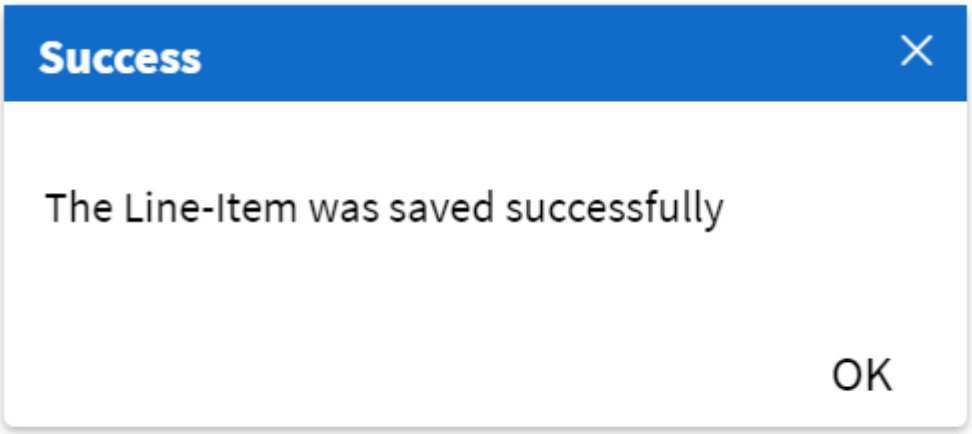


Figure 356: 10. Success modal

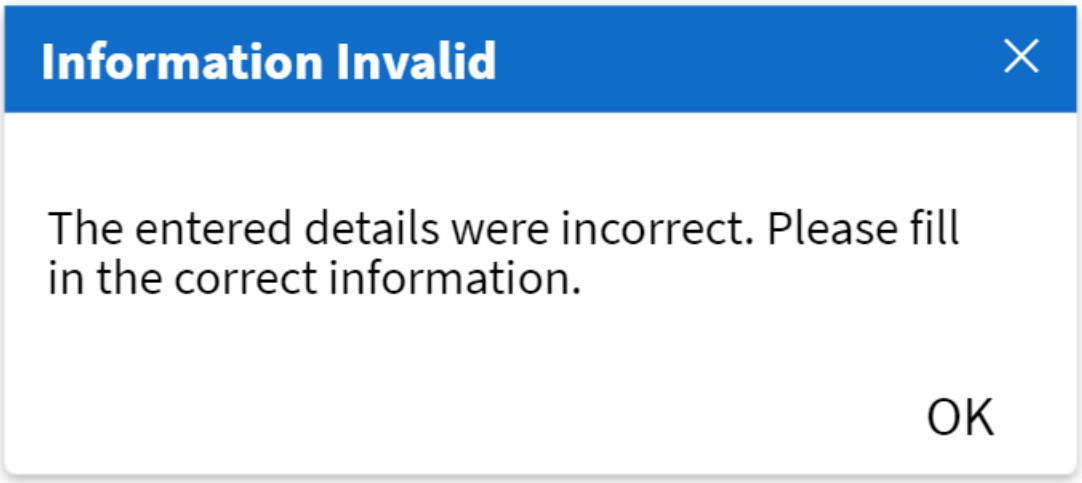


Figure 357: 10. Invalid modal

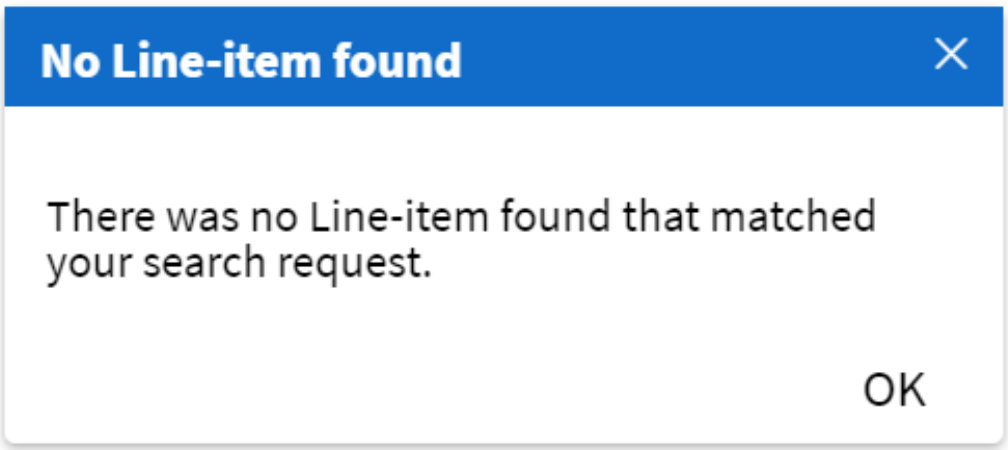


Figure 358: 10. No line-item found modal

6.2.11 Report Subsystem

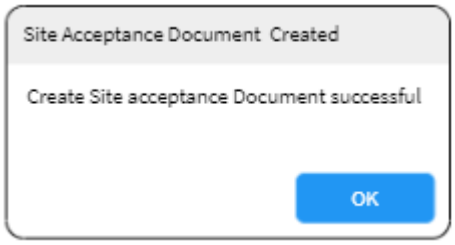


Figure 359: 11. Site Acceptance Document Created Modal

6.2.12 User Management Subsystem

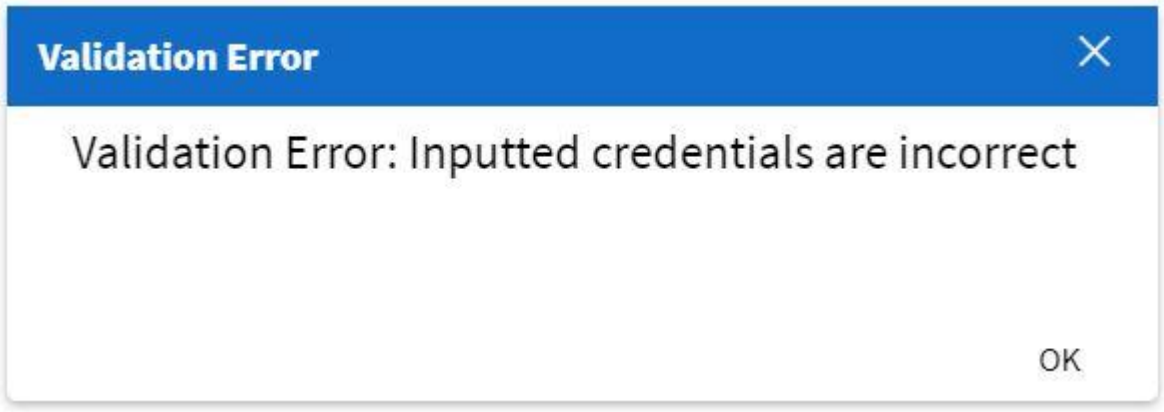


Figure 360: 12. Validation error message modal

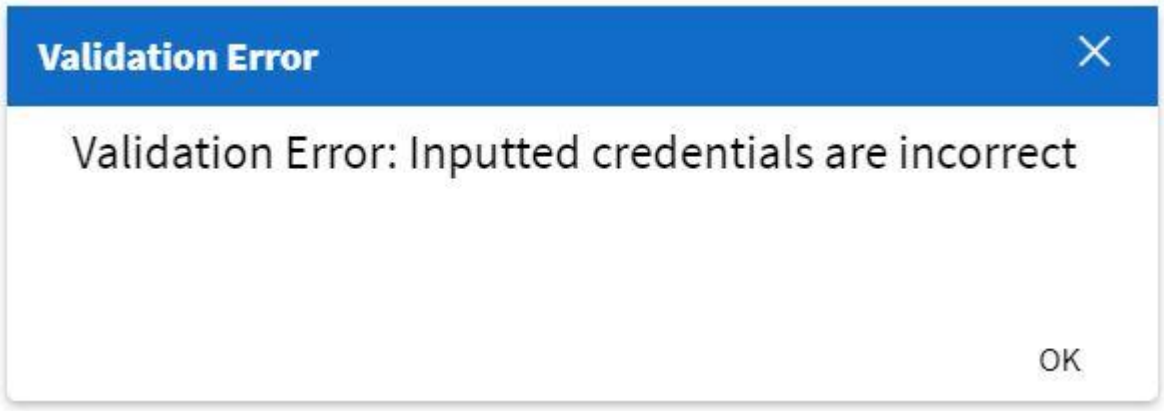


Figure 361: 12. Validation error message modal

6.2.13 Client Subsystem

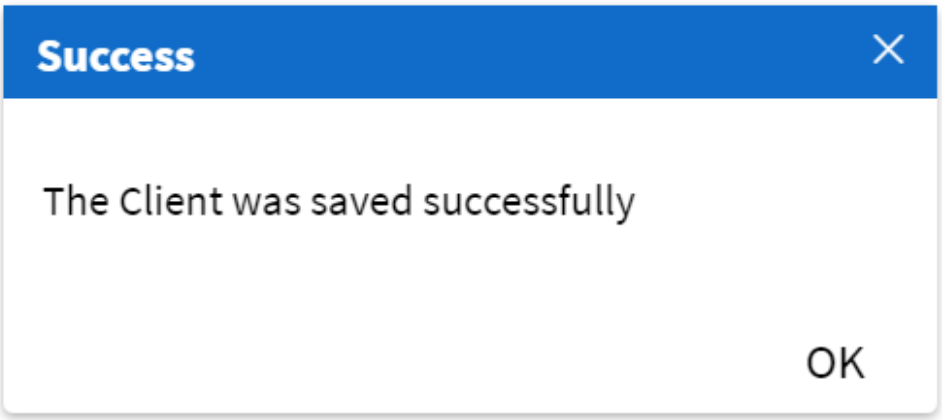


Figure 362: 13. Success modal

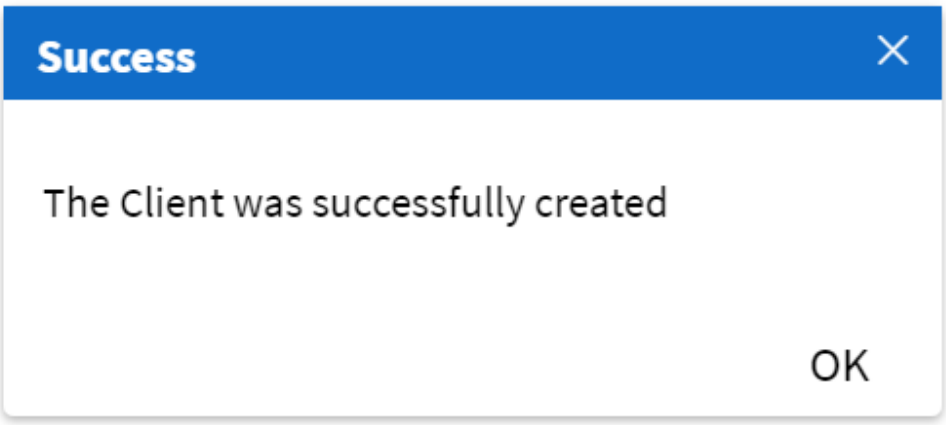


Figure 363: 13. Success modal

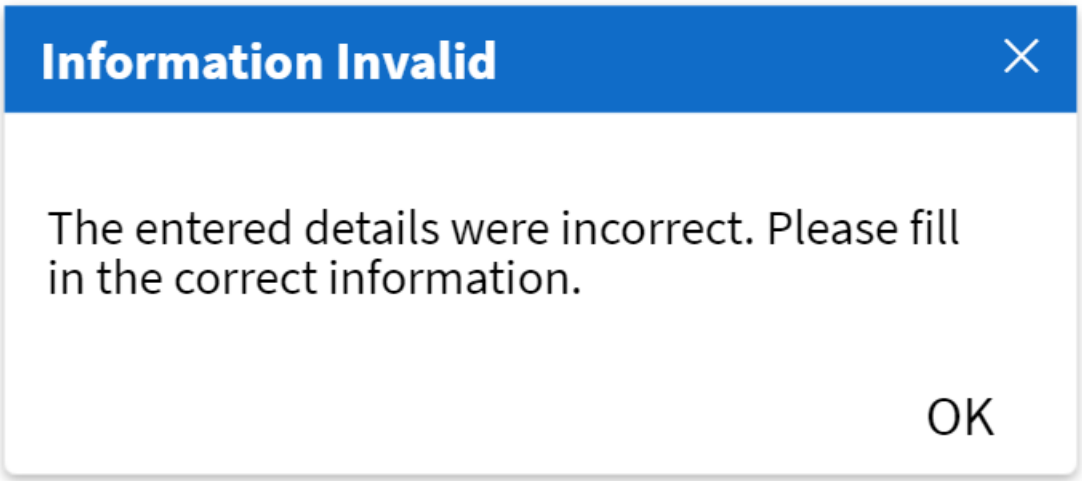


Figure 364: 13. Invalid modal

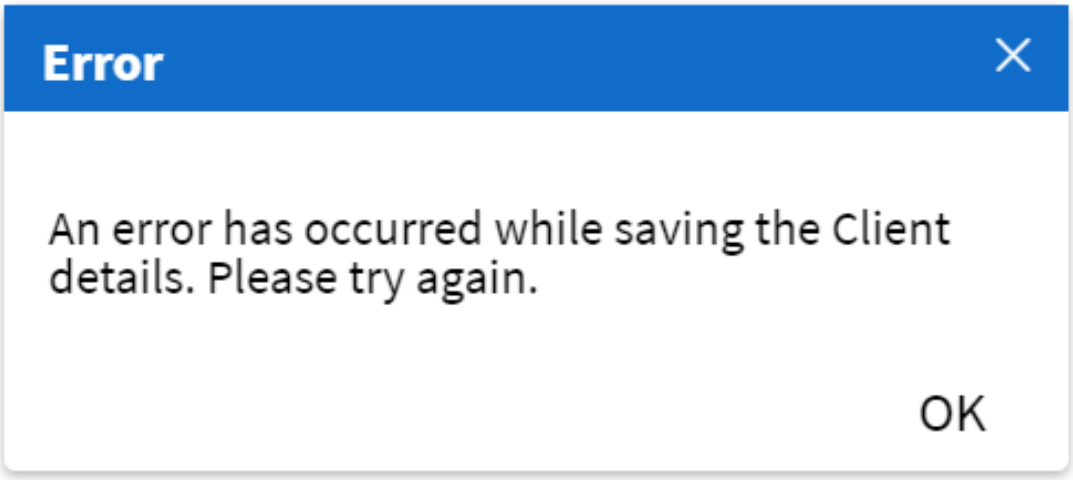


Figure 365: 13. Error modal

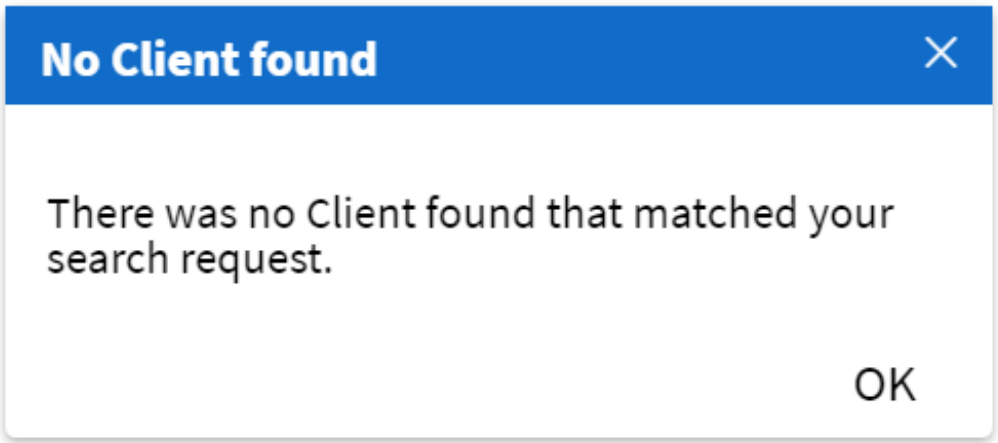


Figure 366: 13. No client found modal

Conclusion

This section contained descriptions of each report and its layout. It also indicated when it will be generated, and possible entities involved.

Package Design Specification

7. Test Specification

Introduction

This section contains the test plans, scenarios, procedures and test data in order to ensure the proper and desired outcome for the proposed system for Gohvan construction.

7.1 Test Plan

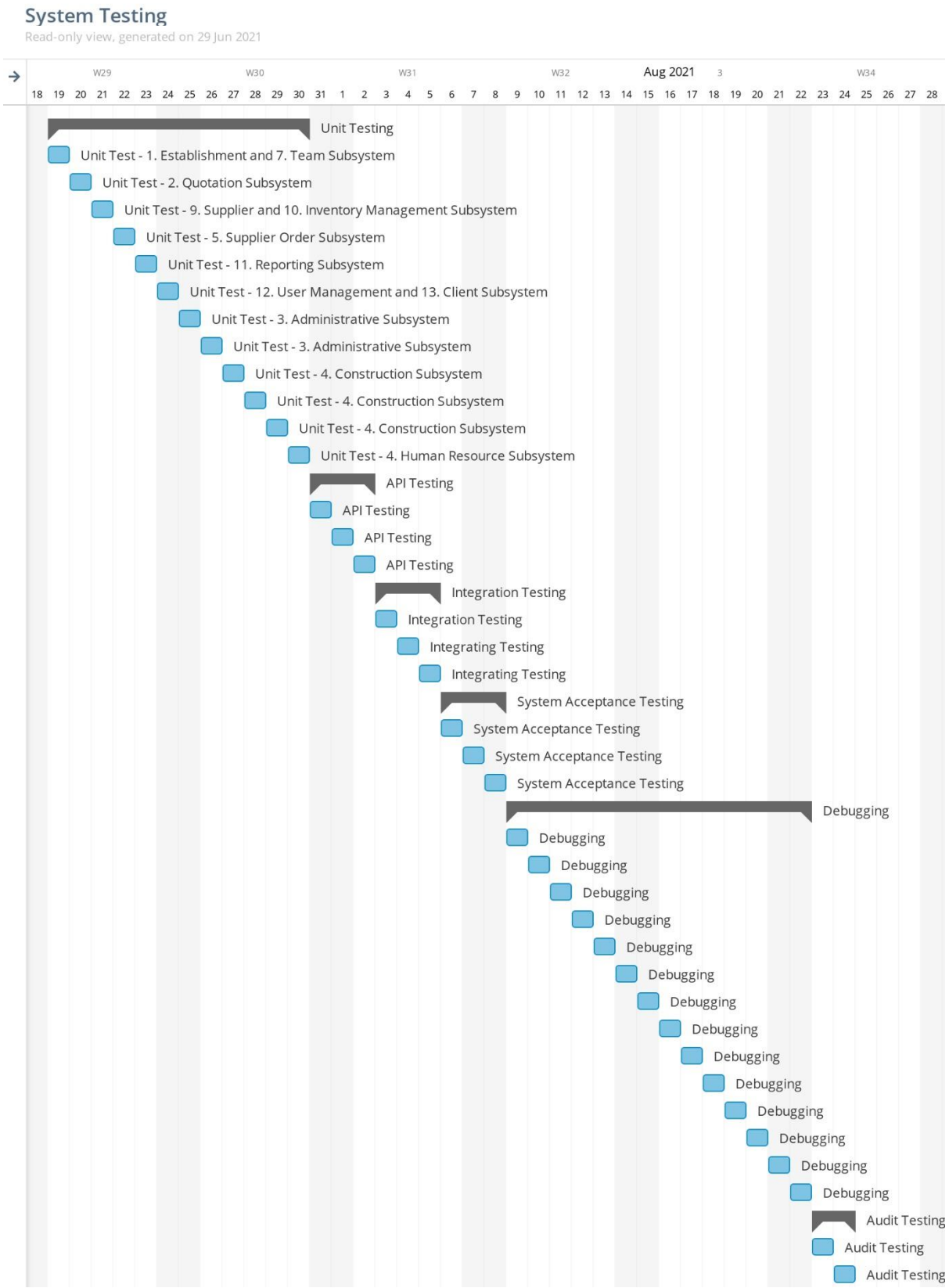
Step 1: Unit testing needs to be performed on the system. Unit testing will include adding necessary data/information before the rest of the system can be tested.

Step 2: API Testing, we will then test the APIs created for the system, e.g., Postman. This ensures proper communication between the system and the database.

Step 3: After unit testing and API testing, we will be doing integration Testing. Is the testing of the system's integration with 3rd party platforms. The system works with other platforms in order to process its functions.

Step 4: System Acceptance Test, we will combine the front-end with the API and the other interfaces and test them all together to see if the system is functioning correctly. The front end of the system needs to be tested. The GUI will be tested to ensure the system is able to handle the typical scenarios that the system will face.

Step 5: Audit Testing will help ensure the system is complete and ready for implementation.


System Testing
Read-only view, generated on 29 Jun 2021
W29
W30
W31
W32
Aug 2021
3
W34
18 19 20 21 22 23 24 25 26 27 28 29 30 31 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28
Unit Testing
Unit Test - 1. Establishment and 7. Team Subsystem
Unit Test - 2. Quotation Subsystem
Unit Test - 9. Supplier and 10. Inventory Management Subsystem
Unit Test - 5. Supplier Order Subsystem
Unit Test - 11. Reporting Subsystem
Unit Test - 12. User Management and 13. Client Subsystem
Unit Test - 3. Administrative Subsystem
Unit Test - 3. Administrative Subsystem
Unit Test - 4. Construction Subsystem
Unit Test - 4. Construction Subsystem
Unit Test - 4. Construction Subsystem
Unit Test - 4. Human Resource Subsystem
API Testing
API Testing
API Testing
API Testing
Integration Testing
Integration Testing
Integrating Testing
Integrating Testing
System Acceptance Testing
System Acceptance Testing
System Acceptance Testing
System Acceptance Testing
Debugging
Debugging
Debugging
Debugging
Debugging
Debugging
Debugging
Debugging
Debugging
Debugging
Debugging
Debugging
Debugging
Audit Testing
Audit Testing
Audit Testing

7.2 Test scenarios

Test Scenario 1:	Test the functionality of Checking Availability					
Test Scenario Description:	This use case describes the process the Health and Safety Officer will go through to create a Health and Safety file containing Health and Safety documents. The use case begins when the Health and Safety officer needs to create a Health and Safety file. The use case concludes when the Health and Safety officer has created and sent the file to the client.					
Pre-Condition:	Health and Safety officer needs to be logged into the system.					
Test Case ID:	3.1					
Test Case Description:	Check the functionality when making a valid, invalid and incomplete Health and Safety file submission.					
Test Case	Check response on filtering the availability function					
Test Data Types	**Test Step**	**Test Data**	**Expected results**	**Actual Results**	**Pass/Fail**	**Comments**
Valid Test Data:	1. Input Name 2. Select type 3. Input Description 4. Upload file 5. Click “Add more” button.	Name: Project Nissan Confirmation Type: Confirmation Description: Confirmation file for project: Nissan. File: “Confirmation File.pdf”	Upload the new Health and Safety file to the database.	The Health and Safety file has been uploaded.		
Invalid Test Data:	1. Input Name 2. Select type 3. Input Description 4. Upload file 5. Click “Add more” button.	Name: Project Nissan Confirmation Type: 212341 (Type incorrectly inputted as numbers) Description: Confirmation file for project: Nissan. File: “Confirmation File.pdf”	Validation Error Notification	Validation Error		
Incomplete Test Data:	1. Input Name 2. Select type 3. Input Description	Name: Project Nissan Confirmation Type: 212341	Validation Error Notification	Validation Error		

	4. Upload file 5. Click “Add more” button.	(Type incorrectly inputted as numbers) Description: Confirmation file for project: Nissan. File: (No file attached)				
Post Condition:	The Health and Safety file has been uploaded to the system. Display project screen.					

Test Scenario 2:	Test the functionality of Checking Availability					
Test Scenario Description:	This use case describes the process the human resource officer will follow when they want to create a new construction team. The system will capture and validate the team information if the information was validated successfully the system will store it in the TEAM table. A construction team consists of employees that are stored in the EMPLOYEE table.					
Pre-Condition:	The Human resource officer must be logged into the system					
Test Case ID:	7.1					
Test Case Description:	Check the functionality when making a valid, invalid, and incomplete create team submission.					
Test Case	Check response on filtering the availability function					
Test Data Types	**Test Step**	**Test Data**	**Expected results**	**Actual Results**	**Pass/Fail**	**Comments**
Valid Test Data:	1. Input team name 2. Select site manager 3. Select Team member 4. Click “Add (+)” button 4. Click “Create Team” button	Team Name: Cullinan Team Site Manager: John Johnson Team Member: Jenny Carlson	Create Team with Site manager and team members	Team with Site manager and team members created		
Invalid Test Data:	none	none	none	none		
Incomplete Test Data:	1. Input team name 2. Select site manager 3. Select Team member 4. Click “Add (+)” button 4. Click “Create Team” button	Team Name: Cullinan Team Site Manager: (None selected) Team Member: Jenny Carlson	Validation error notification	Validation error		
Post Condition:	The team has been added to the TEAM table and EMPLOYEE TEAM table successfully.					

Test Scenario 3:	Test the functionality of Capturing Supplier Quote					
Test Scenario Description:	This use case describes the event when a supplier sends a quotation to the Financial Officer. The Financial Officer will either accept or reject the quote. The system will capture and store the accepted quote.					
Pre-Condition:	Financial Officer has to be logged into the system. Financial Officer needs to be in the specific project to capture Supplier Quotes.					
Test Case ID:	8.4					
Test Case Description:	Check the functionality when making a valid, invalid and incomplete Supplier Quote file submission.					
Test Case	Check response on filtering the availability function					
Test Data Types	**Test Step**	**Test Data**	**Expected results**	**Actual Results**	**Pass/Fail**	**Comments**
Valid Test Data:	1. Click “Capture Quote” button 2.Enter Required Information 3. Click “Submit” button 4. Click “Confirm” button 5. Click “OK” button	File Upload: SupplierQuote.pdf Quote Number: 123456789 Supplier: Proud Boys (PTY) Quote Amount: 12345.00	Confirm Supplier Quote modal. Supplier Quote Captured alert Upload the new Supplier Quote to the database.	The Supplier Quote is stored and the supplier quote details are captured.		
Invalid Test Data:	1. Click “Capture Quote” button 2.Enter Required Information 3. Click “Submit” button 4. Click “Confirm” button 5. Click “OK” button	File Upload: SupplierQuote.pdf Quote Number: OneTwoThree Supplier: Proud Boys (PTY) Quote Amount: A Lot	Validation Error Notification	Validation Error		

Incomplete Test Data:	1. Click “Capture Quote” button 2.Enter Required Information 3. Click “Submit” button 4. Click “Confirm” button 5. Click “OK” button	File Upload: SupplierQuote.pdf Quote Number: Supplier: Proud Boys (PTY) Quote Amount:	Validation Error Notification	Validation Error		
Post Condition:	The Supplier Order has been uploaded to the system. Display Supplier Order screen.					

Test Scenario 4:	Test the functionality of Creating an Account					
Test Scenario Description:	This use case describes the process that the Human Resource Officer will follow when they want to create a new account. The system will capture and validate the information. If all processes are successful, the system will save the new account in the USER table.					
Pre-Condition:	The Human Resource Officer must be logged into the system.					
Test Case ID:	6.7					
Test Case Description:	Check the functionality when making a valid, invalid, and incomplete create account submission.					
Test Case	Check response on filtering the create account function					
Test Data Types	**Test Step**	**Test Data**	**Expected results**	**Actual Results**	**Pass/Fail**	**Comments**
Valid Test Data:	1.Search Employee 2.Input Username 3.Input Email Address 4.Input Password 5.Input Password again to confirm 6. Input Security Question 7.Input Security Answer 8. Click “Create” button	Employee: Sarah Darling Email: sarah.darling@gmail.com Password: Manick Confirmed Password: Manick Security Question: What is the full name of your favourite primary school teacher? Security Answer: Manuel Seems	Create User with account details	User with account details created		
Invalid Test Data:	none	none	none	none		
Incomplete Test Data:	1.Search Employee 2.Input Username 3.Input Email Address 4.Input Password	Employee: Sarah Darling Email: sarah.darling Password: Manick Confirmed Password:	Validation Error Dialog displays	Validation Error		

	5.Input Password again to confirm 6. Input Security Question 7.Input Security Answer 8. Click “Create” button	Mani Security Question: What is the full name of your favourite primary school teacher? Security Answer:				
Post Condition:	The user has been added to the USER table successfully					

Test Scenario 5:	Test the functionality of Searching for a project.					
Test Scenario Description:	This use case entails the process the system will go through when the user requires to search for a specific project that is currently active on the system. The user will enter the details of the project for the system to then read the details of the project from the associated tables.					
Pre-Condition:	• User must be logged onto the system • The project must exist on the system to view.					
Test Case ID:	4.9 Search project					
Test Case Description:	Checks the functionality when searching for a project on the system					
Test Case	Check response on filtering the availability function					
Test Data Types	**Test Step**	**Test Data**	**Expected results**	**Actual Results**	**Pass/Fail**	**Comments**
Valid Test Data:	1: The user hovers over the construction menu strip. 2: The user clicks on the "Project" dropdown menu item. 3: The user enters the project name as a search criteria in the search text box. 4: The user clicks on the list object displayed via the list component.	Project Name: Nissan Rosslyn	Display Searched project result on search project page. Display Project page of searched result	Correct project page is presented to the user.		
Invalid Test Data:	1: The user hovers over the construction menu strip. 2: The user clicks on	Project Name: Nissan Rosslyn	Display Validation error for incorrect search criteria standard.	Project page is not displayed to the user.		

	the "Project" dropdown menu item. 3: The user enters the project name as a search criteria in the search text box. 4: The user clicks on the list object displayed via the list component.					
Incomplete Test Data:	1: The user hovers over the construction menu strip. 2: The user clicks on the "Project" dropdown menu item. 3: The user enters the project name as a search criteria in the search text box. 4: The user clicks on the list object displayed via the list component.	Project Name: Nissan Rosslyn	Display notification for now found project with associated name.	Project page is not displayed to the user.		
Post Condition:	The correct project page is presented to the user for further use of the system.					

Test Scenario 6:	Test the functionality of Capturing Supplier Quote					
Test Scenario Description:	This use case describes the process the Operations Manager will go through once the Construction photos are uploaded to the system. The system will capture the already created Radio Frequency report and store the report in the system. Once the report has been captured and the necessary actors are notified. The ninety percent claim can then be request from the client, as well as to request the site inspection from the quality control officer.					
Pre-Condition:	• Site completion photos must be uploaded to the system. • 100% completion on construction site.					
Test Case ID:	4.14 Upload Radio Frequency report					
Test Case Description:	Checks the functionality when uploading a radio Frequency report to the system.					
Test Case	Check response on filtering the availability function					
Test Data Types	**Test Step**	**Test Data**	**Expected results**	**Actual Results**	**Pass/Fail**	**Comments**
Valid Test Data:	1: The Operations Manager hovers over the Construction Menu strip. 2: The Operations Manager clicks on the “Radio Frequency Report” card displayed. 3: The Operations Manager enters the details for the Radio Frequency report 4: The Operations Manage clicks the “Submit”	Name: Nissan Rosslyn RF report Project Name: Nissan Rosslyn Item Code: W09UCM Serial Number: 240455341057A070150085 Air Conditioner Size: 4m x 4m Description: Alliance Air-Conditioner – 9000 BTU	Confirm upload Radio Frequency report modal. Upload Radio Frequency Report notification.	Upload the Radio Frequency to the system.		

	button displayed. 5: The Operations Manager clicks the "Yes" button					
Invalid Test Data:	1: The Operations Manager hovers over the Construction Menu strip. 2: The Operations Manager clicks on the "Radio Frequency Report" card displayed. 3: The Operations Manager enters the details for the Radio Frequency report 4: The Operations Manage clicks the "Submit" button displayed. 5: The Operations Manager clicks the "Yes" button	Name: Nissan Rosslyn RF report Item Code: W09UCM Serial Number: 240455341057A070150085 Air Conditioner Size: 4m x 4m Description: Alliance Air-Conditioner – 9000 BTU	Validation error notification on Project Name not provided.	Validation Error		
Incomplete Test Data:	1: The Operations Manager	Name: Nissan	Validation error notification	Validation Error		

	hovers over the Construction Menu strip. 2: The Operations Manager clicks on the “Radio Frequency Report” card displayed. 3: The Operations Manager enters the details for the Radio Frequency report 4: The Operations Manage clicks the “Submit” button displayed. 5: The Operations Manager clicks the “Yes” button	Rosslyn RF report Project Name: Nissan Rosslyn Item Code: W09UCM Serial Number: 240455341057A070150085 Air Conditioner Size: 4m x 4m Description: Alliance Air-Conditioner – 9000 BTU	on Radio Frequency report not uploaded			
Post Condition:	The Radio Frequency report is uploaded to the system successfully.					

7.3 Test Data

The test data is contained in the test scenarios’ steps above.

Conclusion

This section contained the test plans, scenarios, procedures and test data in order to ensure the proper and desired outcome for the proposed system for Gohvan construction.

8. Hardware and Software Specification

Introduction

This section contains a list of Gohvan Construction's required hardware and software for the development and use for the proposed system.

8.1 Hardware and software

8.1.1 Software requirements

System Development

During the development of our system, we made use of the following software:

Operating System	Windows 10
	iOS
	Android
IDE	Visual studio 2019 Community
	VS code
Internet Browser	Google chrome / Firefox Developer Edition
Process Model Design Software	(SAP) Power Designer
Documentation Software	Microsoft Office (Word, PowerPoint)
Database Management	Microsoft SQL Server 2014 or later
Framework / Libraries	Bootstrap libraries as plug-in to visual studio
	Angular Framework
	Ionic Framework
	Angular Materials

System In-Use

During the implementation of our system, the following software is required:

Operating System	Windows 10
Internet Browser	Google Chrome
Documentation Software	Microsoft Office (Word, Excel)
PDF Viewer	Adobe Acrobat Reader DC
Screenshot Software	Snip & Sketch

8.1.2 Hardware requirements

System Development

During the implementation of our system, the following hardware is required:

Hardware Requirements – Desktop or Laptop

CPU (Central Processing Unit)	intel CORE i3 2.4 GHz or faster processor
RAM (Random Access Memory)	Minimum of 6GB of RAM
Disk Space	Minimum Capacity of 256GB of SSD (Solid State Drive)
	Minimum Capacity of 1TB of HDD
HDD Speed (Hard Disk Drive)	5400 RPM hard disk drive
GPU (Graphics Processing Unit)	DirectX 9-capable video card (1024 x 768 or higher resolution)

Peripheral Devices	Mouse
	Keyboard
	Printer
Internet Connectivity	Ethernet Port
	Wi-Fi Connectivity
Minimum Internet Speed	10Mbps

Hardware Requirements – Mobile

CPU (Central Processing Unit)	Dual-core 1.2 GHz (Mobile)
RAM (Random Access Memory)	Minimum of 2GB of RAM
Disk Space	Minimum Capacity of 64GB
Battery Size	3000mAh
Camera	8 MP
Display	480 x 853 (Viewport)
Network	3G / 4G / LTE

Conclusion

This section contained a list of Gohvan Construction's required hardware and software for the development and use for the proposed system.

9. Network/ Web Lay out Specification

Introduction

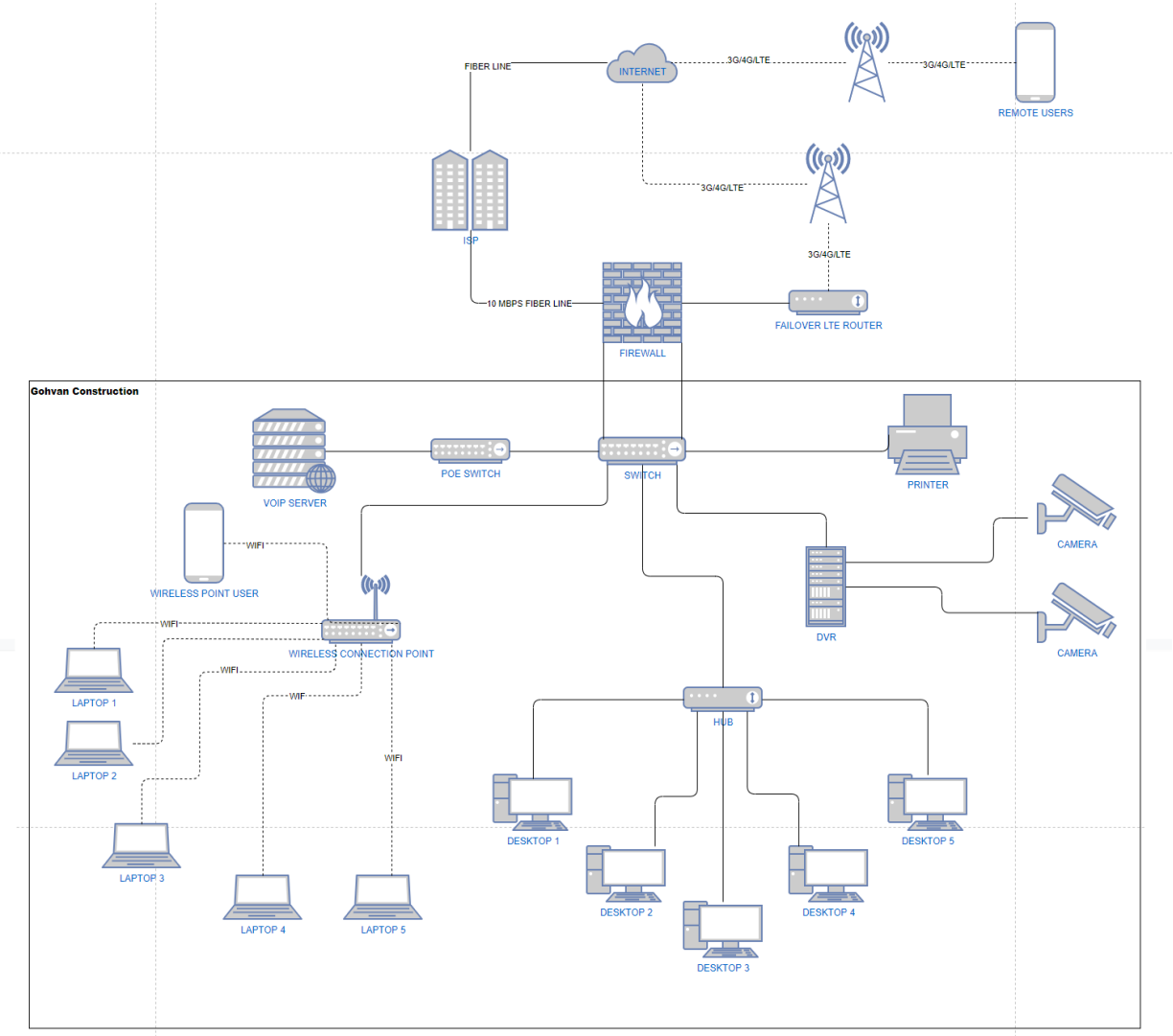
This section contains the network specifications for the proposed solution for Gohvan Construction.

9.1 Network/Web Layout

9.1.1 Network Specifications

Internet Connectivity	10Mbps Fibre Line
Web / Server Hosting	Support for MySQL, ASP.NET and domain

9.1.2 Network Diagram



Conclusion

This section contained the network specifications for the proposed solution for Gohvan Construction.

10. Validation

Introduction

This section in the deliverable displays the validation tables which represent every use case requirement and the data flows within, and which entities are being interacted with.

10.1 Validate the technical specifications against requirements

10.1.1 Establishment Subsystem

1.Establishment Subsystem						
Requirement	Use Case No.	Use Case Title	Processes from Primitive Data Flow Diagram	Entities from ERD	Screen	Report
1.1 Create TSS	1.1	Create TSS	1.1.1 Capture TSS detail Task (Angular/ionic)	Site Type	Create TSS screen	None
			1.1.2 Validate Captured details Task (Angular/ionic)	None	None	None
			1.1.3 ALT: Notification of Validation error Task (Angular)	None	Invalid Notification screen	None
			1.1.3 Confirmation of details Task (Angular/ionic)	None	Confirmation Screen	None
			1.1.3 Save TSS details Task (asp.net Core)	TSS, TSS Photos, Project Line Items, Project	None	None
			1.1.5 Notification of successful save Task (Angular)	None	Success Notification Screen	None
			1.1.5 ALT: Notification of unsuccessful save Task (Angular)	None	Error Screen	None
1.2 Edit TSS	1.2	Edit TSS	1.2.1 Retrieve and display TSS details Task (Angular/ionic)	TSS, TSS Photos, Project Line Items	Edit TSS screen	None
			1.2.2 Request and capture edited TSS info Task (Angular/ionic)	None	None	None

			1.2.3 Validate Captured TSS details Task (Angular/ionic)	None	None	None
			1.2.3 ALT: Notification of Validation error Task (Angular)	None	Invalid Notification screen	None
			1.2.3 ALT Capture Submit TSS request Task (Angular)	None	None	None
			1.2.4 Confirmation of updated TSS info Task (Angular/ionic)	None	Confirmation Screen	None
			1.2.5 Update TSS info Task (asp.net Core)	TSS, TSS Photos, Project Line Items	None	None
			1.2.6 Notification of successful save Task (Angular)	None	Success Notification Screen	None
			1.2.6 ALT: Notification of unsuccessful save Task (Angular)	None	Error Screen	None
1.3 Submit TSS report	1.3	Submit TSS report	1.3.1 Retrieve and display TSS information Task (Angular/ionic)	None	Edit TSS screen	None
			1.3.2 Confirmation of TSS information Task (Angular/ionic)	None	Confirmation Screen	None
			1.3.3 Submit TSS report Task (asp.net Core)	Project	None	None
			1.3.4 Notification of successful Submission Task (Angular)	None	Success Notification Screen	None
			1.3.4 ALT: Notification of unsuccessful Submission Task (Angular)	None	Error Screen	None

10.1.2 Quotation Subsystem

2. Quotation Subsystem					
Use Case No.	Use Case Title	Processes from Primitive Data Flow Diagram	Entities	Screens	Reports
2.1	Create Quote	2.1.1 Display Quote Screen Task (Angular)	None	Quote Screen	None
		2.1.2 Retrieve Project Line Item and Project Information Task (Angular .NET Core)	None	None	None
		2.1.3 Retrieve Company, VAT and Client Information (Angular .NET Core)	Company Information VAT Client	None	None
		2.1.4 Display Create Quote Screen Task (Angular)	None	Create Quote Screen	None
		2.1.5 Capture and Validate Details Task (Angular)	None	None	None
		2.1.5 ALT: Notification of Validation Error Task (Angular)	None	Validation Error Dialog	None
		2.1.6 Display Confirmation Modal Task(Angular)	None	Confirmation Modal	None
		2.1.7 Request confirmation of captured info Task(Angular)	None	None	None
		2.1.8 Save Quote Details Task (Angular .NET Core)	Client Quote	None	None
		2.1.9 Generate Excel Sheets Task (Angular)	None	None	None
		2.1.10 Notification of successful quote creation Task (Angular)	None	Successful Quote Creation Dialog	None
		2.1.10 ALT: Notification of unsuccessful quote creation Task (Angular)	None	Unexpected Error Dialog	None
		2.1.11 Create Excel Sheets Download Task (Angular)	None	None	None
2.2	Search Quote	2.2.1 Display Quote Screen Task (Angular)	None	None	None

		2.2.2 Capture Search Parameter Task(Angular)	None	None	None
		2.2.3 Validate Captured Details Task (Angular)	None	None	None
		2.2.3 ALT: Notification of Validation Error (Angular)	None	Validation Error Dialog	None
		2.2.4 Retrieve Search Results Task (Angular .NET Core)	Client Quote	None	None
		2.2.5 Populate Search Results Table Task(Angular)	None	None	None
		2.2.5 ALT: Notification of no quote found Task (Angular)	None	No quote found Dialog	None
		2.2.6 Display Selected Employee Task(Angular)	None	None	None
		2.2.6 ALT: Notification of display error encountered Task (Angular)	None	Unexpected Error Dialog	None
2.3	Capture Purchase Order	2.3.1 Display Upload Purchase Order Document Modal Task (Angular)	None	Quote Screen Upload Purchase Order Modal	None
		2.3.2 Capture Purchase Order Document Information Task (Angular)	None	None	None
		2.3.3 Validate Purchase Order Document Details Task (Angular)	None	None	None
		2.3.3 ALT: Notification of Validation Error Task (Angular)	None	Validation Error Dialog	None
		2.3.4 Display Confirmation Modal Task (Angular)	None	Confirmation Modal	None
		2.3.5 Save Changes to Client Purchase Order Table Task (Angular .NET Core)	Client Purchase Order	None	None
		2.3.6 Notification of successful upload Task (Angular)	None	Successful Purchase Order Upload Dialog	None
		2.3.6 ALT: Notification of unsuccessful upload Task (Angular)	None	Unexpected Error Dialog	None

10.1.3 Administration Subsystem

3. Administrative Subsystem						
Requirement	Use Case No.	Use Case Title	Processes from Primitive Data Flow Diagram	Entities from ERD	Screen	Report
3.1 Create Health and Safety file	3.1	Create Health and Safety file	3.1.1 Display upload HS file screen task(Angular)	None	None	None
			3.1.2 Capture and validate inputted information task(Angular)	None	None	None
			ALT 3.1.3 Validation error message task(Angular)	None	Validation error message	None
			3.1.3 Display "Confirm upload HS file" modal task(Angular)	None	None	None
			3.1.4 Upload HS file task(Angular .NET Core)	Health and Safety	None	None
			3.1.5 Display "Another HS file upload" modal task(Angular)	None	None	None
			3.1.6 Return to step 3.1.1 task (Angular)	None	None	None
3.2 Search Health and Safety File	3.2	Search Health and Safety File	3.2.1 Display "Search HS file" screen task(Angular)	None	Search HS file screen	None
			3.2.2 Capture and validate inputted information task(Angular)	None	None	None
			ALT 3.2.3 Validation error message task	None	Validation error message	None
			3.2.3 Retrieve HS file information task(Angular .NET Core)	Health and Safety, Health and Safety type	None	None

			ALT 3.2.4 HS file error information task	None	HS file error message	None
			3.2.4 display "Searched HS file" screen task(Angular)	None	Searched HS file screen	None
			3.2.5 Download HS file task (Angular)	Health and Safety	None	None
3.3 Create Health and Safety type	3.3	Create Health and Safety type	3.3.1 Display "Create Hs file type" screen task(Angular)	None	Create Hs file type screen	None
			3.3.2 Capture and validate inputted information task(Angular)	None	None	None
			ALT 3.3.3 Validation error message task(Angular)	None	None	None
			3.3.3 Display "Confirm HS file type" modal task(Angular)	None	Confirm HS file type modal	None
			3.3.4 Create Hs file type task (Angular .NET Core)	Health and Safety type	None	None
3.4 Search Health and Safety type	3.4	Search Health and Safety type	3.4.1 Display "Search HS file type" screen task(Angular)	None	Search Hs file type screen	None
			3.4.2 Capture and validate inputted information task (Angular)	None	None	None
			3.4.3 Retrieve HS file type task(Angular .NET Core)	Health and Safety type	None	None
			3.4.4 Display "Searched HS file type" screen task(Angular)	None	Searched HS file type screen	None
3.5 Create Drawing type	3.5	Create Drawing type	3.5.1 Display "Create drawing	None	Create drawing	None

			type" screen task (Angular)2		type screen	
			3.5.2 Capture and validate inputted information task(Angular)13	None	None	None
			3.5.3 Display "Confirm drawing type" modal task(Angular)2	None	Confirm drawing type modal	None
			3.5.4 Create drawing type task(Angular .NET Core)2	Drawing type	None	None
3.6 Search Drawing type	3.6	Search Drawing type	3.6.1 Display "Search drawing type" screen task(Angular)2	None	Search drawing type screen	None
			3.6.2 Capture and validate inputted information task(Angular)2	None	None	None
			3.6.3 Retrieve Drawing type task(Angular .NET Core)2	Health and Safety type	None	None
			3.6.4 Display "Searched Drawing type" screen task(Angular)2	None	Searched drawing type screen	None
3.7 Upload Formal Drawing	3.7	Upload Formal Drawing	3.7.1 Gui Display: "Upload formal drawing" screen task(Angular)	None	Upload formal drawing screen	None
			3.7.2 Capture and validate inputted information task(Angular)	None	None	None
			ALT 3.7.3 Validation error message task(Angular)2	None	Validation error message	None
			3.7.3 Display "Formal drawing upload" modal task(Angular)	None	Formal drawing upload modal	None

			3.7.4 Upload formal drawing task(Angular)	Drawing	None	None
3.8 Upload second cut drawing	3.8	Upload second cut drawing	3.8.1 Gui Display: "Upload second-cut drawing" screen task(Angular)2	None	Upload second-cut drawing screen	None
			3.8.2 Capture and validate inputted information task(Angular)2	None	None	None
			ALT 3.8.3 Validation error message task(Angular)3	None	Validation error message	None
			3.8.3 Display "second-cut drawing upload" modal task(Angular)2	Drawing	second-cut drawing upload modal	None
			3.8.4 Upload Second-cut drawing task(Angular)	Drawing	None	None
3.9 Search As-built document	3.9	Search As-built document	3.9.1 Display "Search As-built file" screen task(Angular)	None	Search As-built file screen	None
			3.9.2 Capture and validate inputted information task(Angular)	None	None	None
			ALT 3.9.3 Validation error message task(Angular)4	None	Validation error message	None
			3.9.3 Retrieve As-built file task(Angular)	As-Built, Project	None	None
			ALT 3.9.3 Validation error message task(Angular)5	None	Validation error message	None
			3.9.4 Display "Searched As-built file" screen task(Angular)	None	Searched As-built file screen	As-Built
			3.9.5 Download As-built file task(Angular)	None	None	As-Built

3.10 Generate As-Built Document	3.10	Generate As-Built Document	3.10.1 Retrieve project status task(Angular .NET Core)	Project	None	None
			3.10.2 Display "Generate As-built" modal task(Angular)	None	Generate As-built modal	None
			3.10.3 Retrieve all As-built information task(Angular .NET Core)	Project, Health and Safety, Construction Site, Compliance documents, Report, Construction photos, Completion photos, Drawing, TSS, Radio Frequency report	None	None
			3.10.4 Display "Add new section" screen task(Angular)	None	Add new section screen	None
			3.10.5 Capture and validate inputted information task(Angular)	None	None	None
			ALT 3.10.6 Validation error message task(Angular)7	None	Validation error message	None
			3.10.6 Generate As-built document task(Angular)	None	None	As-built
			3.10.7 Upload As-built document task(Angular)	As-built	None	None
			3.10.8 Display "Generated As-built" modal task(Angular)	None	Generated As-built modal	None
			3.10.9 Display "Project" screen task(Angular)	None	Project screen	None

10.1.4 Construction Subsystem

4. Construction Subsystem					
Use Case No.	Use Case Title	Processes from Primitive Data Flow Diagram	Entities	Screens	Reports
4.1	Capture Site - Go-Ahead	4.1.1 Display "Project Page" Task (Angular and .Net Core)	None	Project page screen	None
		4.1.2 Validate for existing Construction Sites Task (Angular and .Net Core)	Construction Site	Project page screen	None
		ALT 4.1.3 Existing Site go Ahead found notification Task (Angular .Net Core)	None	Notification screen	None
		4.1.3 Display Confirmation Modal Task (Angular)	None	Confirmation modal	None
		4.1.4 Display "Capture Site-Go-Ahead" page Task (Angular)	None	Capture Site- Go-Ahead screen	None
		4.1.5 Capture Site-Go-Ahead details Task (Angular)	None	Capture Site- Go-Ahead screen	None
		4.1.6 Validate Captured Site-go-Ahead information Task (Angular)	None	Capture Site- Go-Ahead screen	None
		4.1.7 ALT Display Validation Error Task (Angular)	None	Invalid notification screen	None
		4.1.7 Display Confirmation modal Task (Angular)	None	Confirmation Modal	None
		4.1.8 Save captured Site-Go-Ahead details Task (Angular and .Net Core)	Construction Site	None	None
		4.1.9 Update Project Status Task (Angular and .Net Core)	Project	None	None
		4.1.10 Display notification on successfully Captured of Site-Go-Ahead information Task (Angular)	None	Successfully uploaded notification	None

		4.1.11 Generate Site-Go-Ahead Email Task (Angular and .Net Core)	None	None	None
		4.1.12 Retrieve Employee emails Task (Angular and .Net Core)	Employee	None	None
		4.1.13 Email Employees Task (Angular and .Net Core)	None	None	None
		4.1.14 Display modal on successful emails sent Task (Angular)	None	Successfully notified employee modal	None
4.2	Search construction	4.2.1 Display Construction Dropdown Task (Angular)	None	Home Screen	None
		4.2.2 Display "Search Construction page" Task (Angular)	Construction Site, Site Progress Description	Search Construction page screen	None
		4.2.3 ALT Display Construction page Task (angular)	None	Construction Page screen	None
		4.2.3 Capture Construction Search Criteria Task (Angular and .NET Core)	None	Search Construction page screen	None
		4.2.4 Validate Construction Search Criteria Task (Angular)	None	Search Construction page screen	None
		4.2.5 ALT Display Validation error on search criteria Task (Angular)	None	Invalid notification screen	None
		4.2.5 Capture Construction information Task (Angular and .NET Core)	Construction Site	Search Construction page screen	None
		4.2.6 ALT Display Search Error Task (Angular)	None	Search Error notification	None
		4.2.6 Display Searched Construction Information Task (Angular)	None	Search Construction page screen	None
		4.2.7 Display "Construction Site details page" Task (Angular)	Construction Site	Construction Site details page	None

4.3	Edit construction	4.3.1 Display Construction Site Details Page Task (Angular)	None	Construction Site details page	None
		4.3.2 Display Update Construction Site details page Task (Angular)	None	Update Construction Site details page	None
		4.3.3 Capture updated Construction Site details Task (Angular)	None	None	None
		4.3.4 Validate captured Construction site details Task (Angular)	None	None	None
		4.3.4 ALT Display Validation error on Edit Construction site Task (Angular)	None	Validation error notification	None
		4.3.5 Display confirmation modal Task (Angular)	None	Update Confirmation modal	None
		4.3.6 Update Construction site details Task (Angular)	Construction Site	None	None
		4.3.7 Notification on successfully updated Construction Site Details Task (Angular)	None	Updated Construction site notification	None
4.4	Allocate Team Site	4.4.1 Display "Project Page" Task (Angular and .Net Core)	None	Project page	None
		4.4.2 Display Search Allocated teams page Task (Angular)	None	Search Allocated Teams page	None
		4.4.3 Display Allocated Team Page Task (Angular)	Team Construction Site, Team	Allocate Team Page	None
		4.4.4 Capture Allocate teams details Task (Angular .Net Core)	None	None	None
		4.4.5 Validate captured details Task (Angular)	None	None	None
		4.4.6 ALT Display Validation error Task (angular)	None	Validation Error notification	None

		4.4.6 Display Confirmation modal Task (Angular)	None	Allocate Team confirmation modal	None
		4.4.7 Allocate Team to construction site Task (Angular .Net Core)	Team Construction Site	None	None
		4.4.8 Display notification on successfully allocated team Task (Angular)	None	Successfully allocated team notification	None
		4.4.9 Generate Successfully allocated team message Task (Angular)	None	None	None
		4.4.10 Retrieve Employee emails and number Task (Angular .Net Core)	Employee	None	None
		4.4.11 Notify employee on team site allocation Task (Angular .Net Core)	None	None	None
		4.4.12 Display Modal on successfully notified employees Task (Angular)	None	Successfully notified employee notification	None
4.5	Search Allocated Team	4.5.1 Display Construction Dropdown Task (Angular)	None	Home Page	None
		4.5.2 Display Search Allocated Teams Page Task (Angular .Net Core)	Team Construction Site, Team, Construction Site	Search Allocated teams page	None
		4.5.3 Capture Allocated teams Search criteria Task (Angular)	None	None	None
		4.5.4 Validate Allocated teams search criteria Task (Angular)	None	None	None
		4.5.5 ALT Display Validation on search criteria Task (angular)	None	Validation error notification	None
		4.5.5 Capture allocated teams information Task (Angular .Net Core)	Team Construction Site, Team	None	None

		4.5.6 ALT Display Search Allocated team error Task (Angular)	None	Search Error notification	None
		4.5.6 Display searched allocated team information Task (Angular)	None	Search Allocated teams page	None
		4.5.7 Display Allocated Team Page Task (Angular)	None	Allocated Teams page	None
4.6	Update Allocated Team	4.6.1 Display Allocated team page Task (Angular)	None	Allocated Teams page	None
		4.6.2 Enable Allocated team page inputs Task (Angular)	None	Updated Allocated Teams page	None
		4.6.3 Capture Updated Allocated team info Task (Angular)	None	None	None
		4.6.4 Validate Captured Allocated team info Task (Angular)	None	None	None
		4.6.5 ALT Display Validation error on updated allocated team info Task (Angular)	None	Validation error notification	None
		4.6.5 Display confirmation modal Task (Angular)	None	Update Confirmation modal	None
		4.6.6 Update Allocated Team Info Task (Angular .NET Core)	Team Construction Site	None	None
		4.6.7 Display Notification on successfully updated Allocated team info Task (Angular)	None	Successfully updated team notification	None
		4.6.8 Generate Successfully updated Team Allocation messages Task (Angular .NET Core)	None	None	None
		4.6.9 Retrieve Employee emails and number Task (.Net Core)	Employee	none	None
		4.6.10 Notify Employees on team allocation	None	None	None

		update Task (Angular .NET Core)			
		4.6.11 Display Modal on Successfully notified employees Task (Angular)	None	Successfully notified employees notification	None
4.7	Upload Compliance Document	4.7.1 Display Project page Task (Angular)	None	Project Page screen	None
		4.7.2 Capture Compliance Document Information Task (Angular .Net Core)	Compliance Document	None	None
		4.7.3 Display Compliance Documents Page Task (Angular)	None	Compliance Document page	None
		4.7.4 ALT No compliance Documents upload notification Task (Angular)	None	No Compliance Document notification	None
		4.7.4 Display Upload Compliance Documents Page Task (Angular)	None	Upload Compliance document page	None
		4.7.5 Capture Compliance Document Information Task (Angular)	None	None	None
		4.7.6 Validate Compliance Document Info Task (Angular)	None	None	None
		4.7.7 ALT Display Validation error on entered info Task (Angular)	None	Validation error notification	None
		4.7.7 Display Confirmation modal Task (Angular .Net Core)	None	Upload compliance document confirmation	None
		4.7.8 Upload Compliance Document Info Task (Angular .Net)	Compliance Document	None	None
		4.7.9 Display Notification modal on successfully uploaded compliance document Task (Angular)	None	Successfully upload compliance document notification	None

4.8	Search Compliance Document	4.8.1 Display Construction dropdown (Angular)	None	Home Page	None
		4.8.2 Display Search Compliance Document page (Angular)	None	Search Compliance Document Page	None
		4.8.3 ALT Display Compliance Document page (Angular)	None	Compliance Document page	None
		4.8.3 Capture Compliance Document Search criteria (Angular .Net Core)	Project, Compliance Document	Search Compliance document page	None
		4.8.4 Validate Compliance document search criteria (Angular)	None	None	None
		4.8.5 ALT Display Validation Error on Search criteria	None	Validation Error notification	None
		4.8.5 Capture Compliance Document information (Angular .Net Core)	Compliance Document	None	None
		4.8.5 ALT Display Search Error validation (Angular)	None	Search Error notification	None
		4.8.6 Display Searched compliance Document info (Angular)	None	Search Compliance Document Page	None
		4.8.7 Display compliance document page (Angular)	None	Compliance Document page	None
4.9	Search Project	4.9.1 Display Construction Dropdown Task (Angular)	None	Home Page	None
		4.9.2 Display "Search Project Page" Task (Angular and .NET Core)	Project	Search Project page (home screen)	None
		4.9.3 ALT Display Project page Task (Angular)	None	Project page	None
		4.9.3 Capture Project Search Criteria Task (Angular)	None	None	None
		4.9.4 Validate Project Search Criteria Task (Angular)	None	None	None

		4.9.5 ALT Display Validation error on search criteria Task (Angular)	None	Validation Error notification	None
		4.9.5 Capture Project Information Task (Angular and .NET Core)	Project	None	None
		4.9.6 ALT Display Search Error Task (Angular)	None	Search error notification	None
		4.9.6 Display Searched Project result Task (Angular)	None	Search Project page	None
		4.9.7 Display "Project Page" Task (Angular)	None	Project page	None
4.10	Create Site Progress description	4.10.1 Display Site Progress description page Task (Angular)	None	Site Progress description page	None
		4.10.2 Display Create new Site Progress description page Task (Angular)	None	Create new site progress description page	None
		4.10.3 Capture Site Progress description information Task (Angular .Net Core)	None	None	None
		4.10.4 Validate Site Progress description information Task (Angular)	None	None	None
		4.10.5 ALT Display Validation Error (angular)	None	Validation error notification	None
		4.10.5 Display Confirmation modal Task (Angular)	None	Create Site progress description confirmation modal	None
		4.10.6 Save Site Progress description information Task (Angular .Net Core)	Site Progress Description, Construction Photo	None	None
		4.10.7 Display Successful capture notification Task (Angular)	None	Successfully captured notification	None

		4.10.8 Display Site Progress Task description page Task (Angular)	None	Site Progress description page	None
4.11	View Site Progress description	4.11.1 Display Project page Task (Angular)	None	Project page	None
		4.11.2 capture Site Progress Description Information Task (Angular and .NET Core)	Site Progress Description, Construction Photo	None	None
		4.11.3 ALT Display No site progress description modal Task (Angular)	None	No Site progress description notification	None
		4.11.3 Display Site Progress Description Page Task (Angular)	None	Site Progress description page	None
4.12	Upload complete site photos	4.12.1 Display Project page Task (angular)	None	Project page	None
		4.12.2 Check for existing complete site photos Task (angular .Net Core)	Completion Photo	None	None
		4.12.3 Upload Complete site photos confirmation Task (angular)	None	Confirmation upload initial complete site photos	None
		4.12.4 Display Capture complete site photos modal Task (angular)	None	Capture complete site photos modal	None
		4.12.5 Capture complete site photos Task (angular)	None	None	None
		4.12.6 Validate Complete site photos Task (angular)	None	None	None
		4.12.7 ALT Display Validation Error Task (angular)	None	Validation Error notification	None
		4.12.7 Display confirmation modal Task (angular)	None	Upload Complete site photos confirmation	None
		4.12.8 Upload complete site photos Task (angular .Net Core)	Completion Photo	None	None
		4.12.9 Update Project Status Task (angular .Net Core)	Project	None	None

		4.12.10 Generate email notifying on complete site photos uploaded Task (angular .Net Core)	None	None	None
		4.12.11 Retrieve employee emails Task (angular .Net Core)	Employee	None	None
		4.12.12 Email employees Task (angular .Net Core)	None	None	None
		4.12.13 Display Notification on successfully uploaded complete site photos Task (angular)	None	Successfully upload complete site photos notification	None
4.13	Search complete site photos	4.13.1 Display Project Page Task (Angular)	None	Project Page	None
		4.13.2 Check for existing complete site photos Task (Angular .Net Core)	Completion Photo	None	None
		4.13.3 ALT No Display captured complete site photos Task (Angular)	None	No Captured Complete site photos Notification	None
		4.13.3 Capture complete site photos Task (Angular .Net Core)	Completion Photo	None	None
		4.13.4 Display complete site photos modal Task (Angular)	None	Complete site photos modal	None
4.14	Upload Radio Frequency report	4.14.1 Display Project Page Task (Angular)	None	Project page	None
		4.14.2 Capture Existing Radio Frequency report information Task (Angular .Net Core)	Radio Frequency Report	None	None
		4.14.3 Display Upload Radio Frequency report page Task (Angular)	None	Upload Radio frequency report page	None
		4.14.4 capture Radio Frequency report information Task (Angular.Net Core)	Radio Frequency Reports	None	None
		4.14.5 Validate Radio Frequency Report	None	None	None

		Information Task (Angular)			
		4.14.6 ALT Display Validation Error Task (Angular)	None	Validation error notification	None
		4.14.6 Display confirmation modal Task (Angular)	None	Upload Radio Frequency report confirmation modal	None
		4.14.7 Upload Radio Frequency Report Information Task (Angular .Net Core)	Radio Frequency Reports, Construction Site	None	None
		4.1.8 Update Project Status Task (Angular .Net Core)	Project	None	None
		4.14.9 Display Notification on successfully uploaded Radio Frequency report Task (Angular)	None	Successfully uploaded Radio Frequency report notification	None
		4.14.10 Generate Uploaded Radio Frequency report email Task (Angular.Net Core)	None	None	None
		4.14.11 Retrieve Employee email information Task (Angular.Net Core)	Employee	None	None
		4.14.12 Email Employees Task (Angular .Net Core)	None	None	None
		4.14.13 Display notification on emails sent to employees Task (Angular)	None	Emailed employees notification	None
4.15	Search Radio Frequency report	4.15.1 Display Construction dropdown Task (Angular)	None	Home Page	None
		4.15.2 Display Search Radio Frequency reports page Task (Angular .Net Core)	Project, Radio Frequency Reports	Search Radio frequency report page	None

		4.15.3 Capture Radio Frequency Search criteria Task (Angular)	None	None	None
		4.15.4 Validate Radio Frequency Report Info Task (Angular)	None	None	None
		4.15.5 ALT Display Validation error on search criteria Task (angular)	None	Validation Error notification	None
		4.15.5 Capture Radio Frequency Report Information Task (Angular .Net Core)	Radio Frequency Reports, Project, Construction Site	None	None
		4.15.6 ALT Display Search Error Task (Angular)	None	Search Error notification	None
		4.15.6 Display Searched Radio Frequency report information Task (Angular)	None	Search Radio frequency report page	None
		4.15.7 Display Radio Frequency Report Page Task (Angular)	None	Radio Frequency report page	None
4.16	Request Site Visit	4.16.1 Display Project page Task (Angular)	None	Project page	None
		4.16.2 Determine project status Task (Angular .Net Core)	Project, project Status	None	None
		4.16.3 Display confirmation modal requesting site visit inspection Task (Angular)	None	Confirmation to request site visit modal	None
		4.16.4 Generate email on site acceptance document created Task (Angular .Net Core)	None	None	None
		4.16.5 Retrieve employee emails Task (Angular .Net Core)	Employee	None	None
		4.16.6 Email Quality Control Officer on site acceptance document	None	None	None

		and Site visit Task (Angular .Net Core)			
		4.16.7 Update Project Status Task (Angular .Net Core)	Project	None	None
		4.16.8 Display notification on emailed site visit requested Task (Angular)	None	Successfully requested site visit notification	None
4.17	Create Snag	4.17.1 Display Construction Task dropdown (Angular)	None	Home Page	None
		4.17.2 Display snag details page Task (Angular .Net Core)	Snag	Snag details page	None
		4.17.3 Display create snag page Task (Angular)	None	Create snag details page	None
		4.17.4 Capture snag details Task (Angular)	None	None	None
		4.17.5 Validate captured snag information Task (Angular)	None	None	None
		4.17.6 ALT Display Validation Error Task (Angular)	None	Validation Error notification	None
		4.17.6 Display confirmation modal Task (Angular)	None	Confirmation to create snag details modal	None
		4.17.7 Create snag details Task (Angular .Net Core)	Snag	None	None
		4.17.8 Display notification on successfully created snag details Task (Angular)	None	Successfully created snag notification	None
4.18	Search Snag	4.18.1 Display Construction dropdown Task (Angular)	None	Home Page	None
		4.18.2 Display snag details page Task (Angular .Net Core)	Snag	Snag details page	None
		4.18.3 ALT Display Snag details model Task (Angular)	None	Snag item page	None

		4.18.3 Capture snag search criteria Task (Angular)	None	None	None
		4.18.4 Validate snag search criteria Task (Angular)	None	None	None
		4.18.5 ALT Display validation error on search criteria Task (Angular)	None	Validation Error notification	None
		4.18.5 Capture snag information Task (Angular .Net Core)	Snag	None	None
		4.18.6 ALT Display Search Error Task (Angular)	None	Search Error notification	None
		4.18.6 Display Searched snag results Task (Angular)	None	Snag details page	None
		4.18.7 Display snag item page Task (Angular)	None	Snag item page	None
4.19	Edit Snag	4.19.1 Display snag details page Task (Angular)	None	Snag item page	None
		4.19.2 Capture Updated snag details Task (Angular)	None	None	None
		4.19.3 Validate updated snag details Task (Angular)	None	None	None
		4.19.4 ALT Display Validation Error on edit snag details Task (Angular)	None	Validation error notification	None
		4.19.4 Display confirmation modal Task (Angular)	None	Edit snag confirmation modal	None
		4.19.5 Edit snag details Task (Angular .Net Core)	Snag	None	None
		4.19.6 Display Notification on successfully updated snag details Task (Angular)	None	Successfully edit snag notification	None

4.20	Edit Acceptance Document	4.20.1 Display Acceptance Document Page Task (Angular)	None	Acceptance Document page	None
		4.20.2 Display Edit Acceptance Document Page Task (Angular)	None	Edit Acceptance Document page	None
		4.20.3 Capture updated Acceptance Document information Task (Angular)	None	None	None
		4.20.4 Validate Acceptance Document Information Task (Angular)	None	None	None
		4.20.5 Display Acceptance Document Snag Information Task (Angular)	None	Acceptance Documents snag modal	None
		4.20.6 Capture Acceptance Document Snag Information Task (Angular)	None	None	None
		4.20.7 Validate Acceptance Document Snag Information Task (Angular)	None	None	None
		4.20.8 Display Confirmation Modal Task (Angular)	None	Edit Acceptance document confirmation modal	None
		4.20.9 Update Acceptance Document Information Task (Angular .Net Core)	Acceptance Document, Project Snag List	None	None
		4.20.10 Generate Updated Acceptance Document Email Task (Angular .Net Core)	None	None	None
		4.20.11 Retrieve Employee Emails Task (Angular)	Employee	None	None
		4.20.12 Email Operations Manager on updated Acceptance Document Task (Angular)	None	None	None

		4.20.13 Display Notification on successfully updated Acceptance Document Task (Angular)	None	Successfully updated acceptance document notification	None
4.21	Search Acceptance document	4.21.1 Display Construction dropdown Task (Angular)	None	Home Page	None
		4.21.2 Display Acceptance Document Page Task (Angular .Net Core)	Acceptance Document	Search Acceptance document page	None
		4.21.3 ALT Display Site Acceptance Document Page Task (Angular)	None	Acceptance Document page	None
		4.21.3 Capture Acceptance Document Search criteria Task (Angular)	None	None	None
		4.21.4 Validate Acceptance Document Search criteria Task (Angular)	None	None	None
		4.21.5 ALT Display validation error on search criteria Task (Angular)	None	Validation Error notification	None
		4.21.5 Capture Acceptance Document Information Task (Angular .Net Core)	Acceptance Document	None	None
		4.21.6 ALT Display Search Error Task (Angular)	None	Search Error notification	None
		4.21.6 Display Acceptance Document Information Task (Angular)	None	Search Acceptance document page	None
		4.21.7 Display Acceptance Document Page Task (Angular)	Acceptance Document, Snag, Project Snag List	Acceptance Document page	None
4.22		4.22.1 Display Acceptance Document Page Task (Angular)	None	Acceptance Document page	None

	Confirm acceptance document	4.22.2 Capture Snags on Acceptance Document Task (Angular)	Project Snag List	None	None
		4.22.3 Display acceptance document snag modal Task (Angular)	None	Acceptance document snag modal	None
		4.22.4 Capture Complete snag info Task (Angular)	None	None	None
		4.22.5 validate Complete snag information Task (Angular)	None	None	None
		4.22.6 ALT Display Validation Error Task (Angular)	None	Validation Error notification	None
		4.22.6 Display Confirmation modal Task (Angular)	None	Acceptance Document confirmation modal	None
		4.22.7 Capture Confirmation on acceptance document Task (Angular .Net Core)	Acceptance Document	None	None
		4.22.8 Update Project Status Task (Angular .Net Core)	Project	None	None
		4.22.9 Generate Email on Confirmed Acceptance Document Task (Angular .Net Core)	None	None	None
		4.22.10 Retrieve employee emails Task (.Net Core)	Employee	None	None
		4.22.11 Email employees on confirmed acceptance document Task (Angular .Net Core)	None	None	None
		4.22.12 Display notification on confirmed Acceptance Document Task (Angular)	None	Confirmed Acceptance document notification	None
4.23	Capture Site Sign-off	4.23.1 Display Project Page Task (Angular)	None	Project Page	None
		4.23.2 Check for existing site sign-offs Task (Angular .Net Core)	Acceptance Schedule	None	None

		4.23.3 Prompt confirmation to capture site sign-off Task (Angular)	None	Capture site sign off confirmation modal	None
		4.23.4 Display Capture Site -sign-off page Task (Angular)	None	Capture Site Sign off page	None
		4.23.5 Capture Site sign-off details Task (Angular)	None	None	None
		4.23.6 Validate site sign-off details Task (Angular)	None	None	None
		4.23.7 Display Confirmation modal Task (Angular)	None	Upload Site sign off confirmation modal	None
		4.23.8 Upload Site -sign-off information Task (Angular .Net Core)	Acceptance Schedule	None	None
		4.23.9 Update Project Status Task (Angular .Net Core)	Project	None	None
		4.23.10 Capture Site Acceptance Document Task (Angular .Net Core)	Acceptance Document	None	None
		4.23.11 Merge Acceptance Document and site sign off document Task (Angular .Net Core)	None	None	None
		4.23.12 Update Acceptance Document Task (Angular .Net Core)	Acceptance Document	None	None
		4.23.13 Generate Email on Site sign-off Task (Angular .Net Core)	None	None	None
		4.23.14 Retrieve Employee emails Task (Angular .Net Core)	Employee	None	None
		4.23.15 Email Employees on site sign-off Task (Angular .Net)	None	None	None
		4.23.16 Display notification on successfully emailed employees Task (Angular)	None	Successfully emailed employees notification	None

4.24	Search Site Sign-off	4.24.1 Display Construction Dropdown Task (Angular)	None	Home Page	None
		4.24.2 Display site Sign-offs page Task (Angular .Net Core)	Acceptance Schedule, Construction Site	Site sign-offs page	None
		4.24.3 ALT Display Site Sign-off page Task (Angular)	None	Site sign-off page	None
		4.24.3 Capture Site sign-offs search criteria Task (Angular)	None	None	None
		4.24.4 Validate site Sign-off search criteria Task (Angular)	None	None	None
		4.24.5 ALT Display Validation error on search criteria Task (Angular)	None	Validation Error notification	None
		4.24.5 Capture Site sign-off information Task (Angular .Net Core)	Acceptance Schedule, Construction Site	None	None
		4.24.6 ALT Display Search Error Task (Angular)	None	Search Error notification	None
		4.24.6 Display Site sign-off info Task (Angular)	None	Site sign-offs page	None
		4.24.7 Display site sign-off page Task (Angular)	None	site sign-off page	None
4.25	Capture Commission report	4.25.1 Display project page Task (Angular)	None	Project page screen	None
		4.25.2 Capture existing Commission report information Task (Angular .Net Core)	Commission report	None	None
		4.25.3 Display Commission reports page Task (Angular)	None	Commission reports page	None
		4.25.4 Display upload Commission reports page Task (Angular)	None	Upload Commission report page	None

		4.25.5 Capture Commission report information Task (Angular)	None	None	None
		4.25.6 Validate Commission report information Task (Angular)	None	None	None
		4.25.7 ALT Display Validation error Task (Angular)	None	Validation Error notification	None
		4.25.7 Display confirmation modal Task (Angular)	None	Upload Commission report confirmation	None
		4.25.8 Upload Commission report information Task (Angular .Net Core)	Commission report	None	None
		4.25.9 Display Notification modal on successfully uploaded Commission report Task (Angular)	None	Successfully uploaded Commission report page	None
4.26	Search Commission report	4.26.1 Display Construction Dropdown Task (Angular)	None	Home Page	None
		4.26.2 Display Commissions report page Task (Angular .Net Core)	Project, Commission report	Commission reports page	None
		4.26.3 Capture commission report search criteria Task (Angular)	None	None	None
		4.26.4 Validate Commission report search criteria Task (Angular)	None	None	None
		4.26.5 ALT Display Validation Error Information Task (Angular)	None	Validation Error notification	None
		4.26.5 Capture Commission report information Task (Angular .Net Core)	Project, Commission report	None	None

		4.26 ALT Display Search result Error Task (Angular)	None	Search Error notification	None
		4.26.6 Display Searched commission report information Task (Angular)	None	Commission reports page	None
		4.26.7 Display Commissioner report page Task (Angular)	None	Commission report page	None
4.27	Create Site Type	4.27.1 Display Construction Dropdown (Angular)	None	Hope Page	None
		4.27.2 Display Site type details page (Angular .Net Core)	Site Type	Site type details page	None
		4.27.3 Display new site type modal (Angular)	None	New Site type modal	None
		4.27.4 Capture Site type details (Angular)	None	None	None
		4.27.5 Validate Captured site type information (Angular)	None	None	None
		4.27.6 ALT Display Validation Error (Angular)	None	Validation Error notification	None
		4.27.6 Display Confirmation modal (Angular)	None	Create new site type confirmation modal	None
		4.27.7 Create site type details (Angular .Net Core)	Site Type	None	None
		4.27.8 Display Notification on successfully created site type details (Angular)	None	Successfully create site type notification	None
4.28	Search Site type	4.28.1 Display Construction Dropdown Task (Angular)	None	Home page	None
		4.28.2 Display Site Type details page Task (Angular and .NET Core)	Site Type	Site type details page	None
		4.28.3 Capture Site type search criteria Task (Angular)	None	None	None

		4.28.3 ALT Display Site type details modal Task (Angular)	None	Site type details modal	None
		4.28.4 Validate site type search criteria Task (Angular)	None	None	None
		4.28.5 Capture Site type information Task (Angular and .NET Core)	Site Type	None	None
		4.28.5 ALT Display validation Error on search criteria Task (angular)	None	Validation Error notification	None
		4.28.6 Display searched site type results Task (Angular)	None	Site type details page	None
		4.28.6 ALT Display Search Error Task (Angular)	None	Search Error notification	None
		4.28.7 Display Site type details modal Task (Angular)	None	Site type details modal	None
4.29	Edit Site Type	4.29.1 Display Site type details modal Task (Angular)	None	Site type details modal	None
		4.29.2 Capture updated Site type details Task (Angular)	None	None	None
		4.29.3 Validate updated site type details Task (Angular)	None	None	None
		4.29.4 Display Confirmation Modal Task (Angular)	None	Edit Site type details notification	None
		ALT 4.29.4 Display Validate error on edit information Task (Angular)	None	Validation Error notification	None
		4.29.5 Edit Site type details Task (Angular and .NET Core)	Site Type	None	None
		4.29.6 Display Notification on successfully updated site type details Task (Angular)	None	Successfully edited site type notification	None

10.1.5 Invoicing Subsystem

5. Invoicing Subsystem						
Requirement	Use Case	Use Case Title	Primitive DFD Processes	Entities	Screens	Report
5.1 Search Invoice	5.1	Search Invoice	5.1.1 Retrieve Invoice detail Task (Angular .NET Core)	Invoice	None	None
			5.1.2 Display Search Invoice Screen Task (Angular)	None	Search Invoice Screen	None
			5.1.3 Capture Search Result Task (Angular)	None	None	None
			5.1.4 Validate Search Parameter Task (Angular)	None	None	None
			ALT 5.1.5 Display Validation Error Dialog Task (Angular)	None	Validation Error Dialog	None
			5.1.5 Retrieve Search Result Task (Angular .NET Core)	Invoice	None	None
			5.1.6 Populate Result Table Task (Angular)	None	None	None
			5.1.7 Capture Download Request Task (Angular)	None	None	None
			5.1.8 Retrieve Invoice Task (Angular .NET Core)	Invoice	None	None
			5.1.9 Send Invoice Task (Angular)	None	None	None
5.2 Send Invoice	5.2	Send Invoice	5.2.1 Display Select Invoice Type Screen Task (Angular .NET Core)	None	Select Invoice Type Screen	None
			5.2.2 Capture Selected Type Task (Angular)	None	None	None

			5.2.3 Retrieve Client Purchase Order Details Task (Angular .NET Core)	Client Purchase Order	None	None
			5.2.4 Generate Initial Invoice Task (Angular)	None	None	None
			ALT 5.2.4 Generate Final Invoice Task (Angular)	None	None	None
			5.2.5 Display Confirm Invoice Modal Task (Angular)	None	Confirm Invoice Modal	None
			5.2.6 Capture Invoice Confirmation Task (Angular)	Invoice	None	None
			5.2.7 Save Invoice Task (Angular .NET Core)	Invoice	None	None
			5.2.8 Retrieve Invoice Task (Angular .NET Core)	Invoice	None	None
			5.2.9 Send Invoice Task (Angular)	None	None	None
5.3 Capture Proof of Payment	5.3	Capture Proof of Payment	5.3.1 Retrieve Invoices Task (Angular .NET Core)	Invoice	None	None
			5.3.2 Display Capture Proof of Payment Screen Task (Angular)	None	Capture Proof of Payment Screen	None
			5.3.3 Capture Proof of Payment Details Task (Angular)	None	None	None
			5.3.4 Validate Provided Details Task (Angular)	None	None	None
			ALT 5.3.5 Display Validation Error	None	Validation Error Dialog	None

			Dialog Task (Angular)			
			5.3.5 Display Confirm Proof of Payment Modal Task (Angular)	Proof of Payment	Proof of Payment Modal	None
			5.3.6 Capture Confirmation Task (Angular)	None	None	None
			5.3.7 Save Proof of Payment Task (Angular .NET Core)	Proof of Payment	None	None
			5.3.8 Display Proof of Payment Captured Alert Task (Angular)	None	Proof of Payment Alert	None
5.4 Capture Claim	5.4	Capture Claim	5.4.1 Retrieve Invoice Details Task (Angular .NET Core)	Invoice	None	None
			5.4.2 Display Capture Claim Screen Task (Angular)	None	Capture Claim Screen	None
			5.4.3 Capture Claim Details Task (Angular)	Invoice	None	None
			5.4.4 Validate Provided Details Task (Angular)	None	None	None
			ALT 5.4.5 Display Validation Error Dialog Task (Angular)	None	Validation Error Dialog	None
			5.4.5 Display Claim Confirmation Modal Task (Angular)	None	Claim Confirmation Modal	None
			5.4.6 Capture Confirmation Task (Angular)	None	None	None
			5.4.7 Save Claim Details Task (Angular .NET Core)	Invoice	None	None

			5.4.8 Display Claim Captured Alert Task (Angular)	None	None	None
5.5 Search Proof of Payment	5.5	Search Proof of Payment	5.5.1 Retrieve Proof of Payment Details Task (Angular .NET Core)	Proof of Payment	None	None
			5.5.2 Display Search Proof of Payment Screen Task (Angular)	None	Search Proof of Payment Screen	None
			5.5.3 Capture Search Parameter Task (Angular)	None	None	None
			5.5.4 Validate Search Parameter Task (Angular)	None	None	None
			ALT 5.5.5 Display Validation Error Dialog Task (Angular)	None	Validation Error Dialog	None
			5.5.5 Retrieve Proof of Payment Record Task (Angular .NET Core)	Proof of Payment	None	None
			5.5.6 Populate Search Result Table Task (Angular)	None	None	None
			ALT 5.5.6 Display Result Not Found Label Control Task (Angular)	None	Search Proof of Payment Screen	None
			5.5.7 Capture View Request Task (Angular)	None	None	None
			5.5.8 Display View Proof of Payment Modal Task (Angular)	None	View Proof of Payment Modal	None
5.6 Search Invoice Type	5.6		5.6.1 Retrieve Invoice Type	Invoice Type	None	None

		Search Invoice Type	Details Task (Angular .NET Core)			
			5.6.2 Display Search Invoice Type Screen Task (Angular)	None	Search Invoice Type Screen	None
			5.6.3 Capture Search Parameter Task (Angular)	None	None	None
			5.6.4 Validate Search Parameter Task (Angular)	None	None	None
			ALT 5.6.5 Display Validation Error Dialog Task (Angular)	None	Validation Error Dialog	None
			5.6.5 Retrieve Search Result Task (Angular .NET Core)	Invoice	None	None
			5.6.6 Populate Search Result Table Task (Angular)	None	None	None
5.7 Create Invoice Type	5.7	Create Invoice Type	5.7.1 Retrieve Invoice Type Details Task (Angular .NET Core)	Invoice Type	None	None
			5.7.2 Display Create Invoice Type Screen Task (Angular)	None	Create Invoice Type	None
			5.7.3 Capture Create Request Task (Angular)	None	None	None
			5.7.4 Display Add Invoice Type Modal Task (Angular)	None	Add Invoice Type Modal	None
			5.7.5 Capture Invoice Type Information Task (Angular)	None	None	None
			5.7.6 Validate Invoice Type Information Task (Angular)	None	None	None

			ALT 5.7.7 Display Validation Error Dialog Task (Angular)	None	Validation Error Dialog	None
			ALT 5.7.7 Display Invoice Type Exists Modal Task (Angular)	None	Invoice Type Exists Modal	None
			5.7.7 Display Invoice Type Confirmation Modal Task (Angular)	None	Invoice Type Confirmation Modal	None
			5.7.8 Capture Confirmation Task (Angular)	None	None	None
			5.7.9 Save Invoice Type Task (Angular)	Invoice Type	None	None
			5.7.10 Display Invoice Type Created Alert Task (Angular)	None	Invoice Type Confirmation Alert	None

10.1.6 Human Resource Subsystem

6. Human Resource Subsystem					
Use Case No.	Use Case Title	Processes from Primitive Data Flow Diagram	Entities	Screens	Reports
6.1	Add Employee	6.1.1 Retrieve Employee Skills and Employee Type Information Task (Angular)	Employee Skills Employee Type	Employee Screen	None
		6.1.2 Display Add Employee Modal Task (Angular)	None	Add Employee Modal	None
		6.1.3 Validate Captured Details Task (Angular)	None	None	None
		6.1.3 ALT: Notification of Validation Error Task (Angular)	None	Validation Error Dialog	None
		6.1.4 Display Confirmation Dialog Task(Angular)	None	Confirmation Dialog	None
		6.1.5 Request confirmation of captured info Task(Angular)	None	None	None
		6.1.6 Save Employee Details Task (Angular .NET Core)	Employee	None	None
		6.1.7 Notification of successful employee addition Task (Angular)	None	Successful Employee Addition Modal	None
		6.1.7 ALT: Notification of unsuccessful employee addition Task (Angular)	None	Unexpected Error Dialog	None
6.2	Edit Employee	6.2.1 Retrieve Employee Information Task (Angular)	None	Employee Screen	None
		6.2.2 Retrieve Employee Skills and Employee Type Information Task (Angular .NET Core)	Employee Skills Employee Type	None	None
		6.2.3 Display Update Employee Modal Task (Angular)	None	Edit Employee Modal	None
		6.2.4 Validate Captured Details Task (Angular)	None	None	None

		6.2.4 ALT: Notification of Validation Error Task (Angular)	None	Validation Error Dialog	None
		6.2.5 Display Confirmation Dialog Task (Angular)	None	Confirmation Dialog	None
		6.2.6 Request confirmation of captured info Task (Angular)	None	None	None
		6.2.7 Save Changes to Employee Task (Angular .NET Core)	Employee	None	None
		6.2.8 Notification of successful employee update Task (Angular)	None	Successful Employee Edit Modal	None
		6.2.8 ALT: Notification of unsuccessful employee update Task (Angular)	None	Unexpected Error Dialog	None
6.3	Search Employee	6.3.1 Display Employee Screen Task (Angular)	None	Employee Screen	None
		6.3.2 Capture Search Parameter Task (Angular)	None	None	None
		6.3.3 Validate Captured Details Task (Angular)	None	None	None
		6.3.3 ALT: Notification of Validation Error Task (Angular)	None	Validation Error Dialog	None
		6.3.4 Retrieve Search Results Task(Angular .NET Core)	Employee	None	None
		6.3.5 Populate Search Results Table Task (Angular)	None	None	None
		6.3.5 ALT: Notification of no employee found Task (Angular)	None	No Employee Found Dialog	None
		6.3.6 Display Selected Employee Task (Angular)	None	View Employee Modal	None
		6.3.6 ALT: Notification of display error encountered Task (Angular)	None	Unexpected Error Dialog	None
6.4	Upload Employee Document	6.4.1 Display Upload Employee Compliance Document Modal Task (Angular)	None	Home Screen Employee Screen	None

				Upload Employee Compliance Document Modal	
		6.4.2 Retrieve Employee Information Task (Angular .NET Core)	None	None	None
		6.4.3 Capture Employee Document Information Task (Angular)	None	None	None
		6.4.4 Validate Employee Document Details Task (Angular)	None	None	None
		6.4.4 ALT: Notification of Validation Error (Angular)	None	Validation Error Dialog	None
		6.4.5 Display Confirmation Dialog Task(Angular)	None	Confirmation Dialog	None
		6.4.6 Save Changes of Employee Compliance Document Task (Angular .NET Core)	Employee Compliance Document	None	None
		6.4.7 Notification of successful upload Task (Angular)	None	Successful Document Upload Dialog	None
		6.4.7 ALT: Notification of unsuccessful upload Task (Angular)	None	Unexpected Error Dialog	None
6.5	Edit Employee Document	6.5.1 Retrieve Employee Document Information Task (Angular .NET Core)	None	Home Screen Employee Screen	None
		6.5.2 Display Update Employee Compliance Document Modal Task (Angular)	None	Update Employee Compliance Document Modal	None
		6.5.3 Capture Employee Document Details Task(Angular)	None	None	None
		6.5.4 Validate Captured Details Task (Angular)	None	None	None
		6.5.4 ALT: Notification of Validation Error Task (Angular)	None	Validation Error Dialog	None
		6.5.5 Display Confirmation Dialog Task (Angular)	None	Confirmation Dialog	None

		6.5.6 Request confirmation of captured info Task(Angular)	None	None	None
		6.5.7 Save Changes to Employee Task (Angular .NET Core)	Employee Compliance Document	None	None
		6.5.8 Notification of successful employee update Task (Angular)	None	Successful Document Update Dialog	None
		6.5.8 ALT: Notification of unsuccessful document update Task (Angular)	None	Unexpected Error Dialog	None
6.6	Search Employee Document	6.6.1 Display Search Compliance Document Modal Task (Angular)	None	Home Screen Employee Screen	None
		6.6.2 Capture Search Parameter Task(Angular)	None	None	None
		6.6.3 Validate Captured Details Task (Angular)	None	None	None
		6.6.3 ALT: Notification of Validation Error (Angular)	None	Validation Error Dialog	None
		6.6.4 Retrieve Search Results Task (Angular .NET Core)	Employee Compliance Document	None	None
		6.6.5 Populate Search Results Table Task(Angular)	None	None	None
		6.6.5 ALT: Notification of no document found Task (Angular)	None	None	None
		6.6.6 Display Selected Employee Task(Angular)	None	View Employee Document Modal	None
		6.6.6 ALT: Notification of display error encountered Task (Angular)	None	Unexpected Error Dialog	None
6.7	Create Account	6.7.1 Retrieve Employee Information Task (Angular .NET Core)	None	Account Screen	None
		6.7.2 Display Create Account Modal Task (Angular)	None	Create Account Modal	None
		6.7.3 Validate Captured Details Task(Angular)	None	None	None

		6.7.3 ALT: Notification of Validation Error (Angular)	None	Validation Error Dialog	None
		6.7.4 Display Confirmation Dialog Task (Angular)	None	Confirmation Dialog	None
		6.7.5 Request confirmation of captured info Task (Angular)	None	None	None
		6.7.6 Save User Details Task (Angular .NET Core)	User	None	None
		6.7.7 Notification of successful user addition Task(Angular)	None	Successful User Addition Dialog	None
		6.7.7 ALT: Notification of unsuccessful user addition Task (Angular)	None	Unexpected Error Dialog	None
6.8	Edit Account	6.8.1 Retrieve Employee Information Task (Angular .NET Core)	None	Account Screen	None
		6.8.2 Display Update Account Modal Task (Angular)	None	Update Account Modal	None
		6.8.3 Validate Captured Details Task (Angular)	None	None	None
		6.8.3 ALT: Notification of Validation Error (Angular)	None	Validation Error Dialog	None
		6.8.4 Display Confirmation Dialog Task (Angular)	None	Confirmation Dialog	None
		6.8.5 Request confirmation of captured info Task (Angular)	None	None	None
		6.8.6 Save Changes to User Task (Angular .NET Core)	User	None	None
		6.8.7 Notification of successful user update Task (Angular)	None	Successful User Addition Dialog	None
		6.8.7 ALT: Notification of unsuccessful user addition Task (Angular)	None	Unexpected Error Dialog	None
6.9	Search Account	6.9.1 Display Account Screen Task (Angular)	None	Account Screen	None
		6.9.2 Capture Search Parameter Task (Angular)	None	None	None

		6.9.3 Validate Captured Details Task (Angular)	None	None	None
		6.9.3 ALT: Notification of Validation Error (Angular)	None	Validation Error Dialog	None
		6.9.4 Retrieve Search Results Task (Angular .NET Core)	User	None	None
		6.9.5 Populate Search Results Table Task (Angular)	None	None	None
		6.9.5 ALT: Notification of no user found Task (Angular)	None	No User Found Dialog	None
		6.9.6 Display Selected User Task (Angular)	None	View User Modal	None
		6.9.6 ALT: Notification of display error encountered Task (Angular)	None	Unexpected Error Dialog	None
6.10	Create Employee Type	6.10.1 Display Employee Screen Task (Angular)	None	Employee Screen	None
		6.10.2 Display Add Employee Type Modal Task (Angular)	None	Add Employee Type Modal	None
		6.10.3 Capture Employee Type Information Task(Angular)	None	None	None
		6.10.4 Validate Employee Type Details Task (Angular)	None	None	None
		6.10.4 ALT: Notification of Validation Error (Angular)	None	Validation Error Dialog	None
		6.10.5 Display Confirmation Dialog Task (Angular)	None	Confirmation Dialog	None
		6.10.6 Save Changes of Employee Type Task (Angular .NET Core)	Employee Type	None	None
		6.10.7 Notification of successful creation Task (Angular)	None	Employee Type Creation Successful Modal	None
		6.10.7 ALT: Notification of unsuccessful creation Task (Angular)	None	Unexpected Error Dialog	None

6.11	Search Employee Type	6.11.1 Display Employee Screen Task (Angular)	None	Employee Screen	None
		6.11.2 Capture Search Parameter Task (Angular)	None	None	None
		6.11.3 Validate Captured Details Task (Angular)	None	None	None
		6.11.3 ALT: Notification of Validation Error (Angular)	None	Validation Error Dialog	None
		6.11.4 Retrieve Search Results Task (Angular .NET Core)	Employee Type	None	None
		6.11.5 Populate Search Results Table Task (Angular)	None	None	None
		6.11.5 ALT: Notification of no employee type found Task (Angular)	None	No Employee Type Found Dialog	None
		6.11.6 Display Selected Employee Type Task (Angular)	None	View Employee Type Modal	None
		6.11.6 ALT: Notification of display error encountered Task (Angular)	None	Unexpected Error Dialog	None
6.12	Create Employee Skill	6.12.1 Display Employee Screen Task (Angular)	None	Employee Screen	None
		6.12.2 Display Add Employee Skill Modal Task (Angular)	None	Add Employee Skill Modal	None
		6.12.3 Capture Employee Skill Information Task (Angular)	None	None	None
		6.12.4 Validate Employee Skill Details Task (Angular)	None	None	None
		6.12.4 ALT: Notification of Validation Error (Angular)	None	Validation Error Dialog	None
		6.12.5 Display Confirmation Dialog Task (Angular)	None	Confirmation Dialog	None
		6.12.6 Save Changes of Employee Skill Task (Angular .NET Core)	Employee Skill	None	None

		6.12.7 Notification of successful creation Task (Angular)	None	Successful Employee Type Creation Dialog	None
		6.12.7 ALT: Notification of unsuccessful creation Task (Angular)	None	Validation Error Dialog	None
6.13	Search Employee Skill	6.13.1 Display Employee Screen Task (Angular)	None	Employee Screen	None
		6.13.2 Capture Search Parameter Task (Angular)	None	None	None
		6.13.3 Validate Captured Details Task (Angular)	None	None	None
		6.13.3 ALT: Notification of Validation Error (Angular)	None	Validation Error Dialog	None
		6.13.4 Retrieve Search Results Task (Angular .NET Core)	Employee Skill	None	None
		6.13.5 Populate Search Results Table Task (Angular)	None	None	None
		6.13.5 ALT: Notification of no employee type found Task (Angular)	None	No Employee Skill Found Dialog	None
		6.13.6 Display Selected Employee Skill Task (Angular)	None	View Employee Skill Modal	None
		6.13.6 ALT: Notification of display error encountered Task (Angular)	None	Validation Error Dialog	None

10.1.7 Teams Subsystem

7. Team Subsystem						
Requirement	Use Case No.	Use Case Title	Processes from Primitive Data Flow Diagram	Entities from ERD	Screen	Report
7.1 Create Team	7.1	Create Team	7.1.1 Capture Team detail Task (Angular)	Employee	Create Team Screen	None
			7.1.2 Validate Captured details Task (Angular)	None	None	None
			7.1.3 ALT: Notification of Validation error Task (Angular)	None	Invalid Notification screen	None
			7.1.3 Confirmation of details Task (Angular)	None	None	None
			7.1.4 Save Team details Task (asp.Net Core)	Team, Employee Team	None	None
			7.1.5 Notification of successful save Task (Angular)	None	Success Notification Screen	None
			7.1.5 ALT: Notification of unsuccessful save Task (Angular)		Error Screen	None
7.2 Search Team	7.2	Search Team	7.2.1 Capture Team search parameter Task (Angular)	None	Teams Screen	None
			7.2.2 Validate search parameter Task (Angular)	None	None	None
			7.2.3 ALT: Notification of Validation error Task (Angular)	None	Invalid Notification screen	None
			7.2.3 Retrieve Team search parameter	Team	None	None
			7.2.4 Return Team search results Task (Angular)	None	None	None
			7.2.5 Return Selected Team Information Task (Angular)	Team, Employee Team	None	None

			7.2.5 ALT: Notification of No Team found Task (Angular)	None	Error Screen	None
			7.2.5(A) [ALT] Capture Add Team request Task (Angular)	None	None	None
			7.2.5(B) [ALT] Capture Update Team Request Task (Angular)	None	None	None
7.3 Edit Team	7.3	Edit Team	7.3.1 Retrieve and display Team details Task (Angular)	Team, Employee Team	Edit Team	None
			7.3.2 Request and capture edited Team info Task (Angular)	None	None	None
			7.3.3 Validate Captured Team Details Task (Angular)	None	None	None
			7.3.4 ALT: Notification of Validation error Task (Angular)	None	Invalid Notification screen	None
			7.3.4 Confirmation of updated Team info Task (Angular)	None	None	None
			7.3.5 Update Team info Task (Angular)	Team, Employee Team	None	None
			7.3.6 Notification of successful save Task (Angular)	None	Success Notification Screen	None
			7.3.6 ALT: Notification of unsuccessful save Task (Angular)	None	Error Screen	None

10.1.8 Supplier Order Subsystem

8. Supplier Order Subsystem						
Requirement	Use Case	Use Case Title	Primitive DFD Processes	Entities	Screens	Report
8.1 Search Project Line-Item List	8.1	Search Project Line-Item List	8.1.1 Retrieve Project Details Task (Angular .NET Core)	Project	None	None
			8.1.2 Display Search Project Line-Item List Screen Task (Angular)	None	Search Project Line-Item List Screen	None
			8.1.3 Capture Search Parameter Screen Task (Angular)	None	Search Parameter Screen	None
			8.1.4 Validate Search Parameter Task (Angular)	None	None	None
			8.1.5 Retrieve Search Result Task (Angular .NET Core)	Project	None	None
			ALT 8.1.5 Display Validation Error Dialog Task (Angular)	None	Validation Error Dialog	None
			8.1.6 Populate Search Result Table Task (Angular)	None	None	None
			ALT 8.1.6 Display Result Not Found Label Control Task (Angular)	None	Search Project Line-Item List Screen	None
			8.1.7 Capture View Request Task (Angular)	None	None	None
			8.1.8 Retrieve Project Line-Item List Task (Angular .NET Core)	Project Line-Item	None	None
			8.1.9 Display View Project Line-Item List	None	View Project	None

			Modal Task (Angular)		Line-Item List Modal	
8.2 Edit Project Line-Item List	8.2	Edit Project Line-Item List	8.2.1 Retrieve Project Line-Item Task (Angular .NET Core)	Project Line-Item	None	None
			8.2.2 Display Edit Project Line-item List Screen Task (Angular)	None	Edit Project Line-item List Screen	None
			8.2.3 Capture Quantities Task (Angular)	None	None	None
			8.2.4 Validate Quantities Task (Angular)	None	None	None
			ALT 8.2.5 Display Validation Error Dialog Task (Angular)	None	Validation Error Dialog	None
			8.2.5 Display Confirm Project Line-Item List Modal Task (Angular)	None	Confirm Project Line-Item List Modal	None
			8.2.6 Capture Confirmation Task (Angular)	None	None	None
			8.2.7 Save Project Line-Item List Task (Angular .NET Core)	Project Line-Item	None	None
			8.2.8 Display Project Line-Item List Updated Alert Task (Angular)	None	Project Line-Item List Updated Alert	None
			ALT 8.2.4 Display Remove Project Line-Item Modal Task (Angular)	None	Remove Project Line-Item Modal	None
			ALT 8.2.5 Capture Delete Request Task (Angular)	None	None	None
			ALT 8.2.5 Delete Project	Project Line-Item	None	None

			Line-Item Task (Angular .NET Core)			
			ALT 8.2.6 Display Project Line-Item Removed Alert Task (Angular)	None	Project Line-Item Removed Alert	None
			ALT 8.2.4 Retrieve Line-Item Details Task (Angular .NET Core)	Line-Item	None	None
			ALT 8.2.5 Display Add Project Line-Item Screen Task (Angular)	None	Add Project Line-Item Screen	None
			ALT 8.2.6 Capture Line-Item Information Task (Angular)	None	None	None
			ALT 8.2.7 Display Confirm Additional Line-Items Modal Task (Angular)	None	Confirm Additional Line-Items Modal	None
			ALT 8.2.8 Capture Confirmation Task (Angular)	None	None	None
			ALT 8.2.9 Save Project Line-Item List Task (Angular .NET Core)	Project Line-Item	None	None
			ALT 8.2.10 Display Line-Items Added Alert Task (Angular)	None	Line-Items Added Alert	None
8.3 Create Supplier Order	8.3	Create Supplier Order	8.3.1 Retrieve Project line-item list Details Task (Angular .NET Core)	Project line-item	None	None
			8.3.1 Retrieve Supplier Details Task (Angular .NET Core)	Supplier	None	None
			8.3.3 Display Create Supplier	None	Supplier Purchase	None

			Purchase Order Screen Task (Angular)		Order Screen	
			8.3.4 Capture Supplier Assignments Task (Angular)	None	None	None
			8.3.5 Validate Supplier Assignments Task (Angular)	None	None	None
			ALT 8.3.6 Display Validation Error Dialog Task (Angular)	None	Validation Error Dialog	None
			8.3.6 Display Confirm Supplier Purchase Orders Modal Task (Angular)	None	Confirm Supplier Purchase Orders Modal	None
			8.3.7 Capture Confirmation Task (Angular)	None	None	None
			ALT 8.3.8 Display Supplier Purchase Orders Not Created Alert Task (Angular)	None	Supplier Purchase Orders Not Created	None
			8.3.8 Save Supplier Purchase Orders Task (Angular .NET Core)	Supplier Purchase Order	None	None
			8.3.9 Display Supplier Purchase Orders Created Alert Task (Angular)	None	Supplier Purchase Orders Created Alert	None
			8.3.10 Retrieve Supplier Purchase Orders Task (Angular .NET Core)	Supplier Purchase Order	None	None
			8.3.11 Send Supplier Purchase	None	None	None

			Orders Task (Angular)			
8.4 Capture Supplier Quote	8.4	Capture Supplier Quote	8.4.1 Retrieve Supplier Details Task (Angular .NET Core)	Supplier		None
			8.4.2 Display Capture Supplier Quote Screen Task (Angular)	None	Capture Supplier Quote Screen	None
			8.4.3 Capture Supplier Quote Information Task (Angular)	None	None	None
			8.4.4 Validate Captured Information Task (Angular)	None	None	None
			ALT 8.4.5 Display Validation Error Dialog Task (Angular)	None	Validation Error Dialog	None
			8.4.5 Display Confirm Supplier Quote Modal Task (Angular)	Supplier Quote	Confirm Supplier Quote Modal	None
			8.4.6 Capture Confirmation Task (Angular)	None	None	None
			8.4.7 Save Supplier Quote Task (Angular .NET Core)	Supplier Quote	None	None
			8.4.8 Display Supplier Quote Captured Alert Task (Angular)	None	Supplier Quote Captured Alert	None
8.5 Search Supplier Quote	8.5	Search Supplier Quote	8.5.1 Retrieve Supplier Quotes Task (Angular .NET Core)	Supplier Quote	None	None
			8.5.2 Display Search Supplier Quote Screen Task (Angular)	None	Search Supplier Quote Screen	None
			8.5.3 Capture Search Parameter Task (Angular)	None	None	None

			8.5.4 Validate Search Parameter Task (Angular)	None	None	None
			ALT 8.5.5 Display Validation Error Dialog Task (Angular)	None	Validation Error Dialog	None
			8.5.5 Retrieve Supplier Information Task (Angular .NET Core)	Supplier	None	None
			8.5.6 Retrieve Project Information Task (Angular .NET Core)	Project	None	None
			8.5.7 Populate Search result Table Task (Angular)	None	None	None
			8.5.8 Capture Download request Task (Angular)	None	None	None
			8.5.9 Retrieve Supplier Quote Task (Angular .NET Core)	Supplier Quote	None	None
			8.5.10 Send Supplier Quote Task (Angular)	None	None	None
8.6 Search Supplier Purchase Order	8.6	Search Supplier Purchase Order	8.6.1 Retrieve Supplier Purchase Orders Task (Angular .NET Core)	Supplier Purchase Order	None	None
			8.6.2 Display Search Supplier Purchase Order Screen Task (Angular)	None	Search Supplier Purchase Order Screen	None
			8.6.3 Capture Search Parameter Task (Angular)	None	None	None
			8.6.4 Validate Search	None	None	None

			Parameter Task (Angular)			
			ALT 8.6.5 Display Validation Error Dialog Task (Angular)	None	Validation Error Dialog	None
			8.6.5 Retrieve Supplier Information Task (Angular .NET Core)	Supplier	None	None
			8.6.6 Retrieve Project Information Task (Angular .NET Core)	Project	None	None
			8.6.7 Populate Search result Table Task (Angular)	None	None	None
			8.6.8 Capture Download Request Task (Angular)	None	None	None
			8.6.9 Retrieve Supplier Purchase Order Task (Angular .NET Core)	Supplier Purchase Order	None	None
			8.6.10 Send Supplier Purchase Order Task (Angular)	None	None	None

10.1.9 Supplier Subsystem

9. Supplier Subsystem						
Requirement	Use Case No.	Use Case Title	Processes from Primitive Data Flow Diagram	Entities from ERD	Screen	Report
9.1 Add Supplier	9.1	Add Supplier	9.1.1 Capture Supplier details (Angular)	None	Create Supplier screen	None
			9.1.2 Validate Captured details (Angular)	None	None	None
			9.1.3 ALT: Notification of Validation error Task (Angular)	None	Invalid Notification screen	None
			9.1.3 Confirmation of details (Angular)	None	None	None
			9.1.4 Save Supplier details (asp.Net Core)	Supplier, Supplier document	None	None
			9.1.5 Notification of successful save Task (Angular)	None	Success Notification Screen	None
			9.1.5 ALT: Notification of unsuccessful save Task (Angular)	None	Error Screen	None
9.2 Search Supplier	9.2	Search Supplier	9.2.1 Capture Supplier search parameter (Angular)	None	Supplier Screen	None
			9.2.2 Validate search parameter (Angular)	None	None	None
			9.2.3 ALT: Notification of Validation error Task (Angular)	None	Invalid Notification screen	None
			9.2.3 Retrieve Supplier search parameter (Angular)	Supplier, Supplier document	None	None
			9.2.4 Return Supplier search results (Angular)	None	None	None
			9.2.5 Return Selected Supplier	None	None	None

			information (Angular)			
			9.2.5 ALT: Notification of No Supplier found (Angular)	None	Error Screen	None
			9.2.5(A) [ALT] Capture Add Supplier request Task (Angular)	None	None	None
			9.2.5(B) [ALT] Capture Update Supplier request Task (Angular)	None	None	None
9.3 Update Supplier	9.3	Edit Supplier	9.3.1 Display Supplier details (Angular)	None	Edit Supplier Screen	None
			9.3.2 Request and capture edited Supplier info (Angular)	None	None	None
			9.3.3 Validate Captured Supplier details (Angular)	None	None	None
			9.3.4 ALT: Notification of Validation error Task (Angular)	None	Invalid Notification screen	None
			9.3.4 Confirmation of updated Supplier info (Angular)	None	None	None
			9.3.5 Update Supplier info (asp.Net Core)	Supplier, Supplier document	None	None
			9.3.6 Notification of successful save Task (Angular)	None	None	None
			9.3.6 ALT: Notification of unsuccessful save Task (Angular)	None	Error Screen	None

10.1.10 Inventory Management Subsystem

10. Inventory Management						
Requirement	Use Case No.	Use Case Title	Processes from Primitive Data Flow Diagram	Entities from ERD	Screen	Report
10.1 Add Line-Item	10.1	Add Line-Item	10.1.1 Capture Line-Item details Task (Angular)	None	Create Line-item screen	None
			10.1.2 Validate Captured details Task (Angular)	None	None	None
			10.1.3 ALT: Notification of Validation error Task (Angular)	None	Invalid Notification screen	None
			10.1.3 Confirmation of details Task (Angular)	None	None	None
			10.1.4 Save Line-Item details Task (Angular)	Line-Item	None	None
			10.1.5 Notification of successful save Task (Angular)	None	Success Notification Screen	None
			10.1.5 ALT: Notification of unsuccessful save Task (Angular)	None	Error Screen	None
10.2 Search Line-Item	10.2	Search Line-Item	10.2.1 Capture Line-Item search parameter Task (Angular)	None	Inventory Management screen	None
			10.2.2 Validate search parameter Task (Angular)	None	None	None
			10.2.3 ALT: Notification of Validation error Task (Angular)	None	Invalid Notification screen	None
			10.2.3 Retrieve Line-Item search parameter Task (asp.Net Core)	Line-Item	None	None
			10.2.4 Return Line-Item search results Task (Angular)	None	None	None

			10.2.5 Return Selected Line-Item information Task (Angular)	None	None	None
			10.2.5 ALT: Notification of No line-item found Task (Angular)	None	Error Screen	None
			10.2.5(A) [ALT] Capture line-item Add request Task (Angular)	None	None	None
			10.2.5(B) [ALT] Capture line-item Update request Task (Angular)	None	None	None
10.3 Edit Line-Item	10.3	Edit Line-Item	10.3.1 Display Line-Item details Task (Angular)	Line-Item	Edit line-item screen	None
			10.3.2 Request and capture edited Line-Item info Task (Angular)	None	None	None
			10.3.3 Validate Captured Line-Item details Task (Angular)	None	None	None
			10.3.4 ALT: Notification of Validation error Task (Angular	None	Invalid Notification screen	None
			10.3.4 Confirmation of updated Line-Item info Task (Angular)	None	None	None
			10.3.5 Update Line-Item info Task (asp.Net Core)	Line-Item	None	None
			10.3.6 Notification of successful save Task (Angular)	None	Success Notification Screen	None
			10.3.6 ALT: Notification of unsuccessful save Task (Angular)	None	Error Screen	None

10.1.11 Report Subsystem

11. Reporting Subsystem						
Requirement	Use Case	Use Case	Primitive DFD Processes	Entities	Screens	Report
11.1 Generate Site Progress Report	11.1	Generate Site Progress Report	11.1.1 Display Site Progress Report Screen Task (Angular)	None	Generate Site Progress Report Screen	None
			11.1.2 Capture Report Parameters Task (Angular)	None	None	None
			11.1.3 Validate Report Parameters Task (Angular)	None	None	None
			11.1.4 Retrieve all necessary data from datastores to generate report (Angular .NET Core	• Site Type • Project • Project Status • Company Information • Site Progress description • Construction Photo • Completion Photo	None	None
			ALT 11.1.4 Display Validation Error Task (Angular)	None	Validation Error	None
			11.1.5 Generate Site Progress Report Task (Angular)	None	None	Site Progress Report
			11.1.6 Display	None	Generate Site	None

			Generated Report Task (Angular)		Progress Report Screen	
			11.1.7 Capture Download Request Task (Angular)	None	None	None
			11.1.8 Send Report Task (Angular)	None	None	None
11.2 Generate Site Type Report	11.2	Generate Site Type Report	11.2.1 Display Generate Site Type Report Screen Task (Angular)	None	Generate Site Type Report Screen	None
			11.2.2 Capture Date Range Task (Angular)	None	None	None
			11.2.3 Validate Date Range Task (Angular)	None	None	None
			11.2.4 Retrieve all necessary data from datastores to generate report (Angular. NET Core)	• Site Type • TSS • Project • Project Status • Company Information	None	None
			ALT 11.2.4 Display Validation Error Task (Angular)	None	Validation Error	None
			11.2.5 Generate Site Type Report Task (Angular)	None	None	Site Type Report
			11.2.6 Display Generated	None	Generate Site Type Report Screen	None

			Report Task (Angular)			
			11.2.7 Capture Download Request Task (Angular)	None	None	None
			11.2.8 Send Report Task (Angular)	None	None	None
11.3 Generate Team Site Completed Report	11.3	Generate Team Site Completed Report	11.3.1 Display Generate Team Site Completed Report Screen Task (Angular)	None	Generate Team Site Completed Report Screen	None
			11.3.2 Capture Report Parameters Task (Angular)	None	None	None
			11.3.3 Validate Report Parameters Task (Angular)	None	None	None
			11.3.4 Retrieve all necessary data from datastores to generate report (Angular. NET Core)	• TSS • Site Type • Project • Project Status • Company Information • Team • Employee Team • Employee • Employee Type	None	None
			ALT 11.3.4 Display Validation Error Task (Angular)	None	Validation Error	None

			11.3.5 Generate Team Site Completed Report Task (Angular)	None	None	Team Site Completed Report
			11.3.6 Display generated report Task (Angular)	None	Generate Team Site Completed Report Screen	None
			11.3.7 Capture Download Request Task (Angular)	None	None	None
			11.3.8 Send Report Task (Angular)	None	None	None
11.4 Generate Employee Document Expiry Report	11.4	Generate Employee Document Expiry Report	11.4.1 System Prompted to Generate Employee Document Expiry Report (Angular)	None	None	None
			11.4.2 Retrieve all necessary data from datastores to generate report (Angular. NET Core)	• Employee • Employee Compliance Document	None	None
			11.4.3 Generate Employee Document Expiry Report Task (Angular)	None	None	Employee Document Expiry Report
			11.4.4 Retrieve Human Resource Officer Details Task	• User	None	None

			(Angular .NET Core)			
			11.4.5 Send Report Task (Angular)	None	None	None
11.5 Generate Site Acceptance Document	11.5	Generate Site Acceptance Document	11.5.1 Capture Generate Site Acceptance Document Request Task (Angular)	None	None	None
			11.5.2 Retrieve all necessary data from datastores to generate Site Acceptance Document (Angular. NET Core)	• User • Construction Site • TSS	None	None
			11.5.3 Generate Site Acceptance Document Task (Angular)	None	None	Site Acceptance Document
			11.5.4 Display Site Acceptance Document Screen Task (Angular)	None	Site Acceptance Document Screen	None
			11.5.5 Capture Confirmation Task (Angular)	None	None	None
			ALT 11.5.6 Display Site Acceptance Document Not Saved Alert Task (Angular)	None	Site Acceptance Document Not Saved Alert	None
			11.5.6 Send Site	None		

			Acceptance Document Task (Angular)			

10.1.12 User Management Subsystem

12. User management subsystem						
Requirement	Use Case No.	Use Case Title	Processes from Primitive Data Flow Diagram	Entities from ERD	Screens	Reports
12.1 Login	12.1	Login	12.1.1 Display "login" page task (Angular)	None	Login	None
			12.1.2 Capture and validate information task (Angular)	None	None	None
			ALT 12.1.3 Validation error message task(Angular)	None	Validation Error Dialog Screen	None
			12.1.3 Retrieve email details task(Angular .NET core)	User	None	None
			ALT 12.1.4 Validation error message task (Angular)3	None	Validation Error Dialog Screen	None

			12.1.4 Capture Hash algorithm task (Angular .NET Core)	User	None	None
			ALT 12.1.5 Validation error message task (Angular)2	None	Validation Error Dialog Screen	None
			12.1.5 Retrieve user's role task (Angular .NET Core)	Employee	None	None
			12.1.6 Log in task(Angular)	None	None	None
12.2 Logout	12.2	Logout	12.2.1 Display "Logout" modal task(Angular)	None	Logout modal	None
			12.2.2 Logout task (Angular)	None	None	None
12.3 Reset password	12.3	Reset password	12.3.1 Display "Forgot password" modal (Angular)	None	Forgot password screen	None
			12.3.2 Display "Reset password" modal (Angular)	None	Reset password modal	None

			12.3.3 Display "Request email" screen (Angular)	None	Request email screen	None
			12.3.4 Capture and validate email information (Angular)	None	None	None
			ALT 12.3.5 Email validation error message Angular	None	Email validation error message	None
			12.3.5 Retrieve security question (Angular .NET Core)	User	None	None
			ALT 12.3.6 Retrieve security question error Angular	None	Security question error message	None
			12.3.6 Display "Security question" screen (Angular)	None	Security question screen	None
			12.3.7 Capture and validate security answer information (Angular)	None	None	None
			ALT 12.3.8 Security answer validation error	None	Security answer validation	None

			message Angular		error message	
			12.3.8 Compare and validate security answer (Angular .NET Core)	User	None	None
			ALT 12.3.9 Security answer error message Angular	None	Security answer error message	None
			12.3.10 Capture and validate password (Angular)	None	None	None
			ALT 12.3.11 Password error message Angular	None	Password error message	None
			12.3.11 Display "Confirm reset password" modal (Angular)	None	Confirm reset password modal	None
			12.3.12 Update passwords in database (Angular)	User	None	None
			12.3.13 Display "password reset" modal (Angular)	None	Password reset modal	None

10.1.13 Client Subsystem

13. Client						
Requirement	Use Case No.	Use Case Title	Processes from Primitive Data Flow Diagram	Entities from ERD	Screen	Report
13.1 Add Client	13.1	Add Client	13.1.1 Capture Client details Task (Angular)	None	Create Client screen	None
			13.1.2 Validate Captured details Task (Angular)	None	None	None
			13.1.3 ALT: Notification of Validation error Task (Angular)	None	Invalid Notification screen	None
			13.1.3 Confirmation of details Task (Angular)	None	None	None
			13.1.4 Save Client details Task (Angular)	Client	None	None
			13.1.5 Notification of successful save Task (Angular)	None	Success Notification Screen	None
			13.1.5 ALT: Notification of unsuccessful save Task (Angular)	None	Error Screen	None
13.2 Search Client	13.2	Search Client	13.2.1 Capture Client search parameter Task (Angular)	None	Clients screen	None
			13.2.2 Validate search parameter Task (Angular)	None	None	None
			13.2.3 ALT: Notification of Validation error Task (Angular)	None	Invalid Notification screen	None
			13.2.3 Retrieve Client search parameter Task (asp.Net Core)	Client	None	None
			13.2.4 Return Client search results Task (Angular)	None	None	None
			13.2.5 Return Selected Client	None	None	None

			Information Task (Angular)			
			13.2.5 ALT: Notification of No client found Task (Angular)	None	Error Screen	None
			13.2.5(A) [ALT] Capture client Add request Task (Angular)	None	None	None
			13.2.5(B) [ALT] Capture client Update request Task (Angular)	None	None	None
13.3 Edit Client	13.3	Edit Client	13.3.1 Display Client details Task (Angular)	Client	Edit client screen	None
			13.3.2 Request and capture edited Client info Task (Angular)	None	None	None
			13.3.3 Validate Captured Client Details Task (Angular)	None	None	None
			13.3.4 ALT: Notification of Validation error Task (Angular	None	Invalid Notification screen	None
			13.3.4 Confirmation of updated Client info Task (Angular)	None	None	None
			13.3.5 Update Client info Task (asp.Net Core)	Client	None	None
			13.3.6 Notification of successful save Task (Angular)	None	Success Notification Screen	None
			13.3.6 ALT: Notification of unsuccessful save Task (Angular)	None	Error Screen	None

Conclusion

This section of the validation tables on each functional requirement and the relevant data flows within the functional requirement.

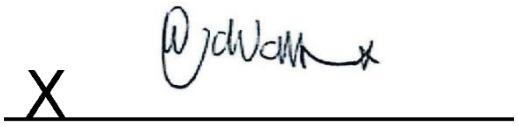
Project Related Documents

11. Sign-off by The Client

Introduction

This section serves as proof that the client has read through the deliverable presented and agrees with the content and work that has been provided

11.1 Client Sign off



X

Michael Van Der Walt
Director

By signing this document, the client verifies that the deliverable 4 and content provided within are acceptable, up to standard and in relates to the business solution proposed from the Project proposal.

Conclusion

This concludes the Client sign off section and the client signature is present indicating that the work is acceptable and relates to the business solution that was proposed.

12. Complexity Matrix

Introduction

The complexity marks document indicates the level of complexity the proposed system for Gohvan construction will have and feature. All the required complexity marks listed must achieve a total score of above 150 points for a five-man team and must be maintained throughout the project lifecycle and finished product.

12.1 Complexity Matrix

Topic	Level		Marks	MAX
1. Special GUI	For online applications: Responsive web design For desktop applications: Form design according to design principles (Schneiderman's golden rule on navigation applies here)	*	3	**42**
	Appropriate use of grids/tables		3	
	Appropriate use of tabs/links		3	
	Use of graphs in an appropriate business context		4	
	The storage and display of graphical information, like photos with a good business reason		3	
	Working e-mail automatically generated from the database in an appropriate business context		2	
	SMS messages automatically generated from the system in an appropriate business context		2	
	Extensive user-friendly search facility		3	
	At least one use of a tree to display data from the database		3	
	Able to dynamically modify a data tree structure and in doing so adjusting the data in the database		4	
	At least one use of a calendar view of data (not a date/time picker; not a plug-in such as Google calendar)		3	
	Uploading a file into the system with appropriate business reason		3	
	The use of audio/video in an appropriate business context		3	
	At least one use of an administrator configurable timer in an appropriate business context		3	
2. Database access	At least 30 tables used (4 member groups) or 40 tables used (5 member groups)	*	6	**15**
	Full referential integrity on all tables	*	6	
	At least one use of master-detail table relationships (Schneiderman's golden rule on system status applies here)	*	3	
3. Reports	At least 3 simple list reports in a reporting tool (no control breaks, no graphs, single table)	*	3	**15**
	At least 2 transactional report with 2 or more control breaks (with heading and calculated values/totals, multiple tables)	*	6	
	At least 1 report with adjustable criteria		3	

	At least 1 management report using a graph		3	
4. Flexibility	All data that can change in future should not be hard coded but maintained in a sub-module of the system (e.g. Lookup tables)		6	**12**
	Some business rules are not hard coded but maintained in a sub-module of the system.		6	
5. Error handling	**All** system-generated errors are trapped, and consistent, user-friendly error messages are displayed		6	**12**
	Appropriate data validation on **all** input fields		6	
6. Help	At least one menu item or other control that opens up a complete help document (HTML, PDF, Help-file)		3	**15**
	Extensive context-sensitive help. E.g. calling Help on a specific screen/function will automatically open the specific help for that screen/function.		6	
	Search Facility on Help		3	
	Extensive use of hints		3	
7. Security	Logon screen with user ID and password and fixed user profiles		3	**13**
	Applying two factor authentication with applicable business reason.		3	
	Encrypted passwords in database	*	1	
	Flexible user profiles (i.e. you can dynamically add user profiles that will enable/disable access to certain parts of the system)		6	
8. Audit Trail	An audit trail of all transactions in the system showing at least date, time, user, transaction type, critical data (such as amount and quantity of transaction)		6	**9**
	Able to search the audit trail on any of the following: date, user, transaction type		3	
9. Deployment	For a desktop application: Fully functional installation disks that take care of application installation requirements (install and uninstall)		3	**15**
	For an online application: Deployment of application to a publicly accessible web server		3	
	For a mobile application: Deployment to an App Market place (such as the PlayStore or the AppStore)		6	
	Deployment of the database to a remote database server		3	
10. Backup and Restore	A backup and restore subsystem exists that backup/restore all data (system may exit during restore)		3	**3**
11. Import/Export Data	Able to open Word or Excel and automatically place data in it based on the parameters provided (with a good business reason)		6	**9**
	XML or JSON: At least 1 XMLor JSON file for Importing or Exporting of data (with good businoutputess reason)		3	

nal INPUT device	Simple Link to an external INPUT device using plugand-play technology, such as a swipe card reader, bar code reader, etc. or a native component such as a QR reader, a GPS component, etc.		3	**18**
	Loose Link to an external INPUT device using device specific software. Data or images must seamlessly be stored in the database, but device specific software is visible to the user. (This could include a digital camera, scanner, voice recording device, thump print reader, etc.)		6	
	Tight Link to an external INPUT device using device specific software. Data or images must seamlessly be stored in the database, but device specific software is **not** visible to the user. (This could include a digital camera, scanner, voice recording device, thump print reader, etc.)		9	
nal APPLICATION / Services	Integrate an existing web service into your application (with good business reason)		3	**9**
	A fully functional link to an installed external application system exists and the interface must be shown to work on the external system. Note that this excludes Microsoft Office Applications		6	
Multiplatform ocessing for an propriate business son	Appropriate business use of static views on an alternative platform.		3	**27**
	Appropriate use of dynamic views on an alternative platform (i.e. data is displayed from the system's database)		3	
	Appropriate use of substantial dynamic views on an alternative platform (i.e. both reading and writing data from the system's database)		9	
	Uploading a file through an alternative platform onto the system's database.		3	
	Substantial processing on a third platform (i.e. both reading and writing data from the system's database)		9	
gramming Principles	The use of a data layer to facilitate interaction between your database and your business layer		3	**12**
	The use of an API to facilitate interaction between your business layer and your presentation layer		6	
	Comprehensive use of stored procedures and/or triggers and/or jobs.		3	
vative addition to the em	Any very advanced innovative addition to the system (e.g. machine learning, AI, block chain, text mining, IOT, etc.)	#	1 - 9	**9**

Maximum Complexity Marks 222

Total Calculated Complexity Marks 171

Conclusion

This concludes the complexity mark section for Team 7 which has been filled out as to the sections we can complete with this current system design.

13. Sign-off by The Team

Introduction

This section includes the sign off by all team members indicating that this is their own work.

11.1 Team Sign off

This section includes the sign off by all team members indicating that this is their own work

X 

Matthew Veltman
u19050608

X 

Michael Vosloo
u19031174

X 

Tshepang Mashao
u19293675

X 

Vincent Yu
u15195059

X 

Amore Rossouw
u19052864

Conclusion

By signing this document, the Team Paralinear Solutions acknowledges the document as a whole and signs off that all work presented is their own.

Deliverable Conclusion

Paralinear solutions has created this document containing, a use case diagram, use case narratives, Data flow diagrams, entity relationship model, input descriptions, output descriptions and validation and hands it over as the final submission for the INF 370 deliverable 4 submission.