

QMePls

by Team Titans

Team Members

Jolene Tan (U1921255B) - Project Manager

Aloysius Seow (U1920159K) - QA Manager/Engineer

Soh Qian Yi (U1922306C) - Release Manager

Jacob Law (U1922430D) - Lead Developer

Samuel Lee (U1920006F) - Back-End Developer

Zeta Chua (U1922610B) - Front-End Developer

Outline

1. Product Introduction
2. Live Demonstration
3. Design for Maintainability
4. Software Quality Assurance
5. Project Management
6. Risk Management

Product Introduction

- Background
- Objective
- Product and Functionalities

Background

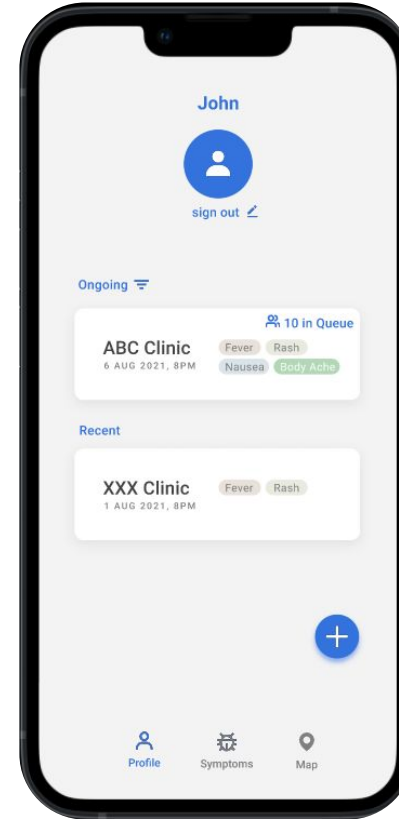
- Increasing waiting time in clinics, in addition to the rise of COVID-19 cases
 - A clinic in Sengkang was observed to have an average waiting time of 86 minutes 95% of time
- Patients dread waiting times at clinics, especially when ill
- Limited capacity due to social distancing measures

Objectives

- Enhance the queuing experience of patients
- Breach the gap between waiting time and which clinic to visit
- End-to-end clinic booking application

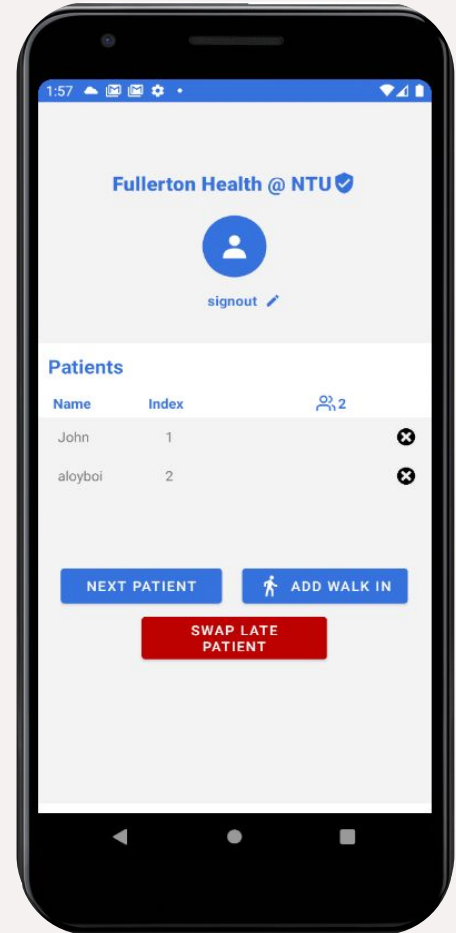
The Product

- 'QMePls' is an end-to-end Android application where users are able to check waiting times at nearby clinics, book queue slots and further input relevant symptoms as a log to ease consultation with doctors



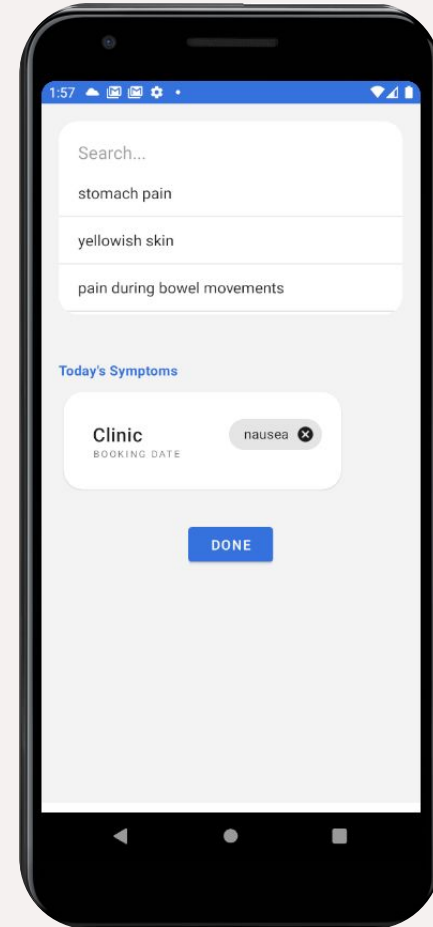
Login/User Authentication

- Two domains - Patient and Clinic
- Register and Login with Gmail Account



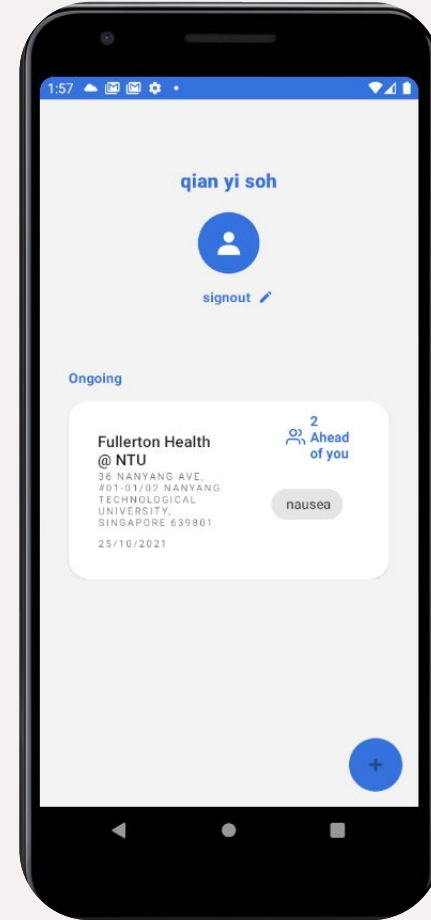
Symptom Selector

- Symptoms retrieved from database
- Users can input symptoms in fuzzy search bar
- Add symptom into health report card



Clinic Map & Book Appointment System

- Allows users to check current queue number for top 3 nearest clinics
- Allows users to book their slots for the selected clinic

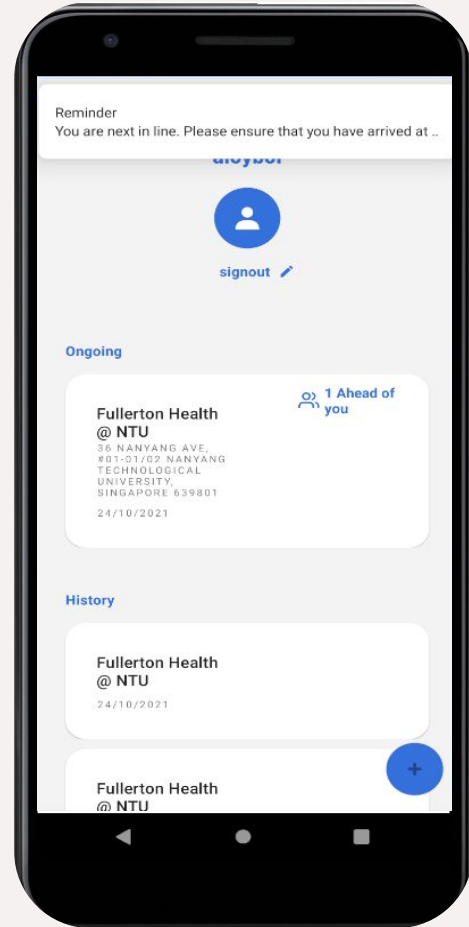


Queue Management System

- System must retrieve the queue information from the User database
- Clinic Staff User must be able to
 - add a walk-in patient
 - skip to the next patient
 - push patient index down by 2

Notification System

- Application sends notification to patient users when they are next in line



Live Demonstration

Design for Maintainability

1. Design Strategies
2. Architectural Design Patterns
3. Software Configuration Management Tools

Design Strategies

1. Planning Phase
2. Process of Development
3. **Correction by Nature**
4. Enhancement by Nature
5. **Maintainability Practices**

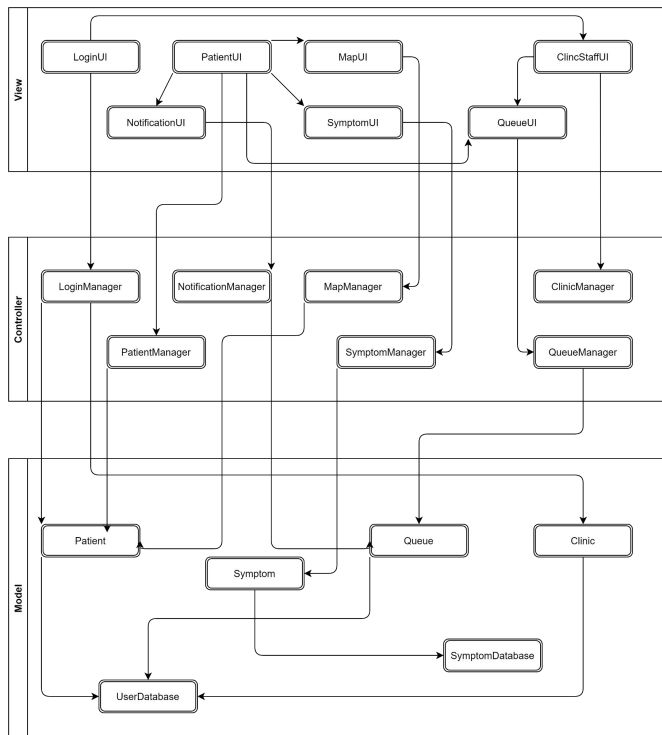
Correction by Nature

- Corrective Maintainability
- Preventive Maintainability

Maintainability Practices

- Codes are commented for better readability and ease of understanding
- Standardized documentation format
- Frequent and incremental changes to the code base
- Version control
- Loose coupling
- High cohesion
- Modularity

Architectural Design Patterns



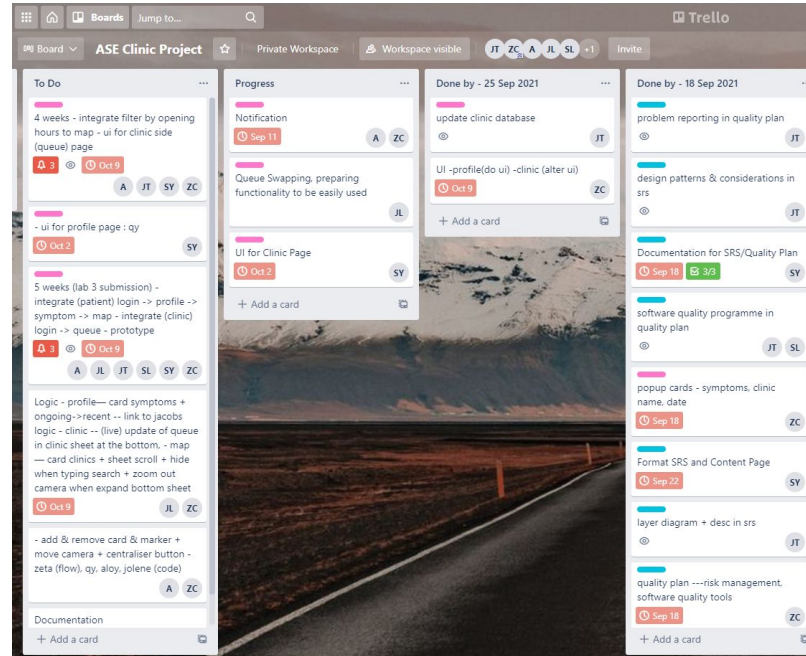
Model-View-Controller

In a Layered System Architecture
format

Software Configuration Management Tools



Trello



Software Quality Assurance

Ensures that QMePls meets and complies with defined or standardized quality specifications

Quality Assurance (QA)

Product Assessments

- Weekly checklist for functional requirements
- Weekly checklist for non-functional requirements

Process Assessments

- Weekly code review to be in line with software metrics
- Weekly progress check for QMePls against proposed Gantt Chart timeline
- Weekly meeting to verify adherence to requirements specification



Management Reviews

1. Weekly scheduled meetings
2. Cloud storage for documentations (Google Drive)
3. Evaluate effectiveness of management approach



Technical Reviews

1. Ensure design practices are followed and consistent
2. Follow closely with Software Metrics
3. Testing



Testing

Blackbox Test: Clinic Staff Testing


Test Case ID:	16
Test Executed By:	Aloysius Seow
Test Description:	Testing the ability to remove the first person in clinic queue with only one person in queue.
Priority:	Medium
Precondition:	<ol style="list-style-type: none">1. The system has an active connection to the internet.2. Clinic Staff User has already logged into the system under 'Clinic' Domain.3. The clinic queue must only have one patient in Queue
Test Data:	-
Test Steps:	<ol style="list-style-type: none">1. Click on 'Next Patient' button2. Click on 'Confirm'
Expected Results:	Clinic queue reduce to 0
Actual Result:	java.lang.NullPointerException: Attempt to invoke virtual method 'void android.widget.TextView.setVisibility(int)' on a null object reference
Status (Pass / Fail):	Fail
Comments:	App crashes when queue number becomes 0

Blackbox Re-test: Clinic Staff Testing



Re-test Case ID:	1
Test Executed By:	Aloysius Seow
Test Description:	Retesting Test Case 16 after code fix
Priority:	High
Precondition:	<ol style="list-style-type: none">1. The system has an active connection to the internet.2. Clinic Staff User has already logged into the system under 'Clinic' Domain.3. The clinic queue must only have one patient in Queue
Test Data:	-
Test Steps:	<ol style="list-style-type: none">1. Click on 'Next Patient' button2. Click on 'Confirm'
Expected Results:	Clinic queue reduce to 0
Actual Result:	Clinic queue reduce to 0
Status (Pass / Fail):	Pass

Comments:

Fullerton Health @ NTU ✓


signin ✎

Patients


Name	Index	 1
aloyboi	1	

NEXT PATIENT **ADD WALK IN**


SWAP LATE PATIENT

(Before)

Fullerton Health @ NTU ✓


signin ✎

Patients

Name	Index	 No one in queue
------	-------	---

NEXT PATIENT **ADD WALK IN**

SWAP LATE PATIENT

(After)

Build Verification Test (BVT)

```
> Task :app:mergeDebugAndroidTestAssets UP-TO-DATE
> Task :app:compressDebugAndroidTestAssets UP-TO-DATE
> Task :app:processDebugAndroidTestJavaRes NO-SOURCE
> Task :app:mergeDebugAndroidTestJavaResource UP-TO-DATE
> Task :app:mergeDebugAndroidTestJniLibFolders UP-TO-DATE
> Task :app:mergeDebugAndroidTestNativeLibs UP-TO-DATE
> Task :app:desugarDebugAndroidTestFileDependencies UP-TO-DATE
> Task :app:dexBuilderDebugAndroidTest UP-TO-DATE
> Task :app:checkDebugAndroidTestDuplicateClasses UP-TO-DATE
> Task :app:mergeExtDexDebugAndroidTest UP-TO-DATE
> Task :app:mergeLibDexDebugAndroidTest UP-TO-DATE
> Task :app:mergeProjectDexDebugAndroidTest UP-TO-DATE
> Task :app:validateSigningDebugAndroidTest UP-TO-DATE
> Task :app:packageDebugAndroidTest UP-TO-DATE
```

```
> Task :app:connectedDebugAndroidTest
```

```
Starting 1 tests on Pixel_2_API_30(AVD) - 11
```

```
> Task :app:connectedAndroidTest
```

```
BUILD SUCCESSFUL in 1m 13s
```

```
94 actionable tasks: 2 executed, 92 up-to-date
```

```
Build step 'Invoke Gradle script' changed build result to SUCCESS
```

```
No emails were triggered.
```

```
Finished: SUCCESS
```

Project Management

1. Project Organization
2. Project Estimation
3. Project Schedule

Project Organisation

1. **Team Structure**
2. Roles and Responsibilities
3. **Team Communication**

Team Structure

- Project Manager - Jolene Tan
- QA Manager/QA Engineer - Aloysius Seow
- Release Manager - Soh Qian Yi
- Lead Developer - Jacob Law
- Back-End Developer - Samuel Lee
- Front-End Developer - Zeta Chua

Team Communication

- Bi-weekly physical meetings
- Telegram group
- Weekly online discussions
- Splitting into subgroups

Project Estimation

1. Unadjusted Function Points
2. **Efforts, Duration and Team Size Estimation**
3. Cost Estimates

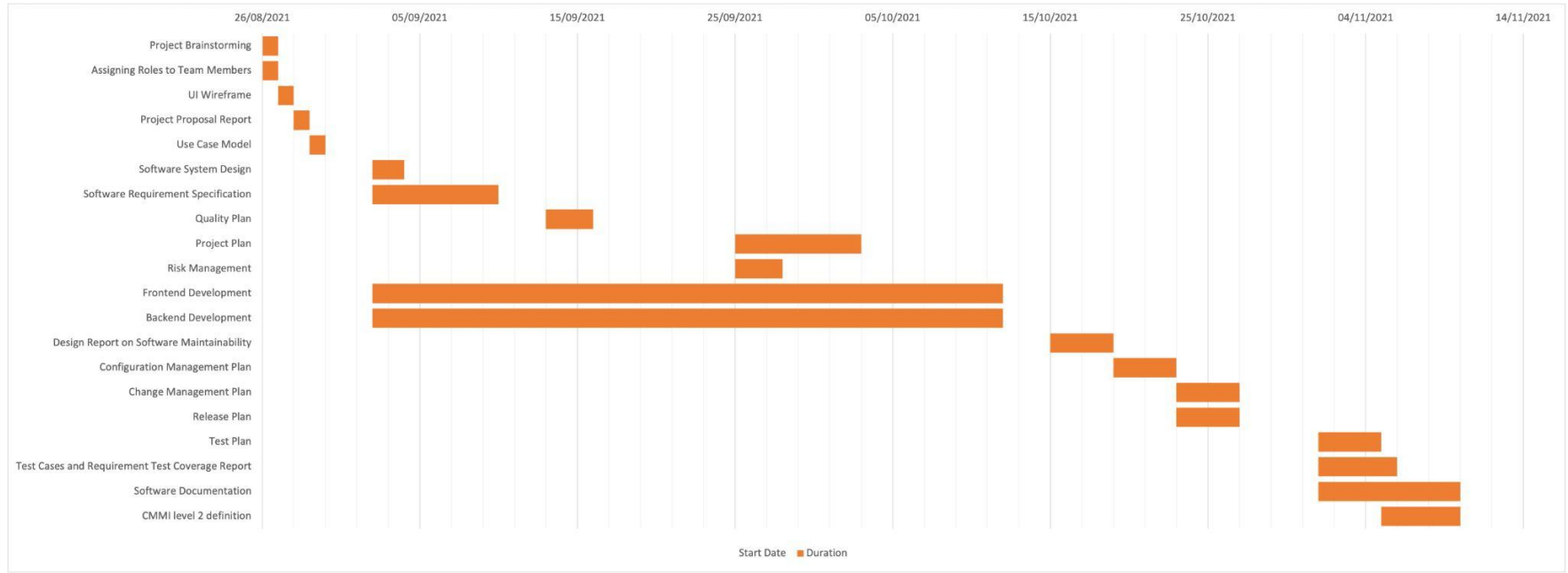
Efforts, Duration and Team Size Estimation

- Working days include 5 days in a week
- $\text{Effort} = \text{Size} / \text{Production Rate} = (6919.68 \text{ LOC}) / (62 \text{ LOC/PD}) = 111.61 \text{ PD}$
- $\text{Duration} = 3 \times (111.61)^{1/3} = 14.44 \text{ Days}$
- $\text{Initial schedule} = 14.44 \text{ Days} / 5 \text{ Days a week} = 2.89 \text{ Days}$
- $\text{Team Size} = \text{Effort} / \text{Duration} = 111.61 / 14.44 = 7.73 \text{ P} = 8 \text{ Persons}$
- Working hours include 8 hours in a working day.
- $\text{Total person-hours (PH)} = 111.61 \times 8 = 892.88 \text{ PH}$

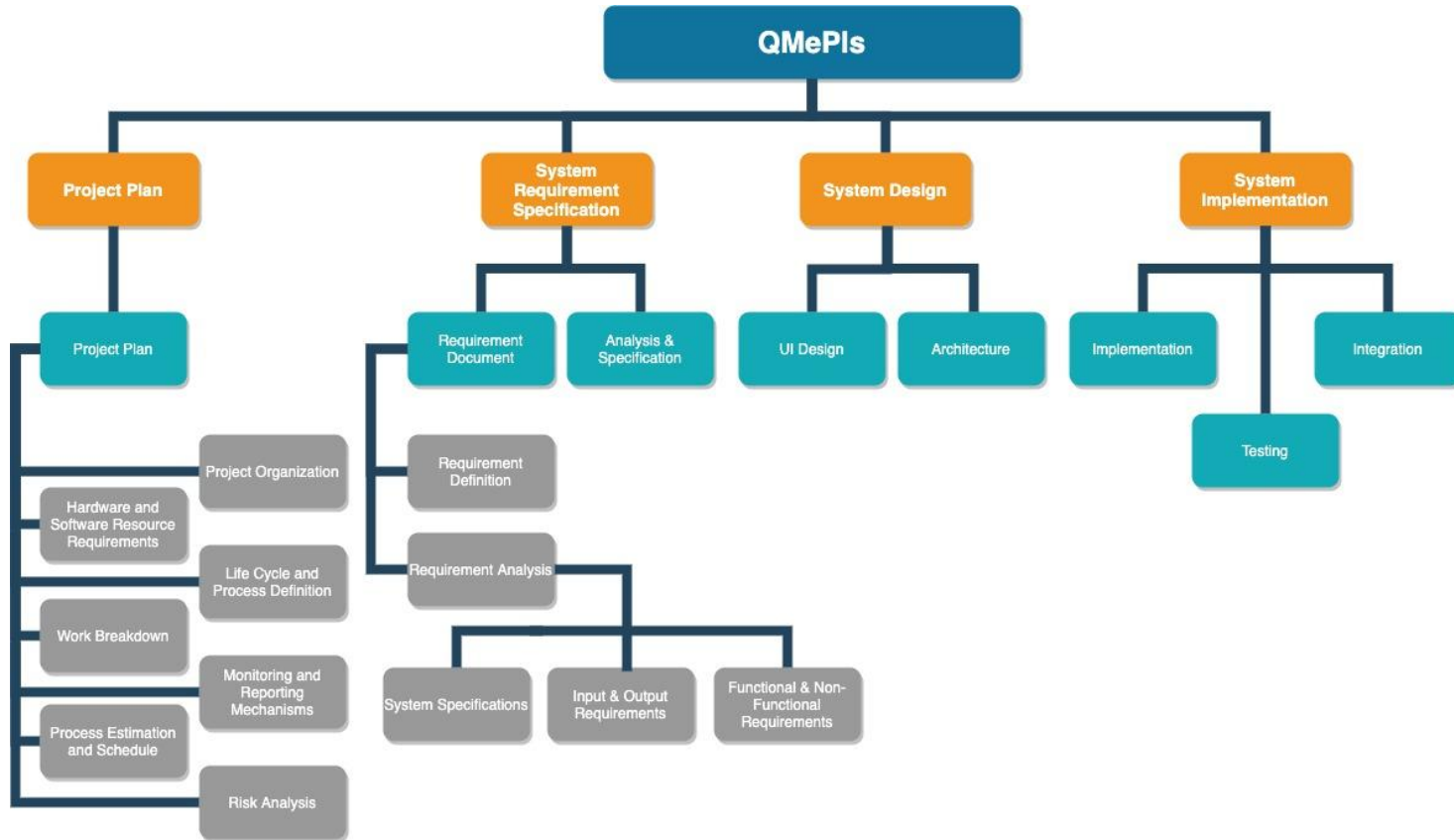
Project Schedule

1. **Activity Dependencies
and Schedule**
2. **Work Breakdown
Schedule**
3. Work Packages
4. Activity Dependencies
5. Work Package Details

Activity Dependencies and Schedule



Work Breakdown Schedule



Risk Management

Purpose of Risk Management

1. Risk Identification
2. Analysis of the risk to reward/satisfaction
3. Risk Response Planning
4. Monitoring of said risks

Risk Identification

- Technical - Data related issues
- Project Management - Project objectives
- Resource Management - Budget & Time
- Political - Legal Obligations
- Project Team - Communications & Leadership

Risk Analysis

- Qualitative Risk & Quantitative Risk
- Probability - Chance of Occurrence
- Impact - Cost, Schedule or Performance

I m p a c t	High			
	Mid			
	Low			
		Low	Mid	High
Probability				

Risk Analysis – Qualitative

Risk	Probability	Impact	Consequences
Fully loaded database	Medium	Medium	<ul style="list-style-type: none">- Slow fetching times- Inability to add new Patient Users/Clinics
Malicious attacks on the server	Medium	High	<ul style="list-style-type: none">- Disclosure of confidential information
Loss of server data	Low	High	<ul style="list-style-type: none">- Loss of confidential information- Loss of functionality of system
Lack of awareness of project objectives	Medium	Medium	<ul style="list-style-type: none">- Project does not meet business requirements- Miscommunication and confusion within the team

Risk Analysis – Quantitative

Risk	Numerical Risk Value (1-10)
Fully loaded database	5
Malicious attacks on the server	8
Loss of server data	9
Lack of awareness of project objectives	5

Risk Response Planning

- Avoid - Malicious Attacks, Fully Loaded Database
- Mitigate - Fully Loaded Database, Project awareness/objective changes
- Accept - Change in stakeholders direction/plans
- Transfer - Loss of Server Data

Risk Monitoring

- Generate a risk of high risk issues and have regular updates/reports as a component of the project status
- Draft a Standard Operating Procedure (SOP) document detailing steps to mitigate/avoid/transfer said risks
- Quarterly risk assessments to update and asses current and new potential risks.
- Notify managements on important changes to risks statuses as a component of the Executive Project Status Report

Thank You!