ICT2101/2201
Introduction to Software Engineering

Milestone 1: Software Development Plan & Specification

For

Ambulance Services

Prepared by

Team's Name: ***Weewoooweewoooo***

Tennyson Koh	**2100816**
Anna Ng	**2101528**
Tan Ai Xin	**2102468**
Tan Chun Guan	**2102892**

Lab Group: p*3-3*

Github handle: *https://github.com/waltertan98/ICT2101-P3-Team03-Project*

Date: *25/9/2022*

Table of Contents

Introduction 3
- Product Scope 3
- Related Background Literature 3
- Intended Audience and Document Overview 4
- References and Acknowledgments 5

Overall Description 6
- Product Overview 6
- Product Functionality 8
- Assumptions and Dependencies 9

Specific Requirements 10
- User Interface Requirements 10
- Functional Requirements 16
- Use Case Model 17
- Non-functional Requirements 18
  - Performance Requirements 18
  - Safety and Security Requirements 18

Project Estimation and Plan 19
- Software Estimation 19
- Project Management 23

Individual Members Task Reflections 25

Appendix A – Use Case Descriptions 27

Appendix B – Data Dictionary 36

1 Introduction

1.1 Product Scope

The project scope is to have an efficient work scheduling system developed for the ambulance team that helps the manager to manage and plan out the weekly work schedule of the staff easily and quickly. Also, to facilitate the decision-making process of the managers when it comes to job scheduling of the staff.

By adopting this system, it would immediately allow the manager to see the staff's workload which can help the manager to plan the weekly job schedule with reference to it. On the view all staff workload page, it would show the manager straight away the total working hours of each staff member with the least working hours on the top. By doing so, it allows the manager to visualize who are the ones that have not exceeded the hours and whom they can assign them. At the same time, on the allocated job page, the staff are able to indicate when they are able to work for the week. Thus, with the feature, the manager could cross reference it and view the workload of the staff page to assist them with planning and faster decision making which would definitely reduce time wastage.

1.2 Related Background Literature

From the source [1][2], we can understand the importance of having a good, well-managed and efficient job scheduling system. Because if the job scheduler did not do a good job in planning, it would lead to the chances of creating chaos and inefficiency in the operations. Such as lacking manpower during operations thus, needing to spend extra time to find backup to cover the job of those that are unable to work or when the scheduler did not allocate enough manpower for the day or week. Therefore, being able to plan efficiently and be prepared for unexpected circumstances would be beneficial as it could help reduce unnecessary chaos and affect the efficiency of the operations.

Also, from this source [2][3], we have learned that time wastage is a serious problem when there is a poor scheduling system. As not only does the scheduler need to waste time remaking or remodifying the schedule again, it would also be a hassle for the scheduler to look through the list of the available employees and assign or call them to work. On top of that, the work productivity of the scheduler would be affected as they would pause their job on hand and prioritize finding substitutes to ensure that there is sufficient manpower for the day. Therefore, during planning, it would be good that the managers should also plan out the schedule of a list of backups for each day so that it would allow the employees to be prepared if they were to be called back and they would not take it as a surprise.

According to an article [4], encryption of information is important and should be adopted by healthcare providers to prevent any data breaches or information from being stolen [5]. Thus, a certain industry standard should be maintained to avoid such situations to occur. As statistics have proven that the healthcare industry is the most vulnerable to data breaches and there have also been

cases where data breaches did occur in the Singapore healthcare sector [6]. Therefore, after knowing the importance of maintaining a good industry standard, we would be adopting Advanced Encryption Standard (AES) in our system to increase the safety and security of any data that is being transferred around or inputted into our system.

Thus, in order to tackle such issues, a good job scheduling system could be a solution to these problems. As with the system, features could be implemented to best suit the problems the schedulers are facing. In the long run, benefiting them by improving the efficiency of planning and reducing the time required to plan the schedule. On top of that, a good security standard or practice should be adopted to protect the staff or any information that is passing through the system.

1.3 Intended Audience and Document Overview

The purpose of this document is to propose the possible features and functions that we have in mind that we believe would be able to help solve the problems that our clients are facing when it comes to job scheduling for the ambulance crew.

This document is organized in a way that it is sectioned into 4 individual parts. The first part (Introduction), it would be focusing on summarizing and identifying the project scope, goals and objectives that we wish to help our client solve. Also, research on the possible problems faced and possible solutions for them.

The second part (Overall Description), it would be focusing on the overview of the entire project, it would be focusing on the overview of the entire project where our system architecture diagram would illustrate the different layers which include the presentation, business logic, and data layer. Each of the layers would show the connection between the users, the system, and other external interfaces. Also, summarizing the major functionalities that we have would help counter the problems that our client is facing.

The third part (Specific Requirements), it would be going in-depth into the functions that we have that touches on the purpose of each function and how each function works. Also includes a use case model to map out the system flow that would help readers to understand how each function is linked. In this section, we would also be including all the functional and non-functional requirements that we have identified through the project description.

The fourth part (Project Estimation and Plan), it would focus on coming out with a rough estimation of the cost and time taken for the whole project. Use cases have been created based on the expected system flow to help with the calculation of the use case points, unadjusted weight of use case and actors, TCF, and UCP to get the overall effort estimation. Additionally, different charts such as Work Breakdown Structure (WBS), Gantt chart, and Burn Down Chart have been used to provide a plan and an estimated duration of each task to act as a guide.

1.4 References and Acknowledgments

[1] Major consequences of poor employee scheduling - A complete guide. eResource Scheduler - Resource Management, Planning, and Scheduling Software. (n.d.). Retrieved September 23, 2022, from https://www.eresourcescheduler.com/blog/consequences-poor-employee-scheduling

[2] Waller, J. (2021, April 15). *The cost of poor scheduling*. Findmyshift. Retrieved September 23, 2022, from https://www.findmyshift.com/blog/the-cost-of-poor-scheduling

[3] Biz 3.0. 2022. *8 Common Employee Scheduling Issues (And How To Tackle Them) - Biz 3.0*. Retrieved 23 September 2022. from https://biz30.timedoctor.com/scheduling-issues/

[4] Blume, Ashley. "Healthcare Data Encryption Methods for Healthcare Providers." HealthITSecurity, HealthITSecurity, 7 Nov. 2012, https://healthitsecurity.com/news/healthcare-data-encryption-methods-for-healthcare-providers

[5] "5 Industries Most at Risk of Data Breaches." Www.ekransystem.com, 27 June 2019, www.ekransystem.com/en/blog/5-industries-most-risk-of-data-breaches.

[6] "Worst Breach of Personal Data in Singapore's History" Attracts Highest Penalties Totalling S$1 Million." Technology Law Dispatch, 15 Jan. 2019, www.technologylawdispatch.com/2019/01/privacy-data-protection/worst-breach-of-personal-data-in-singapores-history-attracts-highest-penalties-totalling-s1-million/.

2 Overall Description

2.1 Product Overview

The Workload Management System is the company's new self-contained product. It requires the usage of computers, web systems, and also databases to work together with one another. The Workload Management system consists of these 3 different layers and Fig 2.1 shows the overall system and their interconnections.

- Presentation Layer
  In this layer, there will be 3 different types of users: Staff, Manager, and IT Admin. Each of these users will have its own interfaces, and each of them will have its own responsibilities and roles. As this is a web-based website, the users need to use either a computer or a mobile phone to access it. Therefore, they are required to know how to use them. By making it accessible on both the website and mobile, it will be more convenient for the users to access this system at any time, anywhere. The website also needs to be user-friendly (easy to use), and understandable.

- Business Logic Layer
  In this layer, we use HTML and Python. There are also a few web interfaces that allow the different users to input and read the information.

- Data Layer
  In the data layer, we are using a database that is hosted on a website called myWindowsHosting (a cloud platform). This database will include data like employee's information, staff work schedule, staff job allocation, etc.

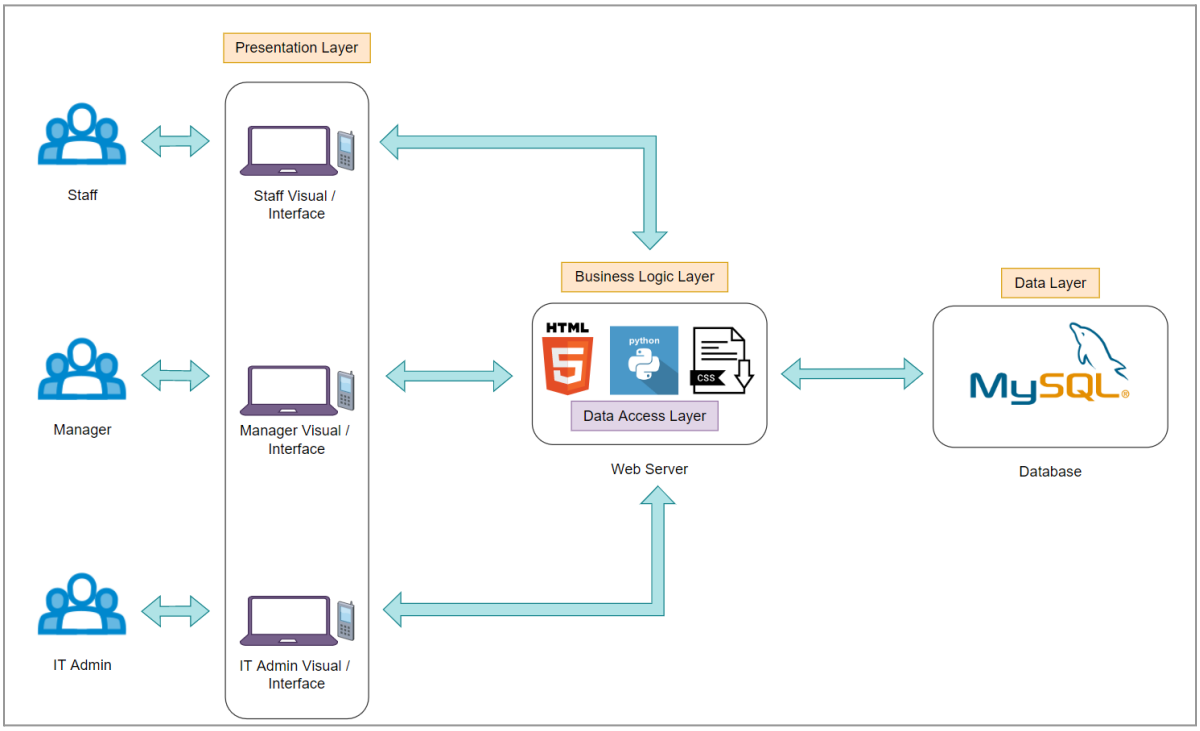


Figure 2.1: Systems Architecture Diagram

How the product works is that the system has to communicate with the web server. The web server will then communicate with the database server to retrieve the relevant information that is needed for the website to run. it then returns the requested data and the website will format and display it nicely for the users to view. When the user wants to make changes to the database, the website will relay this request to the web server, which will then communicate to the database server to update these changes. Once it is successful, the web server will show a success message on the screen to inform them.

To improve the security of the website, only the IT administrator is allowed to add and remove new or old staff from the system. for those new users, a new account User ID and password will be handed over to them.

2.2 Product Functionality

PF1	The product shall have different login rights for the different users (Staff, Manager, IT Admin) to log in to their account.
PF2	The product shall allow the manager account to view all staff workload on the manager landing page.
PF3	The product shall allow the manager account to view all staff details.
PF4	The product shall allow the manager account to view all rejected job assignments and make a decision for each record.
PF5	The product shall allow the manager account to allocate jobs for staff in the view job allocation page.
PF6	The product shall allow the manager account to edit the allocated jobs for staff in the schedule in view job allocation page
PF7	The Product shall allow the manager account to view staff job preference.
PF8	The product shall allow staff accounts to view their own profile.
PF9	The product shall allow a staff account to view rejected job status at the staff individual landing pages.
PF10	The product shall allow staff accounts to edit job availability at staff individual landing pages.
PF11	The product shall allow the staff account to add job availability to the individual staff landing page.
PF12	The product shall allow the staff account to view the overall workload for the month at the individual staff landing page.
PF13	The product shall allow staff accounts to view weekly job assignments at staff individual landing pages.
PF14	The product shall allow staff accounts to accept or reject weekly job assignments at individual staff landing pages.
PF15	The product shall allow staff accounts to indicate their preferred shifts at the individual staff landing page.

PF16	The product shall allow staff to indicate job preference on the individual staff landing page.
PF17	The product shall allow staff to indicate their location on the individual staff landing page.
PF18	The product shall allow the IT Admin account to create an account for new users (Staff. Manager, IT Admin).
PF19	The product shall allow the IT Admin account to delete the account for users who had quit.
PF20	The product shall allow the IT Admin account to view all user account details.

2.3 Assumptions and Dependencies

Some of the following assumptions made are:

1. IT Administrators are the only ones responsible for adding new staff, and removing those staff that has left.
2. The product will be deployed on cloud hosting services.
3. All staff shall own at least 1 device, and these devices must be able to connect to the internet.
4. All staff must be able to read and understand English.
5. All staff must know how to use the basic skills of the mobile phone or computer.
6. Clients want it to be used on all the devices available (mobile devices, tablets, computers)
7. All the staff have been taught and read through how to use the website.
8. All users will adhere to their roles and the company rules (eg, the manager assign work every Monday, staff informs job availability by Wednesday)

Some of the dependency are:

1. MySQL (We depend on this database to store all the company's records.)

3 Specific Requirements

3.1 User Interface Requirements

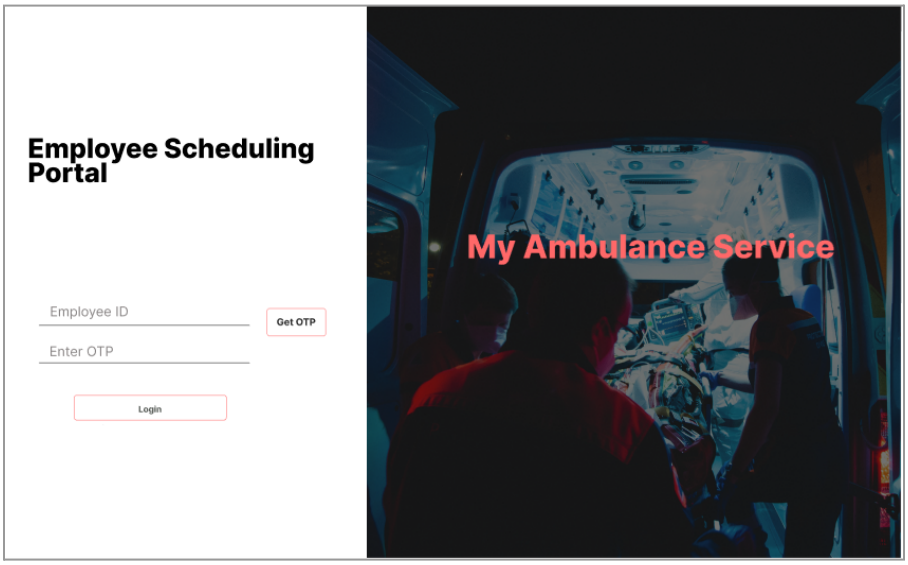


Figure 3.1 Employee Scheduling Portal

Users click the “Get OTP” button for an OTP
Users enter employee ID and OTP to gain access to the system.
Once both ID and OTP are keyed in, users will press the login button to enter the system.

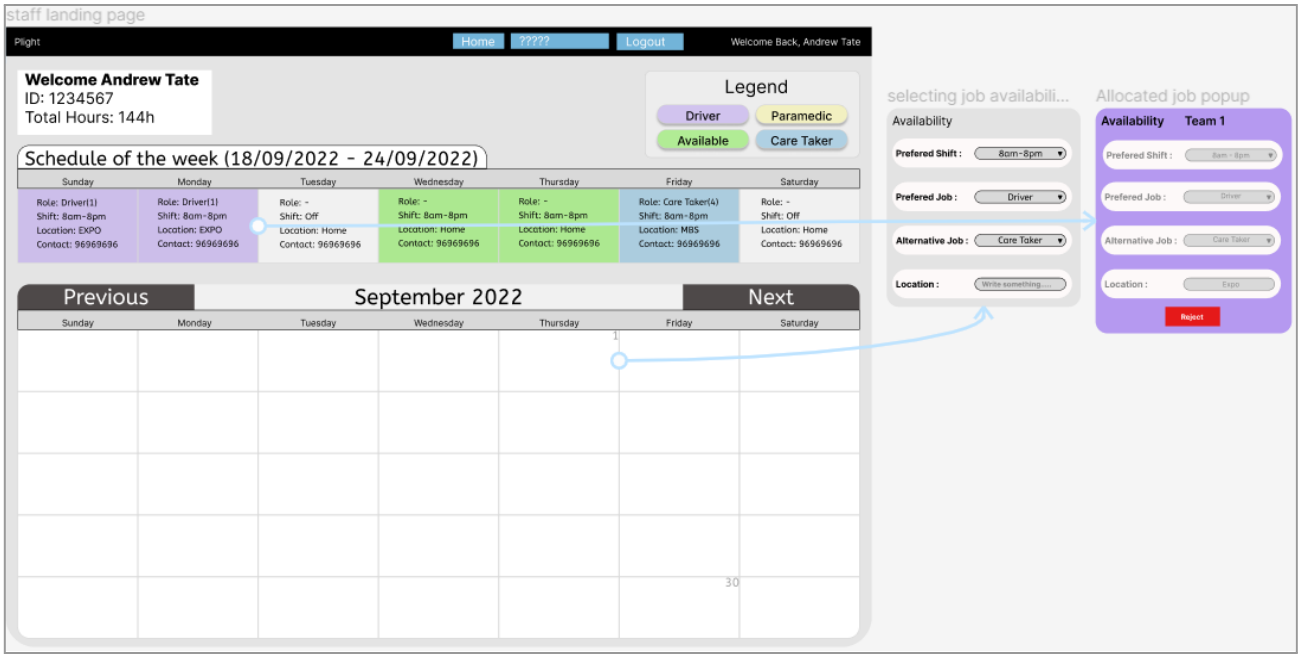


Figure 3.2 Staff Landing Page

Users can click on the home or logout button to go to either the landing page or logout respectively.
Users can click on the segregated boxes in the schedule for the week to reject allocated jobs.
Users can click on empty boxes on the calendar to input their job and shift preferences of the day, and input location for that day.

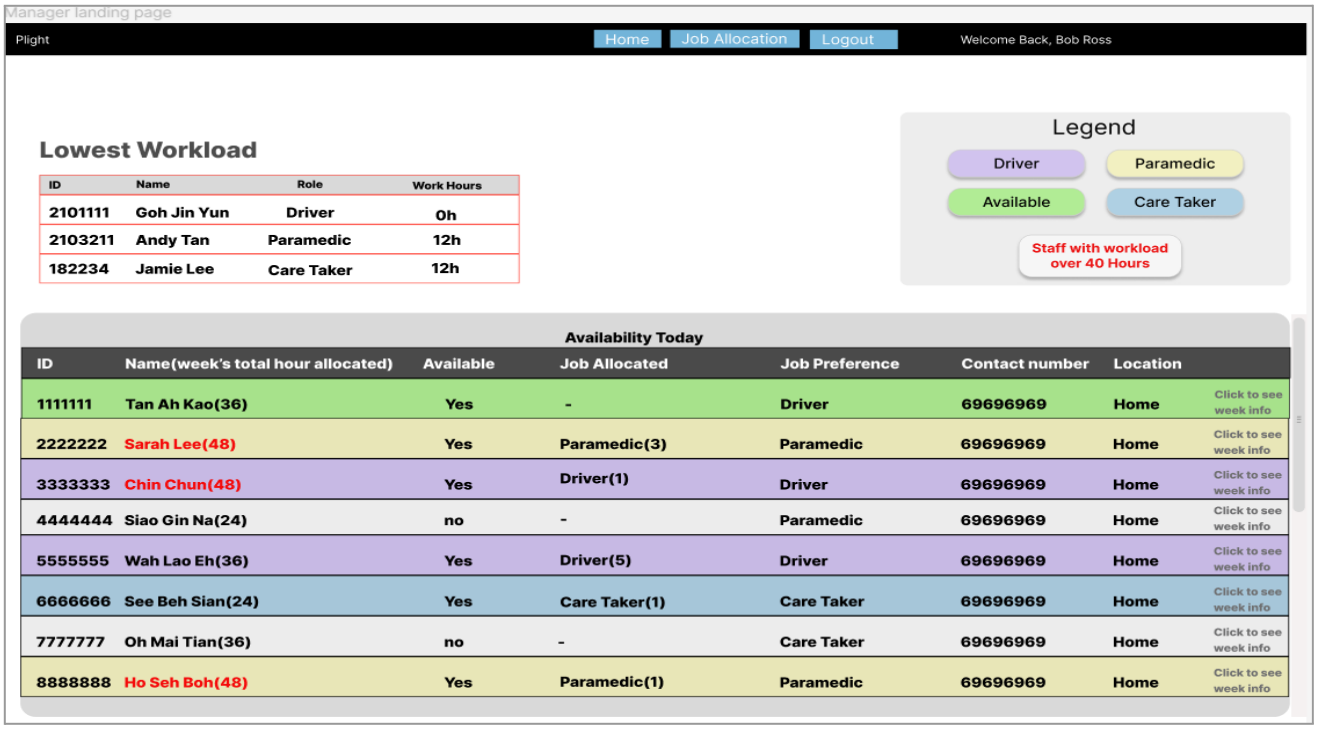


Figure 3.3 Manager Main Landing Page

Managers can click on the home button to go to the landing page, the job allocation button to the job allocation page, or the logout button to log out of the system.
They are able to scroll down to see all staff availability of the day, if they need to know the details of the week, they can click the hyperlink on the right-hand side of each row.

Figure 3.4 Job Allocation Page

The manager will have 3 staff with the lowest hours allocated for the week staff shown for each job at the top right-hand side of the screen. names can be dragged and dropped into each specific box for ease of allocation.
Staff names will change based on the day and shift selected through clicking according to the availability set by staff.
A drop-down box is given to let the manager select the team they allocate the staff to. If they require more teams, they can click the “+” button below the 6th team option to add more teams as needed.

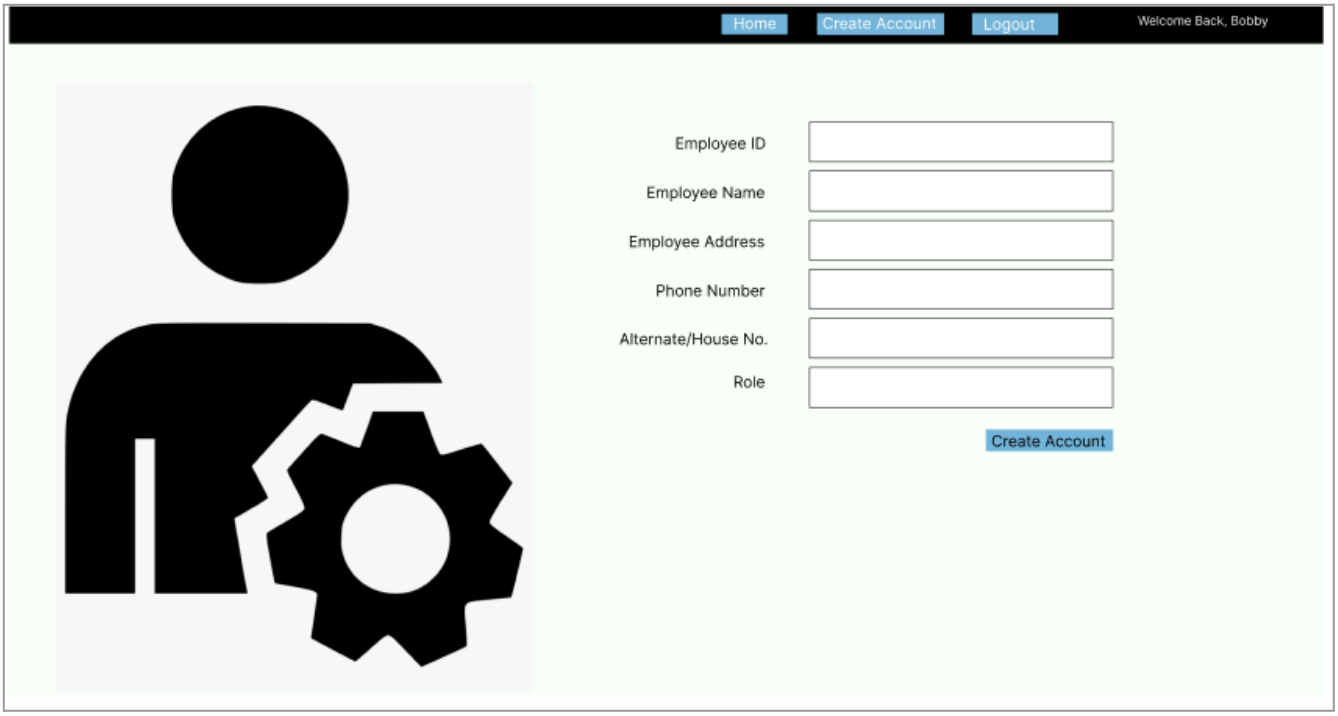


Figure 3.5 Create an Account (Only)

IT admins can create accounts for the new employees by inputting a new ID, the employee's name, address, phone number, other address if any, and the role they are applying for.
After all the inputs are entered, the IT admins will click the create account button to create the account.

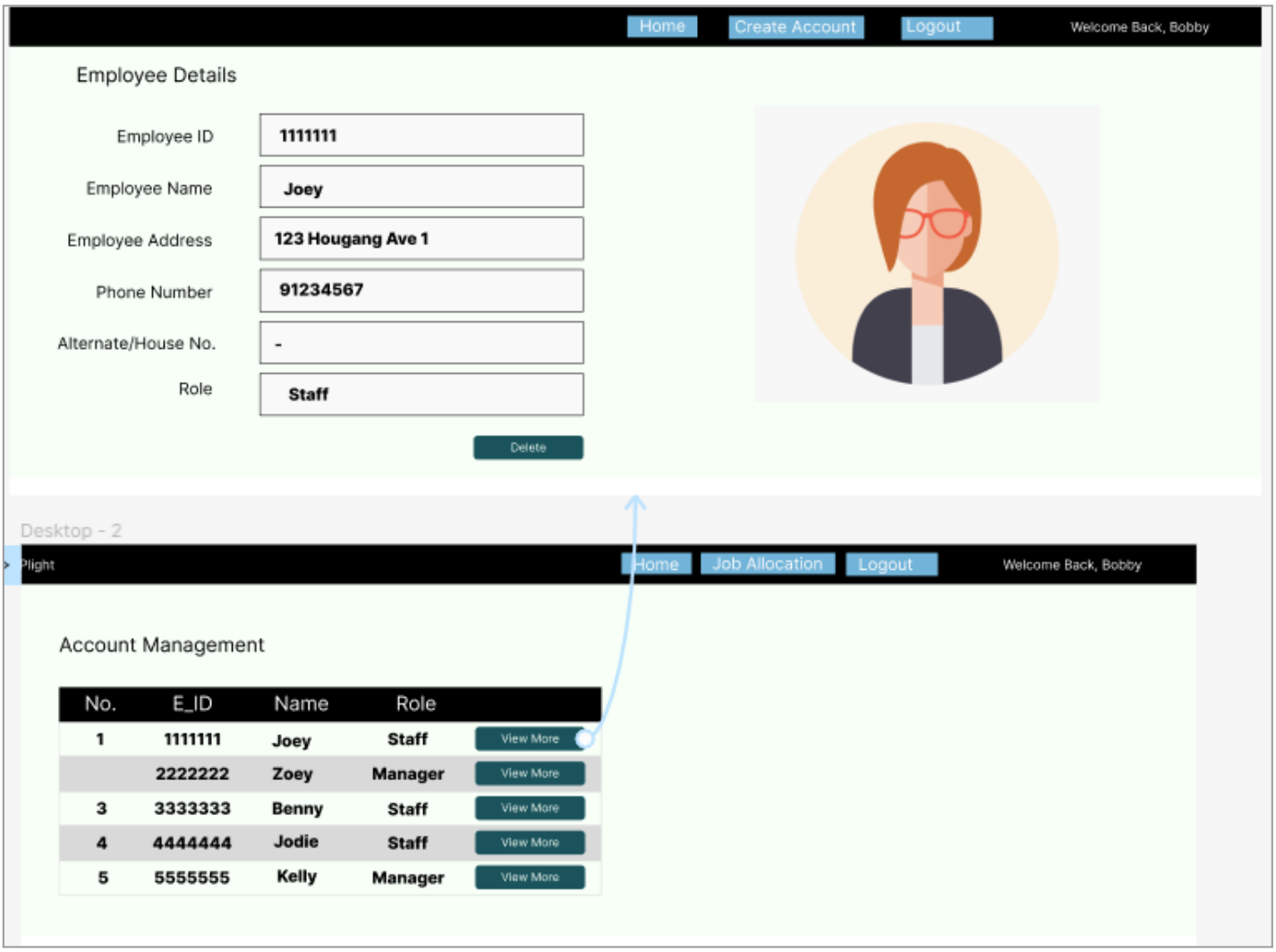


Figure 3.6 Remove an Account (IT Admin)

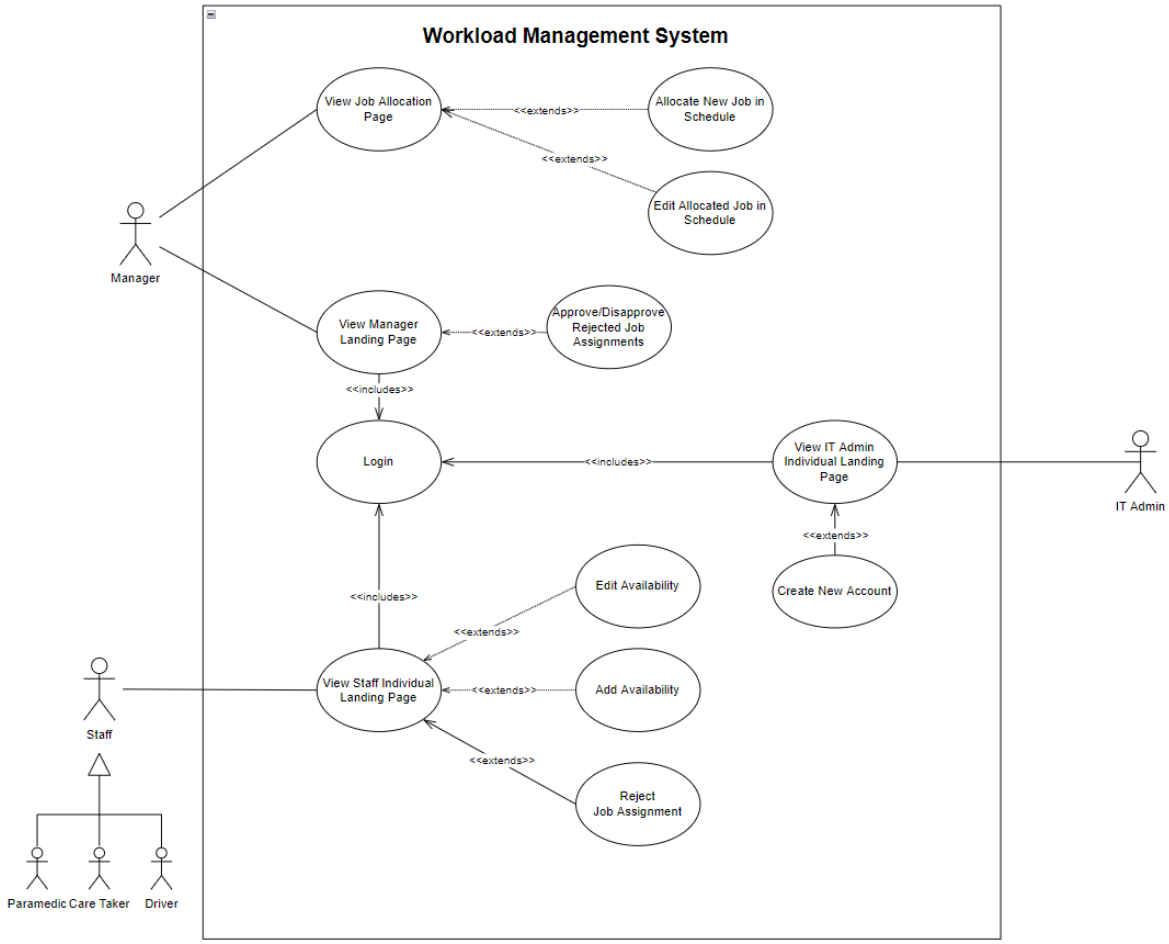
IT admins are able to manage the accounts of all employees, the IT admins would be able to go to the employee's account page using the "view more" button to edit the employee information or click the delete button to delete the account.

3.2 Functional Requirements

FR1	The system shall require that all users enter their identification number as the username and one-time password (OTP) when logging in so that the system can authenticate their identity.
FR2	All managers shall be able to visualize all staff workload.
FR3	The system shall show all staff information sorted in ascending order starting with the lowest hours at the top of the manager loading page.
FR4	The system shall highlight all staff with work hours above 40 for the week on the manager loading page.
FR5	All managers shall only allocate 1 weeks worth of jobs at a time.
FR6	The system shall show up to 3 staff availability on the job allocation page.
FR7	The system shall show the available staff information (name, id, preferred shifts, job preference, location) on the job allocation page.
FR8	All managers shall be allowed to select all available staff to allocate jobs to.
FR9	The system shall display the following information (location of the job driver name, caretaker name, both paramedic names) for both main and backup teams on the job allocation page.
FR10	The system shall notify employees' respective job rejections through app notification.
FR11	All staff shall be able to view their respective weekly job and overall workload assignments on the landing page.
FR12	The system shall allow staff to add and edit availability selected shifts and preferred jobs they are available for up to 5 weeks ahead of time on the respective staff landing page.
FR13	All staff shall be able to reject job allocations on the respective staff landing page.
FR14	The system shall allow staff to view rejected job status on the respective staff landing page.

FR15	All staff shall be able to indicate their preferred shift, job, and location of selected shift on the respective staff landing page when submitting job availability.

3.3 Use Case Model



3.4 Non-functional Requirements

NFR1	Loading of all screens should not take longer than 2 seconds.
NFR2	Encryption would be done for all information in the system.
NFR3	Login is required to use the system.

3.4.1 Performance Requirements

NFR4	The system shall not have a downtime of more than 5 seconds in any one day.
NFR5	The system shall give a response after authentication for login within 2 seconds.
NFR6	The system shall not take longer than 5 seconds to load all staff information on the manager landing page.
NFR7	The system shall update all given information within 2 seconds of submission.
NFR8	The system has a simplistic design to enhance understandability, allowing users to immediately understand the information given.

3.4.2 Safety and Security Requirements

NFR9	Data transmitted from and received are always encrypted with AES-128.
NFR10	All approving roles must use employee ID and randomized OTP to log in.

4 Project Estimation and Plan

4.1 Software Estimation

4.1.1 Use Case Points

Unadjusted weight for use case

Use case	number of transactions	
1	3	Simple
2	5	Average
3	5	Average
4	3	Simple
5	6	Average
6	5	Average
7	1	Simple
8	5	Average
9	5	Average
10	5	Average
11	1	Simple
12	1	Simple

Use Case complexity and weight		Number of use case	Product of weight with number
Simple	5		5*5 = 25
Average	8		7*10 = 70
Complex	0		-

Unadjusted weight of use case = 95

Unadjusted weight of actors

Actor type	Number of actors	Product
Simple (1)	0	0
Average (2)	1	2
Complex (3)	3	9

Unadjusted weight of actors = 0 + 2 + 9 = 11

Unadjusted Transactions Weight + Unadjusted Actors Weight = 95+ 11 = 106

Technical Complexity Factor (TCF)

	Factor	Weight	degree of influence	Weight
Distributed system	1	2	2	4
Response time/performance objectives	2	1	4	4
End-user efficiency	3	1	5	5
Internal processing complexity	4	1	3	3
Code reusability	5	1	1	1
Easy to install	6	.5	1	.5
Easy to use	7	.5	5	2.5
Portability to other platforms	8	2	5	10
System maintenance	9	1	3	3
Concurrent / parallel processing	10	1	1	1
Security features	11	1	3	3
Access for third parties	12	1	1	1
End user training	13	1	2	2
			Total	40

TCF = 0.6 + 0.01 * 40 = 1

Environment Factor (EF)

Environment Factors	Weight	Assessment 0-5	
Familiar with development process	1.5	3	4.5
Part time workers	-1	0	0
Analyst capability	.5	4	2
Application experience	.5	3	1.5
Object oriented experience	1	2	2
Motivation	1	4	4
Difficult programming language	-1	1	0
Stable requirements	2	4	8
		Total	22

EF = 1.4 + (-0.03 * 22) = 0.74

UCP = UUCW * TCF * EF

UCP = 106 * 1 * 0.74

UCP = 78.44

Effort Estimation

15 * 78.44 to 30 * 78.44

Total Hours 1176.6 Hours to 2353.2 Hours

Individual Hours 294.15 Hours to 588.3 Hours

4.2 Project Management

4.2.1 Work Breakdown Structure (WBS)

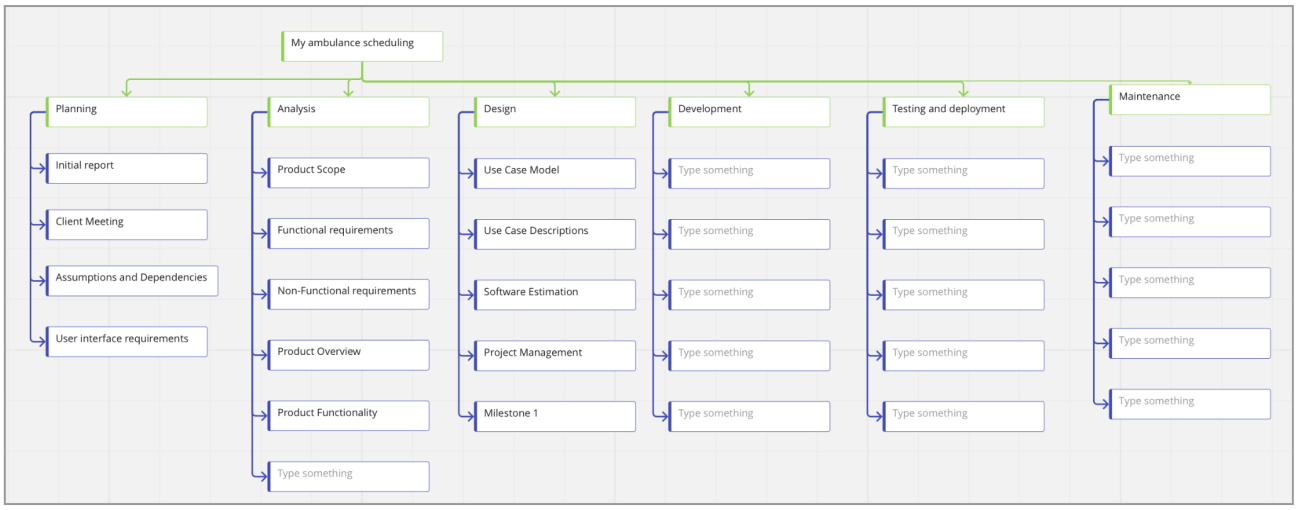


Figure 4.1 Work Breakdown Structure (WBS)

4.2.2 Gantt Chart

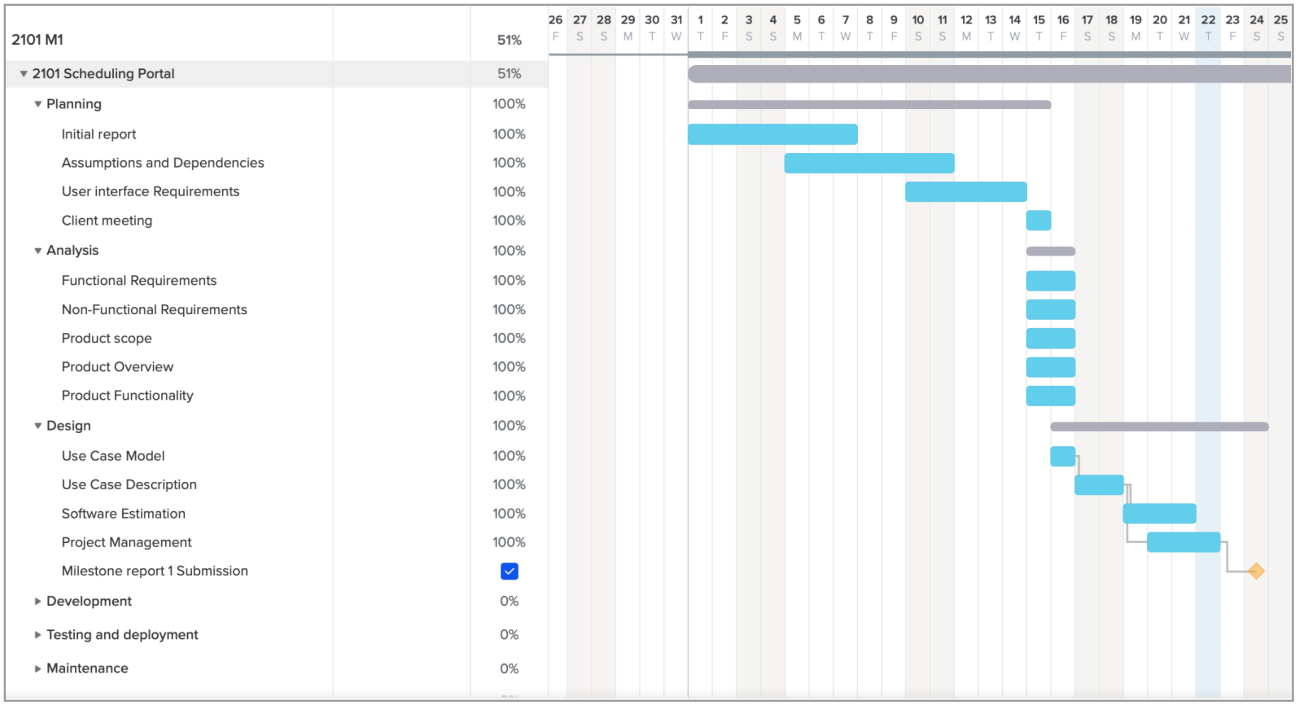


Figure 4.2 Gantt Chart

4.2.3 Burn Down Chart

Burn down chart

	Burned down		Balance		Daily
Week	Planned	Actual	Planned	Actual	Completed
0			1176.6	250	#N/A
1	12	15	1164.6	235	15
2	18	15	1146.6	220	15
3	25	23	1121.6	197	23
4	30	32	1091.6	165	32
5	60		1031.6	#N/A	#N/A
6	45		986.6	#N/A	#N/A
7	20		966.6	#N/A	#N/A
8	34		932.6	#N/A	#N/A
9	30		902.6	#N/A	#N/A
10	30		872.6	#N/A	#N/A
11	25		847.6	#N/A	#N/A
12	40		807.6	#N/A	#N/A
13	30		777.6	#N/A	#N/A

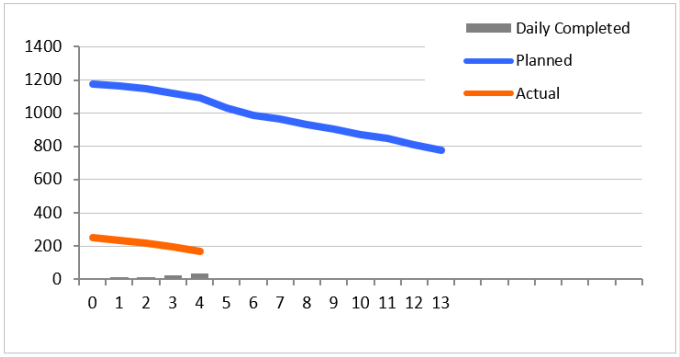


Figure 4.3 Burn Down Chart

5 Individual Members Task Reflections

Tennyson Koh (2100816)

I have done the Project Management and software estimation. After doing all these tasks, I learned that creating softwares/programs is not all about coding. There are so many more things that happen in the background like the software planning in order to ensure that money is not lost due to budget and time. A platform to properly display tasks is also very important as it helps the developer visualize what needs to be done, and which stage they are currently in. I feel that communication is also something that is very important. We as developers have to communicate with our team members in order to ensure maximum efficiency and also it also can help avoid miscommunications.

If I have another chance to do this all over again, I would ensure me and my team document our progress more specifically on our project board on GitHub, as this helps greatly in communication which is stated above. Fewer miscommunications would happen.

Anna Ng (2101528)

For milestone one, I have done the introduction. Through the client meeting, we were able to identify some of the issues they have but it was not sufficient to fully understand why such issues happen or what would be the consequences of these issues. Thus, through the related background literature section, I have learnt the importance of understanding the troubles the clients are facing when it comes to scheduling. With the help of doing some research, some consequences and solutions were identified for us to take into consideration when planning out the functionalities of our system for the client. Given the chance to do this project again, meeting the group physically to discuss the project would be preferred as discussing the project physically allows everybody's points or ideas to be understood better. If there were points that we did not understand, we were able to get clarifications immediately and clearly as compared to texts. Ultimately, preventing any form of misconceptions or misunderstandings from happening.

Tan Ai Xin (2102468)

In milestone one, I have done the project overall description. After completing the task, I have learned that there are many different methods of doing this project. When I was doing the overall description, I learned to put myself in the different user’s point of view, so that I could understand the problems that they are currently facing. By understanding their current needs, as a team, we can then come up with a product that can assist them and help to solve their problems in their daily life. I also learned the importance of asking the clients of their needs so that we can create a product that is up to their standards and also check that we are still on the correct path of what the client really wanted. By listing out all the products functionality, we can also visualise better on what should be the key points that we should be focusing on. If I am given another chance to do this project again, I would meet my group to discuss and write down all the questions that we really wanted to know. Before the client meeting, I would also list out all the product functionality, functional and nonfunctional requirements, and assumptions also. In my opinion, when we list out all these points first, we can better understand which are the things that we know or do not know.

Tan Chun Guan (2102892)

In milestone one, I have done the specific requirements. I learned that the specification of requirements has to be very precise to give us a good idea of the customer's wants and needs. Breaking down all the information given to us at the start and moving on with a client meeting to extract more precise information to generate a product for them is not an easy task as we need to analyze every detail so that we would not misinterpret the information at hand. Creating clear and concise functional requirements benefitted us in making the interface of the product. Given another chance to do this again, I would like to have focused on information relating to the statements given to us at the start of the project hand in hand with the milestone 1 requirements, and not go off track by asking things that are a little out of scope during the client meeting.

6 Appendix A – Use Case Descriptions

Use Case ID:	UC-1
Use Case Name:	**Create new account**
Description:	A page IT admin to create an account for employees for the online scheduling portal
Primary Actor:	IT Admin
Preconditions:	1. Employee must not already have an account 2. IT Admin must already be in the IT Admin individual landing page
Postconditions:	Employee account created

Main Success Scenarios:	1. IT admin click create an account button
	2. System displays a few imputes which includes
	  1. Employee ID
	  2. Employee name
	  3. Employee address
	  4. Phone number
	  5. Alternate/house number
	  6. Role
	3. IT admin fills in all imputes and click submit
	4. Systems validates information
	5. System send all data into the database, creating an account for the employee
	6. The system sends a confirmation message that the account has been successfully created.
	7. Use case ends.

Alternative Scenarios:	3a. IT admin keys in data in the wrong format 1. Systems validate the information. 2. The system displays an error message advising IT admin to follow the given format. 3. The use case resumes at main flow step 3.

Use Case ID:	UC-2
Use Case Name:	**Login**
Description:	A page for all employees including managers to login to their account for the online scheduling portal
Primary Actor:	All Employees
Preconditions:	Employee must already have created an account
Postconditions:	Employee will be logged in into portal
Main Success Scenarios:	1. Employee click Login option 2. System displays an input for the employee ID 3. Employee fills in employee ID and click submit 4. System shows an extra field for ONE TIME PASSWORD with the a short text “OTP has been send to your mobile number” 5. Employee keys in ONE TIME PASSWORD from their SMS and click submit 6. Systems validates information 7. System send data to database to check for the correct details 8. Systems displays success message that says Login success 9. System shows the landing page of the scheduling portal 10. Use case ends
Alternative Scenarios:	3a. Employee keys in wrong or non-existent employee ID. 1. The system displays an error message advising employees to key in the correct employee ID. 2. Systems validate information. 3. The use case resumes at main flow step 3. 5a. Employee keys in wrong ONE TIME PASSWORD 1. Systems displays error message advising employee to key in the correct ONE TIME PASSWORD 2. The use case resumes at main flow step 5.

Use Case ID:	UC-3
Use Case Name:	**Add availability**
Description:	A page where staff can indicate when they are available to work for the online scheduling portal
Primary Actor:	Employees for ambulance team
Preconditions:	Employees are in their respective landing page
Postconditions:	Employees are able to see updated landing pages where recently added availability is there.

Main Success Scenarios:	1. Employee clicks add availability option
	2. System displays a few imputes which includes
	    1. Day they are available
	    2. Shift for that day
	    3. Job preference
	    4. Location for that day
	3. Employee selects/fills in inputs and click submit
	4. System validates information
	5. System sends all data into the database, updating employee availability
	6. System displays success message saying availability successfully added
	7. System returns to employee landing page

Alternative Scenarios:	3a. Employee missed out certain inputs 3a1. System displays error message indicating inputs that are missing, then returns to step 3 5a. Updating failed 5a1. System gives a message that the updating has failed and offer retry option 5a2. Employee can choose to retry or quit adding screen

Use Case ID:	UC-4
Use Case Name:	**Rejecting job assignment**
Description:	A page where staff can reject their job assignment
Primary Actor:	Employees for ambulance team
Preconditions:	Employees are in their respective landing page
Postconditions:	Manger will see the reason for rejecting the job
Main Success Scenarios:	1. Employee find that there is a job assignment they do not want 2. Employee clicks on job assignment 3. System displays selected job assignment in full detail with rejection button 4. Employee clicks rejection button 5. System displays input for rejection reason 6. Employee fills in input 7. System saves input into the database 8. System displays success message saying manager will review your rejection ticket and you can view your rejection status on the employee landing page once it has been resolved. 9. System returns to employee landing page
Alternative Scenarios:	7a. Message was not sent 7a1. System gives a message that the rejection message was not sent and offer retry option 7a2. Employee can choose to retry or quit rejection screen

Use Case ID:	UC-5
Use Case Name:	**View Manager Landing Page**
Description:	This is a page for the Manager to view all the staff's workload in the manager landing page.
Primary Actor:	Manager
Preconditions:	1. The manager must have an existing account. 2. The manager must be logged in to their account. 3. Manager must be at the Landing Page
Postconditions:	NIL
Main Success Scenarios:	1. system retrieves all staff details (job assignments, staff working hours engaged, staff working hours assigned, staff availability (Status), and staff role) from the database. 2. System displays retrieved data 3. system retrieves the top 3 staff with the lowest workload record. 4. System displays retrieved data 5. system retrieves all staff records who work for more than 40h. 6. System displays retrieved data 7. system displays "All records retrieved successfully". 8. The use case ends.
Alternative Scenarios:	1a. Error in retrieving all Staff Workload records 1. The system displays "Unable to load Staff Workload. Please try again". 2. The Use case resumes at main flow step 1. 2a. Error in retrieving top 3 staff with the lowest record. 1. The system displays "Unable to load top 3 staff with lowest record. Please try again". 2. The Use case resumes at main flow step 1. 3a. Error in retrieving all staff records who work for more than 40h. 1. The system displays "Unable to load all staff records with who work for more than 40h. Please try again". 2. The Use case resumes at main flow step 1.

Use Case ID:	UC-6
Use Case Name:	**Approve/Disapprove rejected job assignments**
Description:	This is a page for the Manager to view all the rejected job assignments that the staff had rejected. The manager can choose to approve or reject the job request.
Primary Actor:	Manager
Preconditions:	1. The manager must have an existing account. 2. The manager must be logged in to their account. 3. Manager must be at the Landing Page
Postconditions:	
Main Success Scenarios:	1. Manager selects the "Rejected Job Assignments" option. 2. System loads Rejected Job Assignments page. 3. The system displays all records of jobs that have been rejected by the staff. 4. The manager selects a staff record that has been rejected. 5. System displays a. Name of employee who rejected the job b. Job assignment c. Time and date of job assignment d. Reason for rejection e. Approve and Reject button 6. The manager selects the "Approve" button to approve the rejection on that job assignment. 7. system display confirmation message "Approve this job rejection?" 8. System updates **status** for staff. 9. System displays success message 10. System will update the status of the job on the view rejected job status page 11. Use case ends
Alternative Scenarios:	5a. Manager selects "Reject" button 1. system display confirmation message "disapprove this job rejection?" 2. the use case resumes at main flow step 8

Use Case ID:	UC-7
Use Case Name:	**View Job Allocation**
Description:	This is a page for the Manager to view at least 3 staff availability.
Primary Actor:	Manager
Preconditions:	1. The manager must have an existing account. 2. The manager must be logged in to their account. 3. Manager must be at the Landing Page
Postconditions:	
Main Success Scenarios:	1. Manager selects "View Job Allocation" menu link 2. The system retrieves all records of the staff that are available. 3. System display the results in a table 4. The use case ends
Alternative Scenarios:	

Use Case ID:	UC-8
Use Case Name:	**Allocate new Job in Schedule**
Description:	This is a page for the manager to add new job assignments for the staff.
Primary Actor:	Manager
Preconditions:	1. The manager must have an existing account. 2. The manager must be logged in to their account. 3. Manager must be at the Add Job Allocation Page 4. Calendars are automatically generated within 1 month from the current date. (5 weeks, September & October)
Main Success Scenarios:	1. System display “Add Job Allocation” page 2. The manager selects a month. 3. The manager selects a day. 4. System prompts manager for details 5. Manager enters job allocation schedule for that day and submits 6. System validates the details 7. System verify the details 8. The manager confirms the details entered. 9. System saves the details of the staff job allocation in Database 10. System display success message 11. The use case ends
Alternative Scenarios:	6a. Duplicated staff recorded. 1. System display error message “Duplicated Staff Record Found”. 2. System prompt manager to re-enter details 3. The use case resumes at main flow step 5 9a. Records is not save successfully 1. System display error message “Records are not saved successfully. Please try again.”. 2. The use case resumes at use case step 2.

Use Case ID:	UC-9
Use Case Name:	**Edit Allocated Job in Schedule**
Description:	This is a page for the manager to edit the job assignments of the staff.
Primary Actor:	Manager
Preconditions:	1. The manager must have an existing account. 2. The manager must be logged in to their account. 3. Manager must be at the Add Job Allocation Page 4. Calendars are automatically generated within 1 month from the current date. (5 weeks, September & October)
Postconditions:	
Main Success Scenarios:	1. System display “Edit Job Allocation” page 2. The manager selects a month. 3. The manager selects a day. 4. Manager selects on “Edit” button 5. Manager enters job allocation schedule for that day and submits 6. System validates the details 7. System verify the details 8. The manager confirms the details entered and clicked submit. 9. System saves the details of the staff job allocation in Database 10. System display success message 11. The use case ends
Alternative Scenarios:	6a. Duplicated staff recorded. 1. System display error message “Duplicated Staff Record Found”. 2. System prompt manager to re-enter details 3. The use case resumes at main flow step 5 9a. Records is not updated successfully 1. System display error message “Records are not updated successfully. Please try again.”. 2. The use case resumes at use case step 2.

Use Case ID:	UC-10
Use Case Name:	**Edit availability**
Description:	A page where staff can edit job availability which they have set in the past
Primary Actor:	All employees for ambulance team
Preconditions:	Employees are in their respective landing pages.
Postconditions:	Employees are able to see updated landing pages where recently added availability is there.
Main Success Scenarios:	1. Employee finds job availability they want to change 2. Employee clicks on job availability they set previously 3. System displays selected job availability in full detail with edit button 4. Employee clicks edit button 5. System displays a few imputes which includes 1. Day they are available 2. Shift for that day 3. Job preference 4. Location for that day 6. Employee selects/fills in inputs and click submit 7. System validates information 8. System sends all data into the database, updating employee availability 9. System displays success message saying availability successfully added 10. System returns to employee landing page
Alternative Scenarios:	6a. Employee missed out certain inputs 1. The system displays an error message indicating inputs that are missing. 2. The use case resumes at main flow step 3.

	8a. Updating failed 1. The system gives a message that the updating has failed and offers a retry option. 2. Employees can choose to retry or quit adding screen.

Use Case ID:	UC-11
Use Case Name:	**View staff individual landing page**
Description:	A landing page to show overall workload for the month in hours, job assignment for the week for staff
Primary Actor:	Staff
Preconditions:	Staff must be logged in with a staff account
Postconditions:	
Main Success Scenarios:	1. Systems loads and displays a. overall workload for the month in hours b. weekly job assignment
Alternative Scenarios:	

Use Case ID:	UC-12
Use Case Name:	**View IT Admin individual landing page**
Description:	A landing page to show tools that are only accessible by the IT admin
Primary Actor:	IT Admin
Preconditions:	IT Admin must be logged in with a IT Admin account
Postconditions:	
Main Success Scenarios:	1. Systems loads and displays a button that links to the Create New Account page.
Alternative Scenarios:	

7 Appendix B – Data Dictionary

EF

Environmental Factor

UCP

Use Case Points

TCF

Technical Complexity Factor