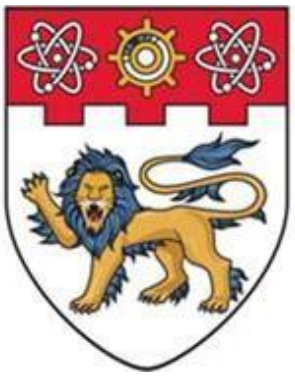


NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

# Release Plan for QMePls

## By Team Titans

**Date: 10 October 2021**


Seow Jing Hng Aloysius
Jacob Law Zhen
Jolene Tan
Soh Qian Yi
Samuel Lee Si En
Zeta Chua Hui Shi

Team Titans
School of Computer Science and Engineering

# Approvals

**Submitting Organization's Approving Authority:**

| | | | |
|---|---|---|---|
| | SCSE, NTU | | 6790 5786 |
| Signature | Printed Name | Date | Phone Number |

Nanyang Technological University

**Team Titans (QMePls)'s Approving Authority:**

| | | | |
|---|---|---|---|
|  | Jolene Tan | 27/10//2021 | 8321 0630 |
| Signature | Printed Name | Date | Phone Number |

Project Manager

# Revision History

| Version | Date | Organization/ Point of Contact | Description of Changes |
|---|---|---|---|
| 1.0 | 10/10/2021 | TeamTitans / Aloysius | Upload initial template |
| 1.1 | 11/10/2021 | Team Titans / Jolene | Introduction<br>Referenced Documents<br>Overview |
| 1.2 | 21/10/2021 | Team Titans / Jolene | Assumptions, Constraints, Risks<br>Release Approach |
| 1.3 | 25/10/2021 | Team Titans / Jolene | Glossary<br>Acronyms |
| 1.4 | 27/10/2021 | Team Titans / Qian Yi | Format Document |

# Table of Contents

# List of Figures



# List of Tables

# 1. Introduction

This release plan document applies to the QMePls application, In the release planning, the project manager and product owner created a long-term plan to deliver increments to the product and set the base by outlining the need, agenda and key deliverables for each release. The release plan will be updated after a new release is tested and verified to be working. The audiences of this release plan are the development team, test team, release team and the relevant stakeholders of the QMePls application. The following states the purpose of the release plan:

- **Road Map**: Create the release life cycle from the development to official release and future updates of the application
- **Product Vision**: Extend visibility past a single sprint such that executives can make an informed budget and scheduled decision
- **Communication**: Allows stakeholders and project development team to understand the complete set of functionalities delivered in the product
- **Documentation**: The release plan is used to record all related information for the application in the release life cycle

# 2. Referenced Documents

**Table 1: Referenced Documents**

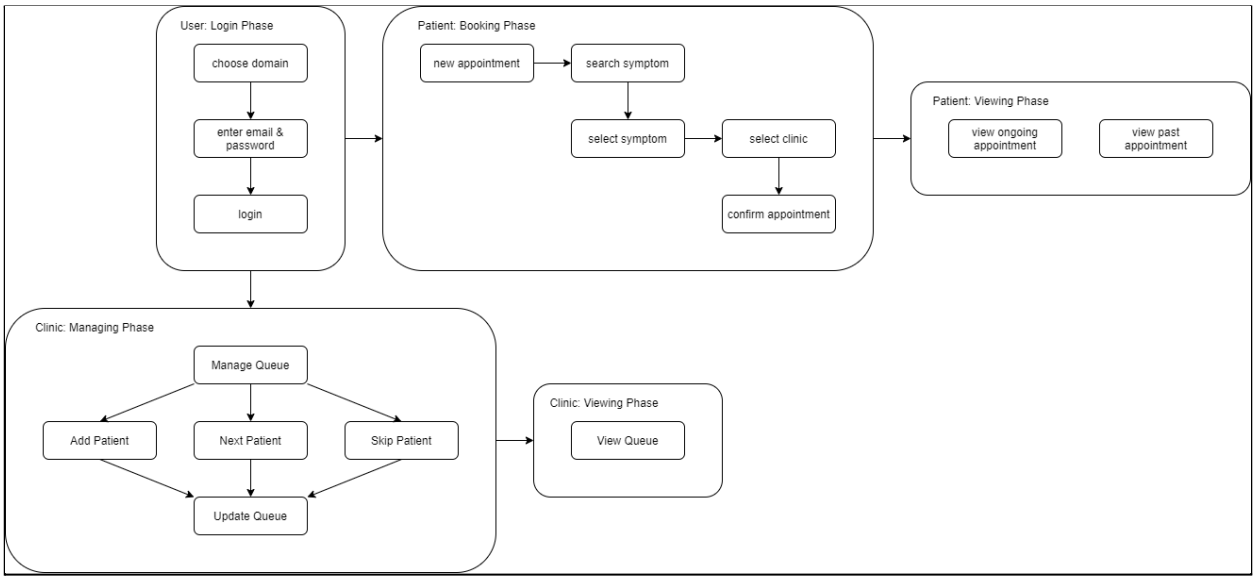
| Document Name | Document Number | Issuance Date |
|---|---|---|
| Project Proposal | Prerequisite Document (PD1) | 26/8/2021 |
| Software Quality Assurance Plan | Related Document (RD1) | 9/9/2021 |
| System Requirement Specification | Relevant Technical Documentation (RT1) | 9/9/2021 |
| Project Plan | Prerequisite Document (PD2) | 23/9/2021 |
| Risk Management Plan | Related and Companion Document (RD2) | 23/9/2021 |

The following section summarizes the relationship of this document with the relevant referenced documents:

- **Project Proposal**: Provide an overview of the project, its objectives and proposed project deliverables, and scopes are defined
- **Software Quality Assurance Plan**: Quality test on the different components & integrating them together before we can release the application to end users
- **Software Requirement Specification**: Acquire requirements and analyse them. Provide features or functions that need to be implemented
- **Project Plan**: Provide us with ideas of the lifecycle model and schedule of the project
- **Risk Management Plan**: The risks associated with the release of the system identified and mitigation strategy are documented and managed in accordance with this referenced document

# 3. Overview

**Figure 1: Overview of QMePls Application**




QMePls is an application aiming to reduce congestion, boost efficiency and provide convenience for patients or people taking their Pre Event Tests (PET) at nearby clinics. Clinic goers are able to check the queue statuses of nearby clinics, add their symptoms to a report card and join the current queue, thus only needing to turn up at the clinic when their turn is nearing.

Furthermore, clinics are able to control this queue to stop taking patients, allow patients to book or even push late patients down the queue line.

QMePls application has 5 phases namely User: Login Phase, Clinic: Managing Phase, Clinic: Viewing Phase, Patient: Booking Phase and Patient: Viewing Phase. The following section explains each phase:

- **User: Login Phase** - Users can login to the app by choosing either the patient or clinic domain and authenticating with their email and password thereafter.
- **Clinic: Managing Phase** - Clinic staff can update the current clinic queue by adding a new walk-in patient, removing a patient that has completed his/her appointment, skipping a patient that has missed his/her turn in the queue.
- **Clinic: Viewing Phase** - Clinic staff can view the current queue status which shows the names of the patients and their positions in the queue.
- **Patient: Booking Phase** - Patient can start a new appointment booking process by adding a list of symptoms they are currently feeling and choosing a clinic they want to visit. They will be added to that clinic queue after confirmation.
- **Patient: Viewing Phase** - Patient can view his/her current position in the queue for an ongoing appointment. Patients can also view the history of his/her past appointments.

# 4. Assumptions, Constraints, Risks

## 4.1 Assumptions

It is assumed that the team is not drastically affected by unforeseeable external circumstances that may impact the release plan.
It is also assumed that the team receives indefinite support for the following technologies:

- SDK License from Google
- Android Open Source Project (AOSP)

## 4.2 Constraints

The main constraint of the project is the short development time. Developers are required to complete the project proposal and receive approval before starting on the project plan and requirement specification document for the application. Moreover, developers are expected to complete release version 1 within approximately 6 weeks which places stress on the team's schedule.

## 4.3 Risks

**Table 2: Project Risk**

| Risk | Description |
|---|---|
| Technical Risk | - Overloaded database<br>- External attacks on the server<br>- Loss of data<br>- Inadequacy of technical personnel |
| Project Management Risk | - Lack of risk awareness<br>- Mid-project changes in objectives, requiring major design rework<br>- Time required to develop the project is underestimated<br>- Rate of repair of defects is overestimated |
| Resource Management Risk | - Budget is insufficient to complete the project<br>- Inadequate resources due to reduction/overloading |
| Political Risk | - Legal and bureaucratic obstructions |
| Project team Risk | - Absence of key figures during |

| | |
|---|---|
| | critical time periods<br>- Poor team communications with upper management and/or within the team |

**Table 3: Risk Impacts and Mitigation Strategy**

| **Risk** | | **Impact(s)** | **Mitigation Strategy** |
|---|---|---|---|
| Technical Risk | Overloaded Database | - Slow fetching times<br>- Loss of Customers | - Conduct server load prediction when requisitioning more server space during expected high load situations. This will be done with historical data.<br>- Should there be sudden spikes in user activity, more server space can be requisitioned as needed |
| | External attacks on server | - Disclosure of confidential information | - Should there be an attack on the server the following steps will be taken<br>- QMePls' servers will be shut down<br>- All users will be logged out<br>- All current sessions will be flushed<br>- Finally, there will be a popup when users try to open the QMePls app on their mobile devices, conveying the message that QMePls will be down till further notice |

| | | | |
|---|---|---|---|
| | Loss of data | - Productivity disruption<br>- Disclosure of confidential information<br>- Loss of | - Ensure there is always 3 back-up copies of all data, 1 copy will be stored off-site<br>- Ensure legacy |

| | | | |
|---|---|---|---|
| | | developmental time to make up for lost work | versions of the product are kept<br>- Ensure all legacy data is kept for at least a year |
| | Inadequacy of technical personnel | - Poor implementation leading to requiring more man-hours<br>- Mistakes made results in downtime | - Ensure technical personnel are properly trained and their skills are up to date<br>- Should the need arise, they will be sent for additional course |
| Project Management Risk | Lack of risk awareness | - Overlooking risks<br>- Slow response to incidents | - Actively identify major risks<br>- Communicate these risks to the team and upper management, ensure that everyone involved understands the risks and its consequences |

| | | | |
|---|---|---|---|
| | Mid-project changes in objectives, requiring major design rework | - Waste of time and effort<br>- Loss of developmental time | - Minimized requirements by reviewing and validating requirements<br>- Minimize impact by using change management logs<br>- Access requirements change impact through the derivation of traceability information and communicate the implications of said changes<br>- Conduct Independent Verification and Validation |

| | | | |
|---|---|---|---|
| | Time required to develop the project is underestimated | - Unexpected costs in increasing project timeline/outsourcing development of components/increasing team size | - Add buffers and constantly cross-review of task requirements |
| | Rate of repair of defects is overestimated | - Extra development time required<br>- May affect project time line | - Replace defective components with reliable components<br>- Ensure there are hot and cold spares for essential equipment<br>- Defect management must be followed |
| Resource Management Risk | Budget is insufficient to complete the project | - Project may be abandoned<br>- Project may have to scale down<br>- Additional resources will have to be obtained | - Prepare a briefing document for senior management and conduct a briefing to ensure sufficient budget allocation |
| | Inadequate resources due to reduction/overloading | - Unexpected costs in increasing project timeline/outsourcing development of components or increasing team size | - Prepare a briefing document for senior management and conduct a briefing to ensure sufficient resource allocation |
| Political Risk | Legal and bureaucratic obstructions | - Unexpected costs in increasing project timeline/outsourcing development of components | - Ensure all policies are in line with the industry<br>- Prepare to change the policies should there be changes in government regulations |

| | | | |
|---|---|---|---|
| | | or increasing team size to meet government regulations<br>- Potential scrapping of project due to lack of economic viability | |
| Project team Risk | Absences of key figures during critical time periods | - Lack of expertise which causes slowdown/halting of development | - Plan the team to ensure overlap of work roles between team members to allow work to continue should any member be incapacitated<br>- Ensure proper chain of command to prevent misunderstanding in event of a personnel is unavailable |
| | Poor team communications with upper management and/or within team | - Low team motivation<br>- Ill will within the team or towards management | - Recruit an experienced and capable project manager<br>- Ensure team is involved with decisions that affect the project |

# 5. Release Approach

## 5.1 Rationale

The release approach ensures that the complete application will be delivered to the end users with the defined requirements. Before the application is released to the stakeholders and end users, testing will be done. The design, implementation and testing phase will strictly adhere to the planned schedule for the project to be carried out according to plan.

## 5.2 Release Strategy

The aim of the release strategy is to design a release plan and decide on the release dates for specific functionalities of the QMePls application. Critical and defining functionalities such as posting of items to donate and item request should be released in the first release version. The subsequent versions should provide functionalities of lower priority aimed at enhancing users' experience such as filtering categories. Before each release, all functionalities will be tested thoroughly by the test team.

### 5.2.1 Release Content

**Table 4: Release Version**

| Release Version | Functionalities |
|---|---|
| 1.0 | Patient:<br>● Login/User Authentication<br>● Book Appointment System<br>● Queue Management System<br>● Clinic Map<br><br>Clinic:<br>● Login/User Authentication<br>● Queue Management System |
| 2.0 | Patient:<br>● Symptom Selector<br>● Notification System |

### 5.2.2 Release Schedule

**Figure 2: Release Schedule**

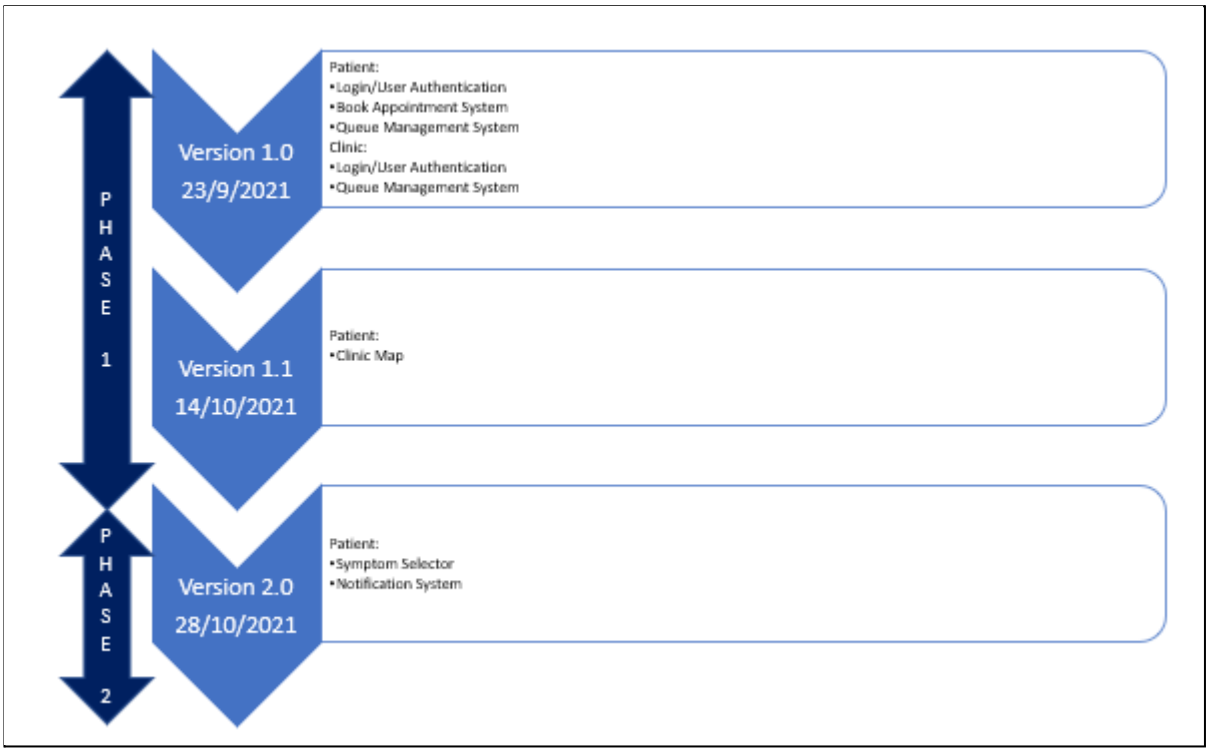



**Table 5: Release Schedule**

| Release Version | Description | Start Date | End Date |
|---|---|---|---|
| 1.0 | Phase 1 | 10/9/2021 | 22/9/2021 |
| 1.1 | Phase 1 | 23/9/2021 | 13/10/2021 |
| 2.0 | Phase 2 | 14/10/2021 | 27/10/2021 |

### 5.2.3 Release Impacts

Version 1.0 serves as the baseline of the QMePls application which is crucial to defining the functionalities of the system as they are the features that will be used most frequently and defines the purpose of the application. Release Version 1.0 is the official functional interface that will be released to Singaporeans. With Release Version 2.0, the application should contain all predefined and essential functionalities. Subsequent releases, starting from Release Version 2.0, will add features targeted at improving the coverage of functionalities and improving users' experience. These features are of lower priority, however, offer a complete and user-friendly application.

### 5.2.4 Release Notification

The version control platform for the QMePls application is GitHub. When implementation is done on a set of features, the code will be uploaded to the version control system. The test team will be informed about the release of the latest version by the lead developer to conduct testing and to provide feedback to the lead developer. The lead developer will then brief the frontend and backend developers on the respective changes or improvements to be made.
Relevant documentations on the release will be uploaded and all team members will be notified via email. Each release version must be tested, and tests have to pass at least 5 days before the official release. Details regarding the new release such as version number, features added or modified, release date and test results will be disseminated to all team members.
After the official release, the end users will receive the new version release notification containing release version number, release date and new or updated features.

# 6. Glossary

**Table 6: Glossary**

| Name | Description |
|---|---|
| QMePls | Name of our Android mobile application |
| GitHub | GitHub is an internet hosting provider for software development and version control using Git |
| Patient | Users of the patient side application interface, which requires a patient account. |
| Clinic | Users of the clinic side application interface, which requires an admin account. |

# 7. Acronyms

**Table 7: Acronyms**

| Acronym | Description |
|---|---|
| SDK | Software Development Kit |