INF2001
Introduction to Software Engineering

SINGAPORE INSTITUTE OF TECHNOLOGY

# Milestone 2: Software Design Specifications

**for**

# Workload Management System for Ferry Company

**Prepared by**

**Team's Name:** ***Team 4***


| Team Members | Student ID |
|---|---|
| ASHLINDER KAUR DO S. AJAIB SINGH | 2202636 |
| LEO EN QI VALERIE | 2202795 |
| OSAMA RASHEED KHAN | 2203385 |
| SERI HANZALAH BTE HANIFFAH | 2201601 |
| TEO XUANTING | 2202217 |
| TIAN YUE XIAO, BRYON | 2201615 |


**Lab Group:** P3 - 4
**Github Handle:** INF2001-P3-T4
**Date:** 28 October 2023

## Table of Contents

# 1 Introduction

Due to globalisation in today's ever-evolving digital landscape, this project aims to develop a web-based workload management system (WMS) tailored for ferry companies. The primary objective of this system is to establish a user-friendly platform that enhances the employee experience and significantly boosts operational efficiency. The four main positive impacts the project strives to achieve are: enhanced operational efficiency, improved work-life balance, transparency, and staff empowerment.

## 1.1 Product Scope Changes

No changes are implemented to the product scope.

# 2 Changes in Requirements

There have been revisions made to certain requirements from the M1 cycle. These changes encompass various aspects, including user interface requirements, use case model, performance requirements, and software estimations. These updates are addressed below in this section, ensuring clarity and transparency in the project development.

## 2.1 User Interface Requirements

| USER INTERFACE | EXPLANATION |
|---|---|
| The web application system shall be aesthetically pleasing. | The design of the web application system should adhere to a defined colour scheme and visual style in order to get user satisfaction.<br>It must provide a visually appealing and engaging environment to enhance the overall usability of the system. |
| The web application system shall be easy to read, intuitive and easy to understand. | The web application system should have clear and intuitive user interfaces. This allows users to navigate the system easily which will result in a clear user experience, allowing users to access and utilise all the functions and features of the web application system. |
| The web application system shall be interactive with responsive elements. | The web application system must have real time updates and interactive components, such as dropdown menus, buttons and forms to provide a seamless and responsive user interaction. Immediate feedback and response to user actions are important to enhance user satisfaction and encourage active engagement with the system. |

## 2.2 Non-Functional Requirements

### 2.2.1 Performance Requirements

**P1:** The system shall respond to user interactions within 2 seconds.

**P2:** The system shall have an average load time for pages and data of less than 3 seconds.

**P3:** The system shall be able to handle concurrent requests from at least 150 simultaneous users during peak periods.

**P4:** The system shall be able to process and update the job assignments and availability in real-time, ensuring managers and staff have immediate access to the information.

### 2.2.2 Safety and Security Requirements

**S1:** The system must ensure that all transmitted and received data is encrypted using AES-128.

**S2:** The system must mandate that all approving roles employ an authentication factor distinct from the one used by regular users (in addition to the existing 2FA) during approval processes.

**S3:** The system must restrict access to sensitive data and critical system functions to authorised users only in order to prevent misuse or unauthorised access.

**S4:** The system must maintain detailed logs of user activities, including login attempts, changes to job assignments, and any modifications to availability preferences. These logs will be used for auditing and evaluating security incidents.

**S5:** The system must implement robust disaster recovery and backup mechanisms to ensure the availability and integrity of data in case of system failures.

## 2.3 Revised Use Case Diagram

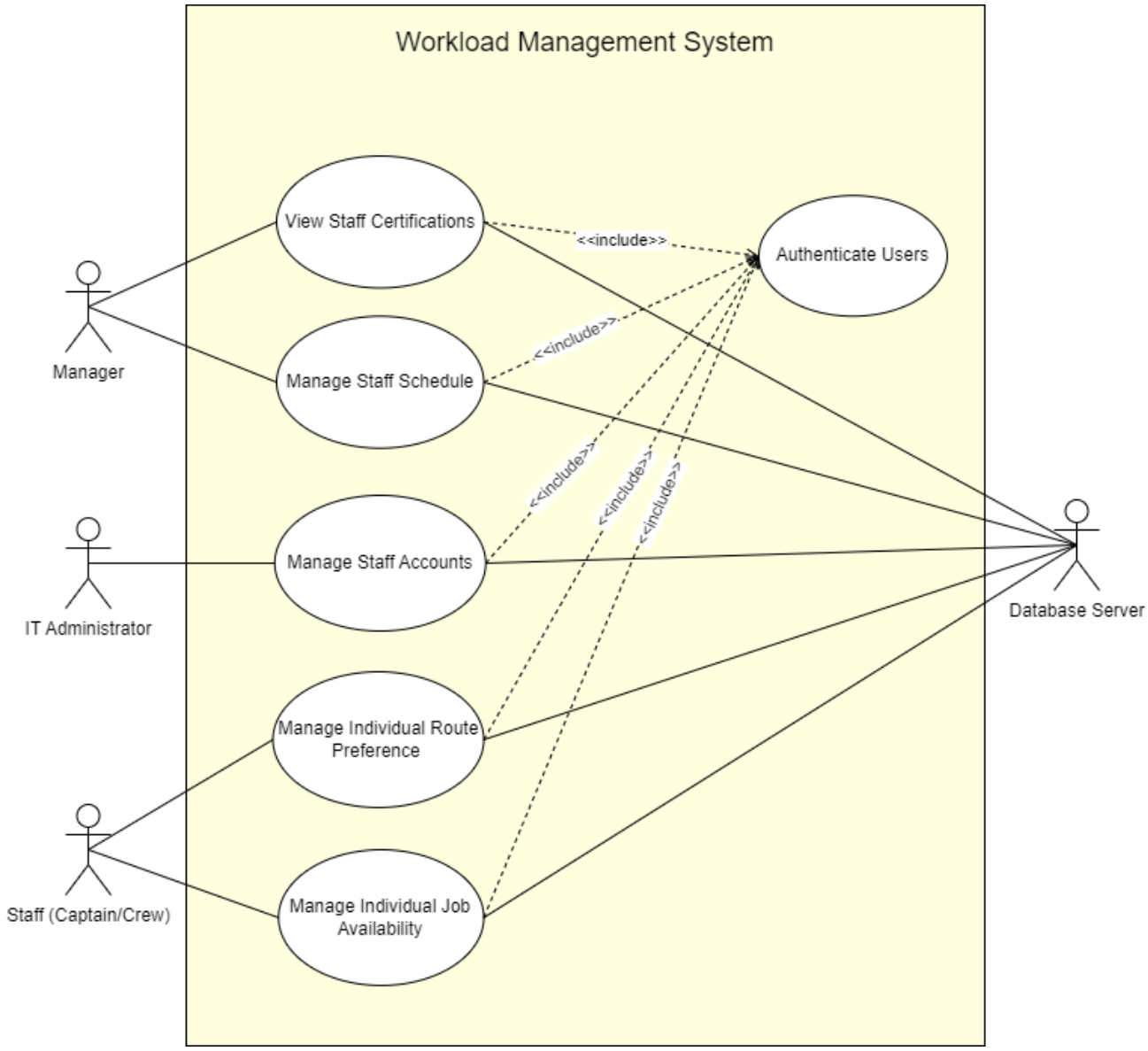


*Figure 1.0: Revised Use Case Diagram*

The revised use case diagram depicts the interactions between the WMS and the 3 primary actors — the Manager, the IT Administrator and the Staff (Captain/Crew) — as well as 1 secondary actor, the Database Server.

For clarity, please refer to Appendix A for the descriptions of each of the use cases.

# 3 Software Design

The Software Design shows the architecture and the relationship between the Ferry company's workload allocation and the Management System. The following Class diagram, Component diagram and Sequence diagram depict the detailed design of the system's architecture which shall be used to construct the Management System.

## 3.1 Class Diagram

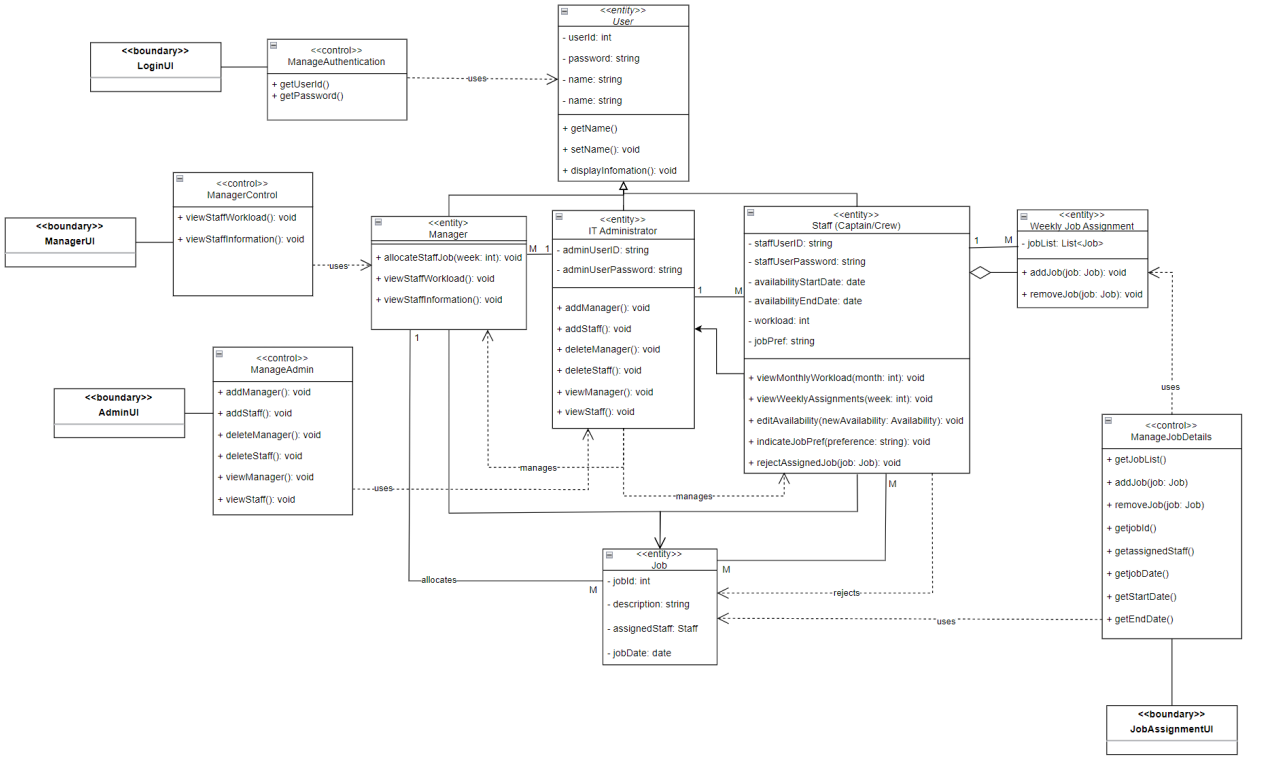


*Figure 2.0: Class Diagram of WMS*

**Entity Classes**

| CLASS | DESCRIPTION |
|---|---|
| User | User acts as a parent class that stores common data such as ID, password and name. |
| Manager | Manager inherits from the user class and performs functions to allocate staff jobs as well as view staff workload and information. |
| IT Administrator | IT Administrator inherits from the user class and has functions to manage both manager and staff such as addition or deletion. |
| Staff (Captain/Crew) | Staff inherits from the user class and contains more staff specific |

| | |
|---|---|
| | attributes and functions related to their availability, workload and job preference. |
| WeeklyJobAssignment | Weekly Job Assignment is an aggregation from Staff that includes job specific functions to add a job or to delete a job. |
| Job | Job is an entity that is allocated by a manager and has a many to many association with staff. Also, the job entity has a dependency as staff can reject a job. |

**Control Classes**

| CLASS | DESCRIPTION |
|---|---|
| ManageAuthentication | It is responsible for user authentication where the class interacts with the User entity class to verify the user credentials. |
| ManagerControl | It is responsible for overseeing the access to the staff workload and information through the Manager entity class, enabling managers to view this information. |
| ManageAdmin | It is responsible for overseeing the administration process and allowing the IT administrator to manage the managers and staff through the interaction with the IT Administrator entity class. |
| ManageJobDetails | It is responsible for overseeing the job-related functions and managing job details and assignments through the interactions with the Job entity and Weekly Job Assignment entity classes. |

**Boundary Classes**

| CLASS | DESCRIPTION |
|---|---|
| LoginUI | It serves as a user interface element which is responsible for presenting the login screen to users and to capture their login credentials. |
| ManagerUI | It creates a boundary between the ManagerControl class and the Manager and IT Administrator entities. It provides a user interface to view staff workload and information. The ManagerUI communicates with the ManagerControl class to perform the requested operations, allowing managers to view staff workload and staff information. |
| AdminUI | It serves as a user interface element to facilitate the administrative tasks and allows the admin to interact with the system for user management. AdminUI communicates with ManageAdmin control class, allowing them to add, delete and view managers and staff. |
| JobAssignmentUI | It serves as an interface for staff to view their assigned job allocations. |

## Software Design Principles

The software is carefully designed in such a way with considerations to several design principles fundamentals.

| DESIGN PRINCIPLE | REASONING |
|---|---|
| **Modularisation** | The software design separates functionality into distinct classes – User, Manager, IT Administrator and Staff classes. These classes are each responsible for specific aspects of the system. This modularisation enhances maintainability and allows for easier updates or changes to individual components. |
| **Abstraction** | Abstraction is used in inheritance. The User serves as a parent class, abstracting common attributes like ID, password and name, which are inherited by more specific user types like Managers and Staff. Hence, this hierarchy allows for a high level of abstraction. |
| **Encapsulation** | Encapsulation is maintained through the use of classes with well-defined attributes and methods. Every class encapsulates its data and operations. This ensures that data integrity is preserved and only specific methods provide controlled access to data. |
| **Coupling** | The design shows loose coupling by separating responsibilities into different classes.<br>E.g. ManagerUI communicates with ManagerControl to perform operations, reducing the interdependence between components. This promotes flexibility and ease of maintenance. |
| **Cohesion** | Each class shows high cohesion which focuses on specific responsibilities.<br>E.g. Manager has methods related to managerial tasks. This ensures that functions within a class are closely related, making the code more understandable and maintainable. |
| **Separation of Interface & Implementation** | The boundary classes – LoginUI, ManagerUI, AdminUI and JobAssignmentUI, separate the interface (user interaction) from the implementation (control classes and entities). This separation allows for user-friendly interfaces to be modified independently of the underlying logic. |
| **Sufficiency** | The classes provide sufficient functionality for the identified roles and responsibilities in the system.<br>E.g. ManageAdmin encapsulates IT administrator functions and interaction with the IT Administrator entity class, ensuring that administrative tasks are adequately covered. |
| **Completeness** | The design is complete as it addresses the specified requirements. Classes (ManagerControl and ManageJobDetails) ensure that the system has the necessary components to oversee and manage staff workload, information and job-related functions. |

## 3.2 Component Diagram

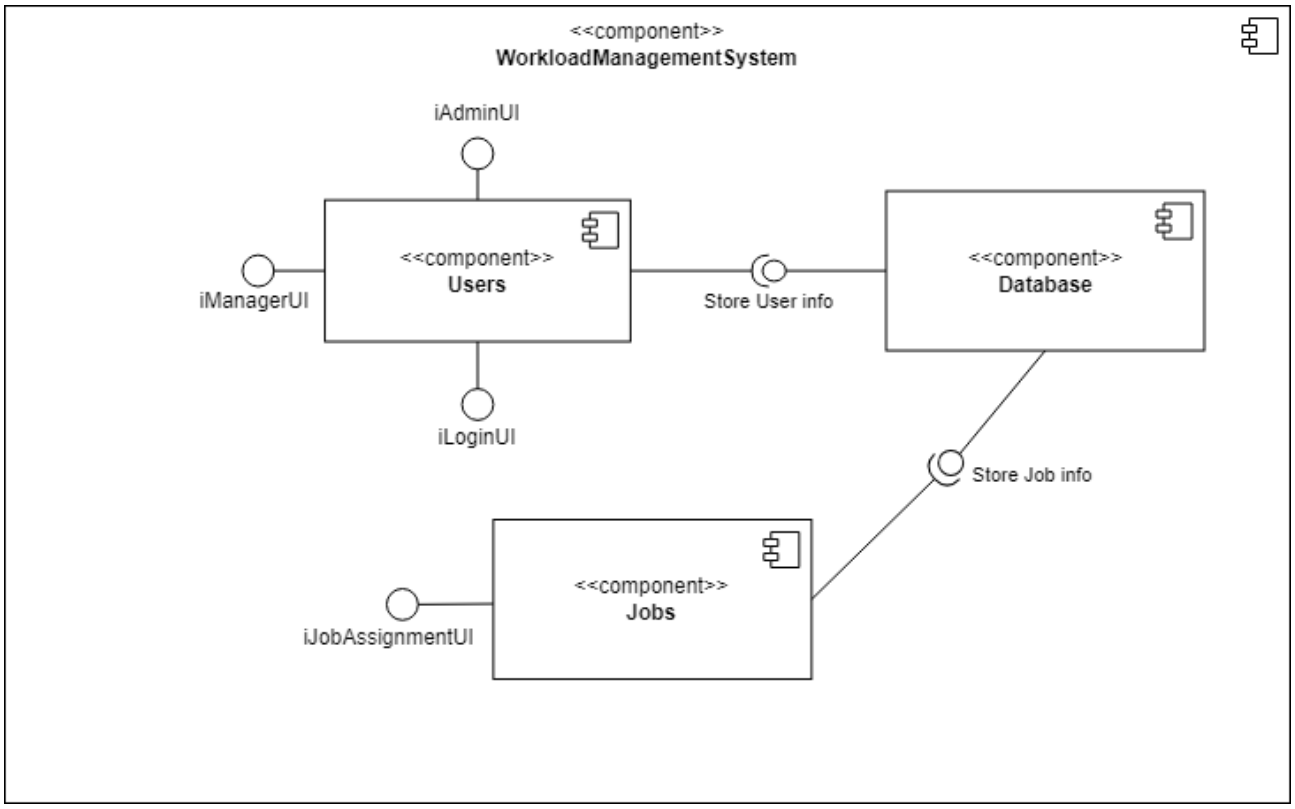


*Figure 3.0: An Overview of the Component Diagram of the WMS*

## User Component

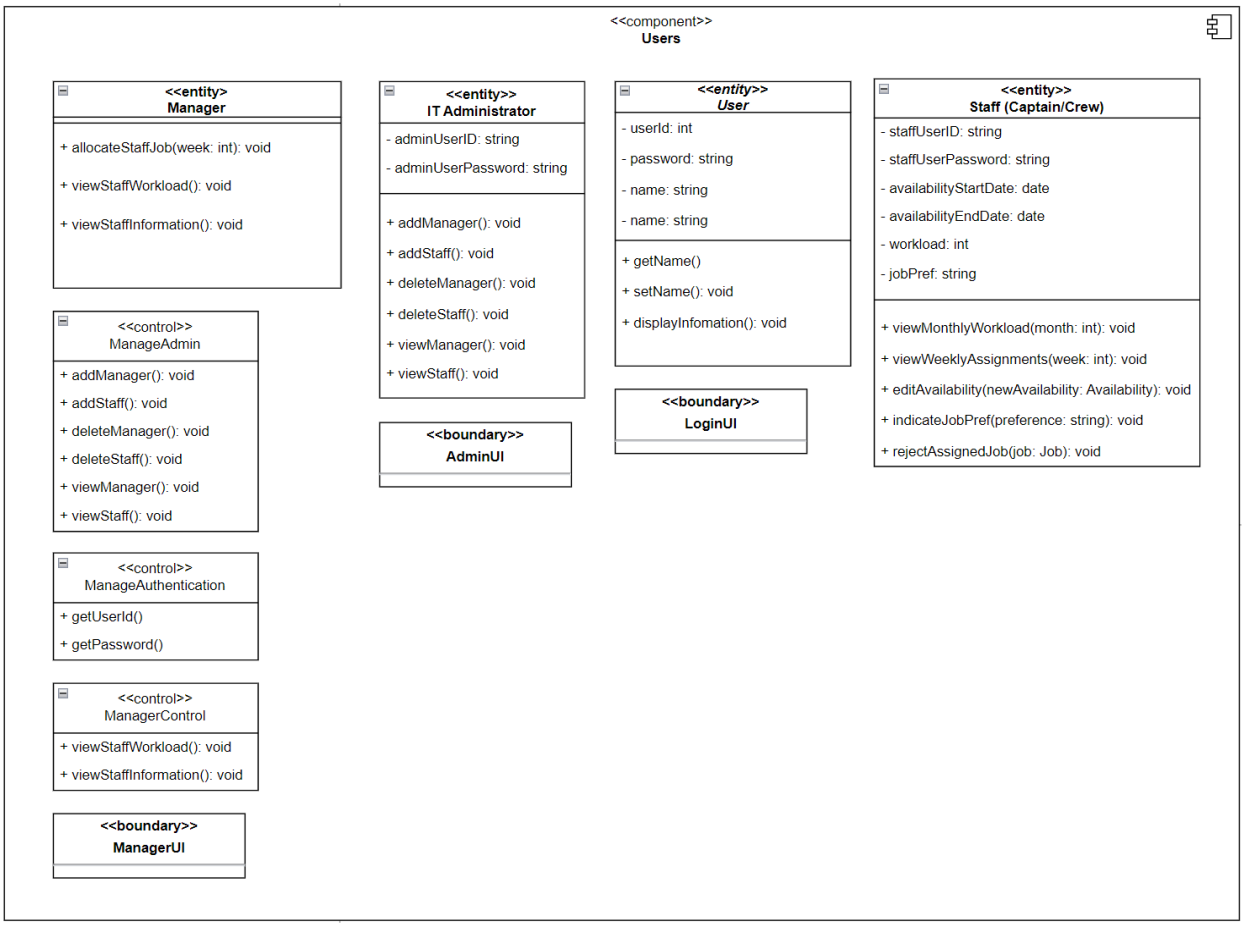


*Figure 3.1: Detailed User Component*

## Jobs Component

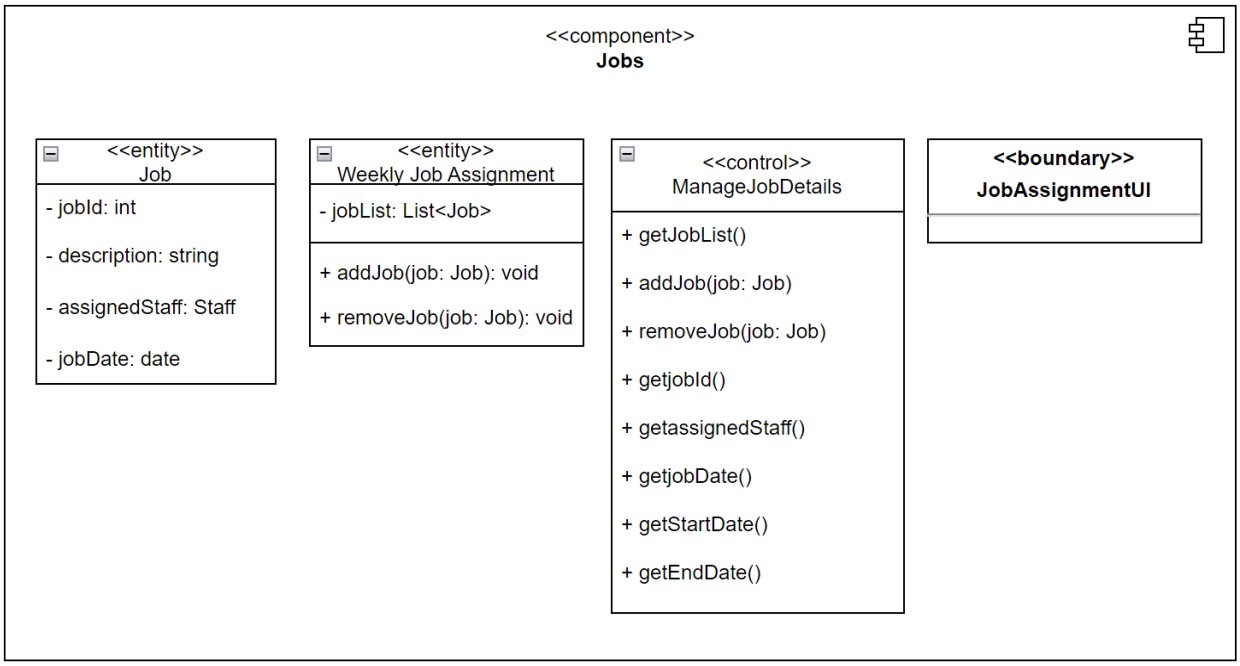


*Figure 3.2: Detailed Jobs Component*

## 3.3 Sequence Diagram

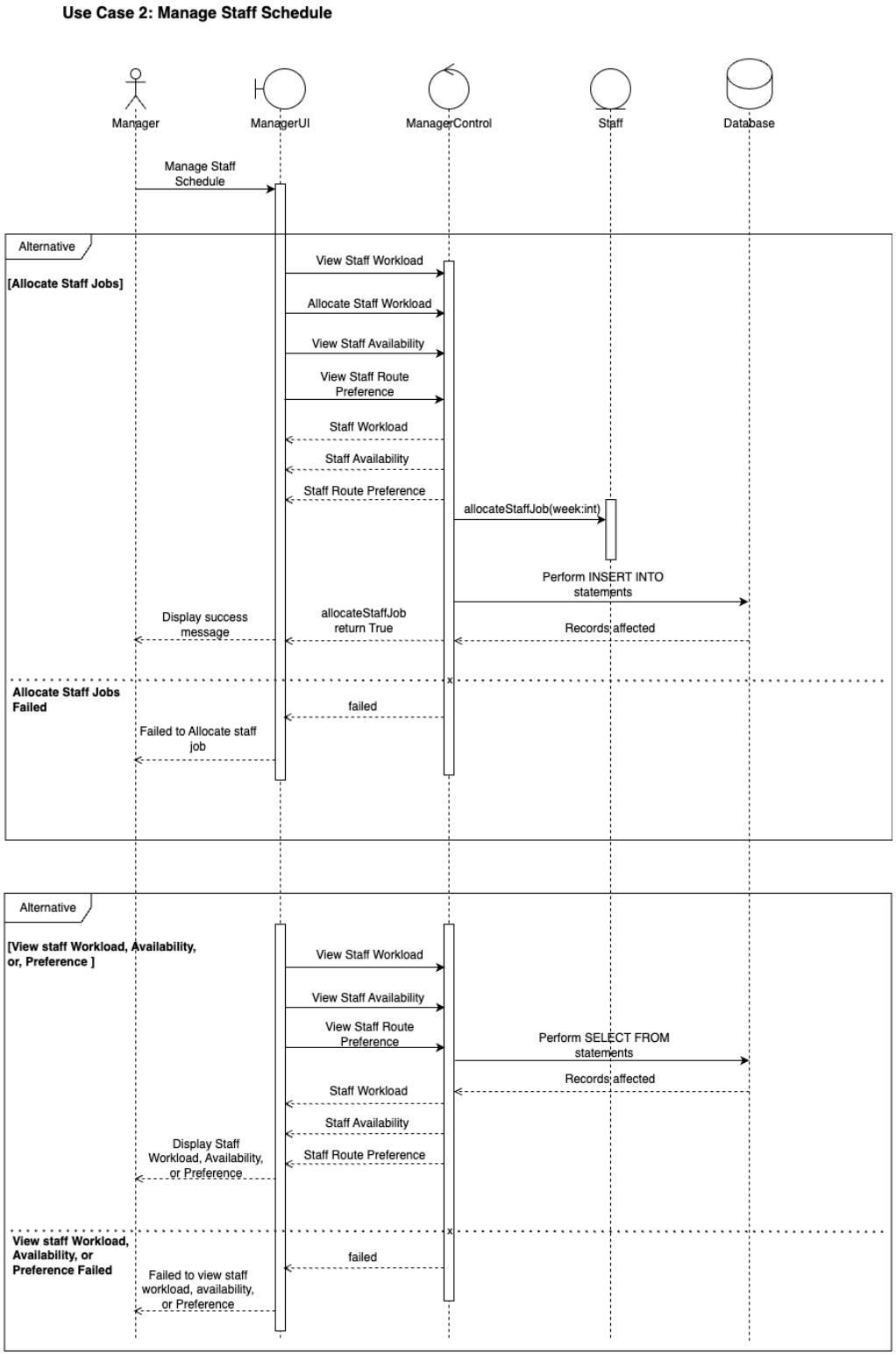


*Figure 4.0: Sequence Diagram on Use Case 2 - Manage Staff Schedule*

**Sequence Diagram on Use Case 2 - Manage Staff Schedule**

Using **Use Case 2: Manage Staff Schedule** for Sequence Diagram 1, there are a total of 4 scenarios. The 4 scenarios include: View Staff Workload, Allocate Staff Workload, View Staff Availability and View Staff Route Preference. In Figure 4.0, the Allocate Staff Workload function is separated from the View Staff Workload, View Staff Availability and View Staff Route Preference functions.

The sequence for the Allocate Staff Workload starts from the actor (Manager) to the manager's user interface (ManagerUI) and passes through the control class (ManagerControl) to the entity class (Staff) with the allocateStaffJob function. The control class (ManagerControl) performs insert into statements to the database. Furthermore, if the allocateStaffJob function returns true, a success message is displayed to the actor (Manager). Alternatively, if the allocateStaffJob function fails to execute, the system fails to allocate jobs to the staff.

For the viewing of functions, the sequence from the user interface (ManagerUI) to the control class (Manager Control) includes: View Staff Workload, View Staff Availability and View Staff Route Preference, where the control class (ManagerControl) performs select statements from the database. As a result, the Staff Workload, Availability or Preference is displayed to the actor (Manager). Alternatively, if the function fails to execute, the staff information will not be displayed to the actor (Manager).

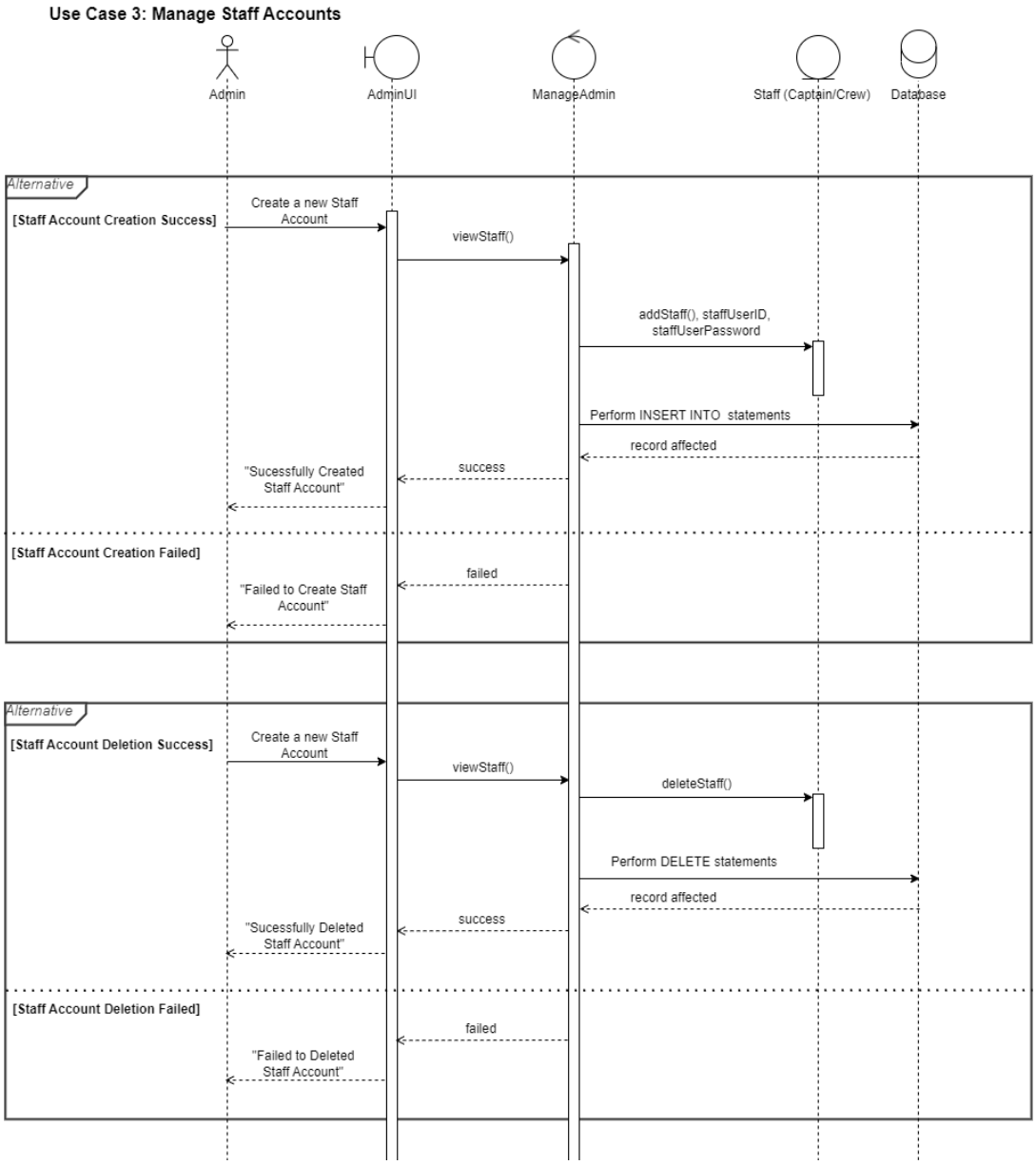


*Figure 5.0: Sequence Diagram on Use Case 3 - Manage Staff Accounts*

**Sequence Diagram on Use Case 3 - Manage Staff Accounts**

Using **Use Case 3: Manage Staff Accounts** for Sequence Diagram 2, there are a total of 4 scenarios with 2 main and 2 alternatives. The first 2 scenarios show the process for Staff Account Creation and Deletion where both contain a success and a failure scenario. The actor (Admin) will send a request through the boundary class (AdminUI) and the control class (ManageAdmin) will use the viewStaff() function to see if the user exists. If the user does not exist, the control class (ManageAdmin) will create a new staff (captain/crew) and update the database. If the user exists, the system will not be able to create the staff account.

### *Sequence Diagram on Use Case 3 Continued:*

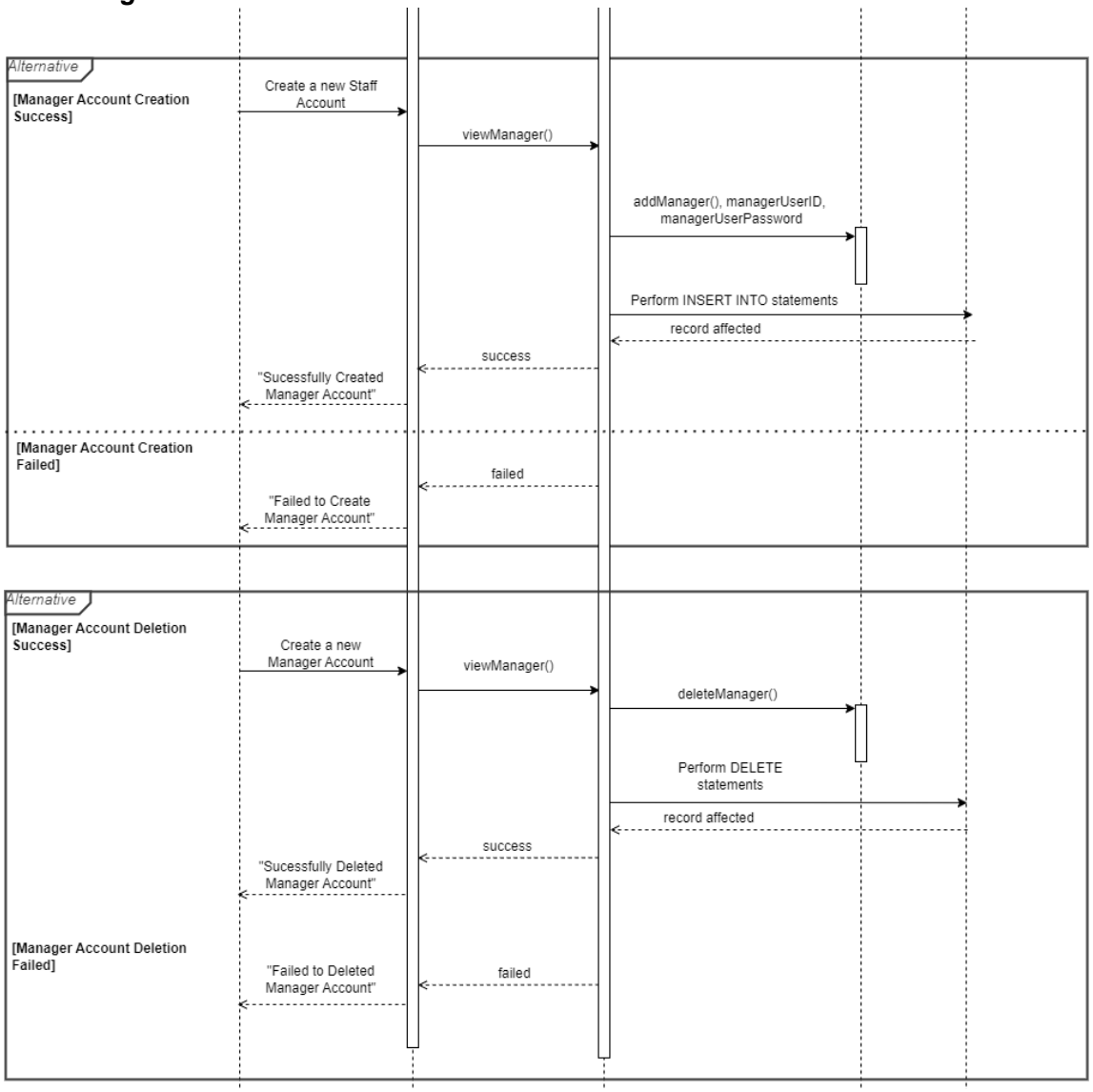


*Figure 5.1: Sequence Diagram on Use Case 3 - Manage Staff Accounts*

The next 2 scenarios show the process for Manager Account Creation and Deletion where both contain a success scenario and a failure scenario. Similarly, the actor (Admin) will send a request through the boundary class (AdminUI) and the control class (ManageAdmin) will use the viewManager() function to view if the user exists. If the user does not exist, the control class (ManageAdmin) will create a new manager and update the database. If the user exists, the system will not be able to create the manager account.

# 4 Updated Project Plan

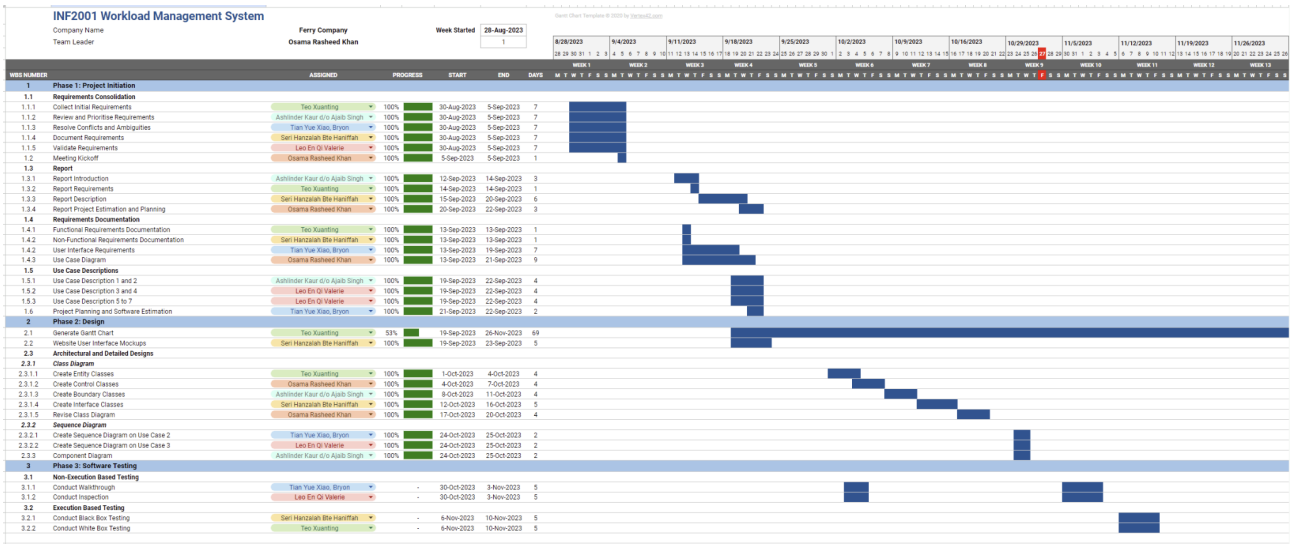


*Figure 6.0: Updated Gantt Chart*

The Gantt Chart's timeline has been updated to include weeks 10 to 13 to handle any changes of dates to the tasks. Each task is also broken down into smaller tasks to allow better resource allocation.

# 5 Appendix A – Updated Use Case Descriptions

| | |
|---|---|
| Use Case ID: | UC1 |
| Use Case Name: | **View Staff Certifications** |
| Description: | This use case describes the process of the manager viewing the staff certifications. The manager can view certificates and inform staff to upload new certifications when existing certificates have expired. |
| Primary Actor: | Manager |
| Secondary Actor: | Database Server |
| Include Use Case(s): | Authenticate Users |
| Priority: | High |
| Preconditions: | Manager must be logged into the system. |
| Postconditions: | Manager shall be able to view staff certifications. |
| Main Success Scenarios: | 1. Manager navigates to the Staff Certifications page.<br>2. System assigns the Database Server to fetch the information on certifications, displays the Staff Certifications page and shows the retrieved information of the staff and their respective certifications with expiry date.<br>3. Manager to ensure all certifications are valid. |
| Alternative Scenarios: | 3a. Certifications are not valid.<br>3a1. Manager to inform staff to renew certifications and have the staff to update the new certifications into the system. |

Steps 1 and 2 of UC1: View Staff Certifications will be kept in the Main Success Scenario as they describe the flow of actions that the Manager will take when viewing the staff certifications. These steps also aligned with the description of “This use case describes the process of the manager viewing the staff certifications.”.

| | |
|---|---|
| Use Case ID: | UC2 |
| Use Case Name: | **Manage Staff Schedule** |
| Description: | This use case allows manager to manage staff schedule with the following functionalities:<br>1. Allocate Staff Workload<br>2. View Staff Availability<br>a. Staff Availability<br>b. Staff Route Preference<br>c. Staff Workload |
| Primary Actor: | Manager |
| Secondary Actor: | Database Server |
| Include Use Case(s): | Authenticate Users |
| Priority: | High |
| Preconditions: | 1. Manager must have successfully logged into his/her account.<br>2. System displays the Job Allocation page to the manager. |
| Postconditions: | 1. Manager shall be able to view staff workload, allocate staff workload and view staff availability. |
| Main Success Scenarios: | **Allocate Staff Workload**<br>1. Manager selects up to 3 staff from a dropdown list to view their job preference and availability.<br>2. Manager to assign the staff their workload for the week by selecting the specific days and saving into the system.<br>3. System assigns the Database Server to update the workload distribution and reflect the assigned workload for each staff.<br><br>**View Staff Availability**<br>1. Manager selects up to 3 staff from a dropdown list to view their job availability.<br>2. System assigns the Database Server to fetch and display the Staff Availability, Staff Route Preference and Staff Workload. |
| Alternative Scenarios: | **Allocate Staff Workload**<br>1a. Manager is unable to view Staff workload.<br>1a1. System displays an empty view as no workload may be assigned yet.<br><br>**View Staff Availability**<br>1a. System does not display any page.<br>1a1. Manager shall try to refresh the page. |

Step 2, “System displays the Job Allocation page to the manager.” has been relocated from the Main Success Scenario to Precondition as this step is a common prerequisite for both scenarios of ‘Allocate Staff Workload’ and ‘View Staff Availability’.

| | |
|---|---|
| Use Case ID: | UC3 |
| Use Case Name: | **Manage Staff Accounts** |
| Description: | This use case allows the IT Administrator to create or remove managers and staff from the system. |
| Primary Actor: | IT Administrator |
| Secondary Actor: | Database Server |
| Include Use Case(s): | Authenticate Users |
| Priority: | High |
| Preconditions: | 1. IT Administrator account must be logged into his/her account.<br>2. System displays the administrator panel and displays the list of managers/staff to the IT Administrator. |
| Postconditions: | 1. New Manager/Staff will be able to access the system.<br>2. Removed Manager/Staff will no longer be able to access the system. |
| Main Success Scenarios: | **Create Managers and Staff**<br>1. IT Administrator selects the “Create New Account” button.<br>2. System displays a form for entering full name, username, password, email, mobile number, and role designation of the new manager/staff’s account.<br>3. IT Administrator fills in the forms and clicks on the “Submit” button.<br>4. System assigns the Database Server to create the new account, displays a list of managers/staff with a successful account creation message, and sends a welcome email containing login username and password to the new manager’s/staff’s email.<br><br>**Remove Managers and Staff**<br>1. IT Administrator selects the manager/staff he/she wishes to remove with the “Remove Account” option.<br>2. System prompts delete confirmation message.<br>3. IT Administrator selects the delete confirmation button.<br>4. System assigns the Database Server to remove manager/staff from the system, and it displays a list of managers/staff with a successful deletion message. |
| Alternative Scenarios: | **Create Managers and Staff**<br>4a. IT Administrator unable to create account due to incomplete form.<br>4a1. System prompts the IT Administrator to fill all information and resubmits the form.<br>4b. IT Administrator cancels the operation for form filling of account creation.<br>4b1. System closes the form and no manager/staff account is created. |

| | |
|---|---|
| | **Remove Managers and Staff**<br>3a. IT Administrator cancels the operation to delete manager/staff account from the system.<br>3a1. System closes the message, and no manager/account is to be deleted. |

The initial step “System displays the list of managers/staff to the IT Administrator.” which was originally part of the Main Success Scenario has been relocated to the Precondition and renamed to “System displays the administrator panel and displays the list of managers/staff to the IT Administrator.”, as this step is common for adding and removing managers and/or staff.

| | |
|---|---|
| Use Case ID: | UC4 |
| Use Case Name: | **Manage Individual Route Preference** |
| Description: | This use case allows staff to indicate route preference. |
| Primary Actor: | Staff (Captain/Crew) |
| Secondary Actor: | Database Server |
| Include Use Case(s): | Authenticate Users |
| Priority: | High |
| Preconditions: | 1. Staff must be successfully logged into his/her account.<br>2. Staff is on the Job Allocation page to choose the route preference. |
| Postconditions: | 1. Staff is able to submit the chosen route preference.<br>2. Staff is able to view the allocated routes and individual preference.<br>3. Staff can accept or reject the allocated routes. |
| Main Success Scenarios: | **Add or edit individual route preference**<br>1. Staff indicates route preference for the week and submits into the system.<br>2. System assigns the Database Server to save the route selection and prompts the success message to staff.<br><br>**View individual allocated and preferred route**<br>1. System assigns the Database Server to retrieve the information on the route allocated by the manager and displays to the staff.<br><br>**Accept or Reject allocated route**<br>1. System shows the staff their respective allocated route scheduled by the manager.<br>2. Staff chooses to either select the accept or reject schedule button.<br>3. System assigns the Database Server to process the staff’s entry and displays their scheduled route. |
| Alternative Scenarios: | **Add or edit individual route preference**<br>2a. Staff chooses route preference for the past week.<br>2a1. System displays a message indicating that the route preference |

| | |
|---|---|
| | can only be chosen from the current week.<br>2a2. Staff will be redirected to the current week for selection.<br>2b. Staff selects an invalid route preference or input.<br>2b1. System displays an error message.<br>2b2. Staff will be prompted to select a valid route preference from the given options.<br><br>**<u>View individual allocated and preferred route</u>**<br>1a. System displays a blank page.<br>1a1. Staff to refresh the page.<br><br>**<u>Accept or Reject allocated route</u>**<br>2a. System experiencing a delay when assigning the Database Server to save the staff's entry, when the staff selects either the 'accept schedule' or 'reject schedule' button.<br>2a1. System will display a message saying that the acceptance or rejection process is still in pending and under processing<br>2a2. Staff is informed that they will get their confirmation of their acceptance / rejection once the processing is complete. |

The initial step "Staff is on the web page to choose the route preference." in the Precondition has been renamed to "Staff is on the Job Allocation page to choose the route preference." for clarity.

The Main Success Scenario is refined to remove a redundant step that has been answered in step 2 of the Precondition.

The Alternative Scenario for 'Accept or Reject allocated route' is expanded to include the scenario where the system faces a delay in entry processing and the steps to resolve this scenario.

| | |
|---|---|
| Use Case ID: | UC5 |
| Use Case Name: | **Manage Individual Job Availability** |
| Description: | This use case allows staff to add or edit work availability for the month. |
| Primary Actor: | Staff (Captain/Crew) |
| Secondary Actor: | Database Server |
| Include Use Case(s): | Authenticate Users |
| Priority: | High |
| Preconditions: | 1. Staff must be logged into their accounts.<br>2. Staff is on the Job Allocation page to manage their availability. |
| Postconditions: | Staff is able to view their indicated availability. If the staff is available on a particular day, the colour will be green and if the staff is unavailable, the colour will be red. |
| Main Success Scenarios: | 1. Staff add or edit their work availability and enter into the system.<br>2. System assigns the Database Server to update their availability and displays the updated work availability to both the staff and manager. |
| Alternative Scenarios: | 1a. System shows a blank screen.<br>1a1. Staff shall refresh the page. |

The step “System displays webpage of the staff work availability.” that was from the Main Success Scenario has been relocated to the Precondition. It is also renamed to “Staff is on the Job Allocation page to manage their availability.” to emphasise that the staff can only manage their job availability on the Job Allocation page.

| | |
|---|---|
| Use Case ID: | UC6 |
| Use Case Name: | **Authenticate Users** |
| Description: | This use case describes the process of the system users logging in to the system. |
| Primary Actor: | Manager, IT Administrator, Staff (Captain/Crew) |
| Secondary Actor: | Database Server |
| Priority: | High |
| Preconditions: | Staff attempts to log into their accounts. |
| Postconditions: | 1. Staff are authenticated and logged into the system.<br>2. Staff will be directed to the landing page. |
| Main Success Scenarios: | 1. User (Manager, IT Administrator, Staff) enters their username and password to log into the system.<br>2. System assigns the Database Server to check if the username and password information exists inside the database server. If the information exists and is correct, the system prompts successful login and directs the user to the landing page. |
| Alternative Scenarios: | 3a. System prompts log in failure message<br>3a1. System prompts the user to enter the correct username and password. |

No changes have been made to this use case.