Planit Release Plan

Team Members	Role
Ryan Tan Jinn En	Project Manager
Neo Yong Tai	Development Lead
Lee Yu Sheng Daniel	Back-End Developer
Mamuduri Paulani	Front-End Developer
Frankie Ye Htet Myat	Release Engineer / Manager
Chen Xueyao	QA Manager
Kundu Koushani	QA Engineer

Name of Design Team: **Team Syan**
School of Computer Science and Engineering, Nanyang Technological University

REVISION HISTORY

Version	Date	Implemented By	Description of Changes
1.0	21/10/21	Chen Xueyao	First draft

Table of Contents

1. INTRODUCTION

This document lays out the release plan for Planit, which is a web application developed by the Planit team as a solution for beginner gardeners in Singapore. It aims to describe the release strategy for Planit and provides full release information for the Planit application for all past, current and future releases.

This document will be updated regularly and used to keep track of all releases of the application. It ensures that the product, Planit web application, is always moving in the right direction. It helps the Planit team to decide on how much must be developed and how long it will take before Planit's next release. The release plan also presents a roadmap of how the team intends to achieve the project vision within the scope of project objectives and constraints.

2. REFERENCED DOCUMENTS

Document Name	**Document No.**	**Issuance Date**
Planit Project Proposal	1.0	Oct 10, 2021
Planit Project Plan	1.0	Oct 10, 2021
Planit System Requirement Specification	1.0	Oct 10, 2021

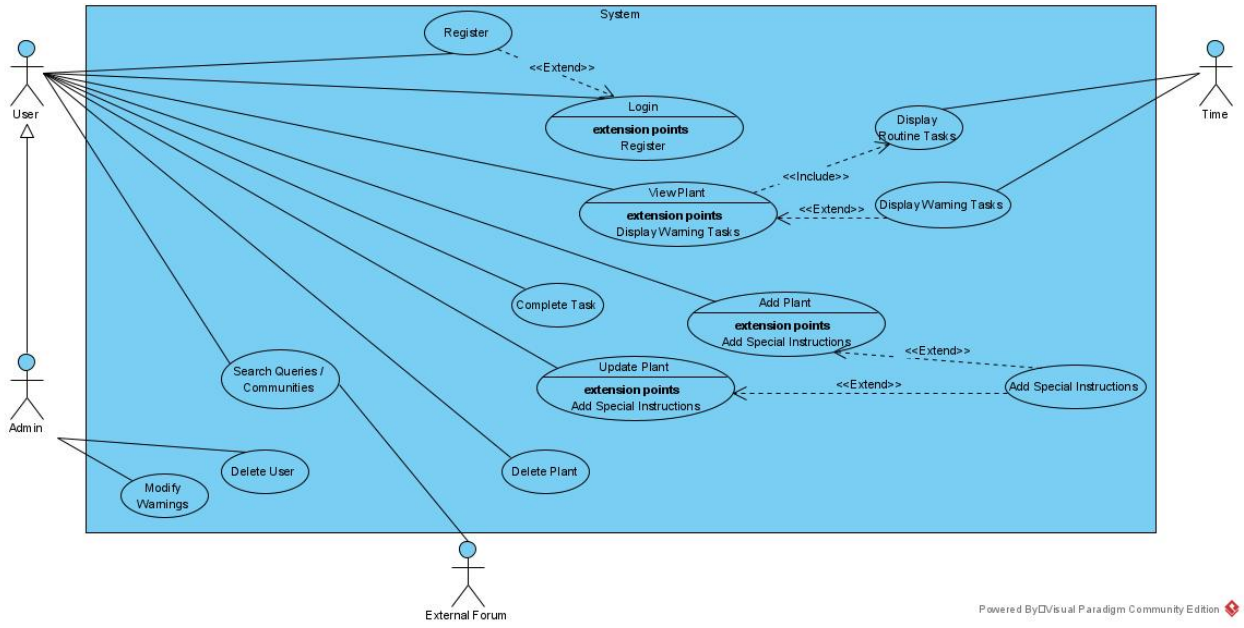
3. OVERVIEW

Island nations such as Singapore have always relied on the importation of food to meet the nation's needs. With the recent disruption of trade routes due the Covid-19 pandemic, the need to find new avenues to counteract this problem has arisen.

One of the recent approaches taken by the Singapore government was to encourage the growth of local gardening communities around the country. However, not all residents have experience in gardening. The Planit application aims to provide a go-to-guide in the form of a web application that allows users to keep track of plant growth and assist them in their gardening journey. Planit aims to alleviate some of the tedium that comes from garden maintenance, providing users with more free time with gardening remaining a hobby.

Planit was developed using the Express web application framework for the frontend and MongoDB for the database. The version 0.0.0 was developed for internal testing and version 1.0.0 has been planned to be the first public release of Planit.

The use-case diagram for the system is as below:



4. ASSUMPTIONS, CONSTRAINTS AND RISKS

4.1. ASSUMPTIONS

Development and Maintenance

1. The developers are skilled and able to meet expectations.
2. Users are willing and able to provide timely feedback in the case of a bug.

Schedule

1. There is no significant change to the development and release timeline.
2. All tasks are completed by their specified deadlines.
3. Buffer time has been provided in case of unforeseen circumstances that will affect the release of the product.

Budget and Financing

1. There are no sudden budget cuts or hidden costs.
2. The planned budget and resource usage have been accurately estimated.

4.2. CONSTRAINTS

1. The team size is relatively small and consists only of 7 members.
2. The schedule provided for the full development cycle of Planit is tight.
3. Planit is reliant on MongoDB for the storage of essential data. Any security breach or server issues faced by MongoDB will impact Planit.

Due to the first 2 constraints, there are limitations for the development of Planit. Budget and manpower is lacking to develop Planit for other platforms such as Android or iOS. There is also insufficient budget and time to provide extensive testing for the application.

4.3. RISKS

A detailed description of the risks involved in the development and release of Planit can be found in the Risk Management Plan. A summary of the risks, their severity and the probability of occurrence is provided below:

Impact				
	High	1. Data Theft 2. Quarantined Team Members	1. Scraper failure 2. Requirement Changes 3. Sick Developer	Server Crash
	Medium			
	Low	1. Overworked Developers	1. Underestimated deadlines	
		Low	**Medium**	**High**
		Probability		

5. RELEASE APPROACH

5.1. RATIONALE

Planit adopts a Waterfall model as the release approach, whereby project activities are broken down into linear sequential phases dependent on the deliverables of the previous phase. As Planit is a relatively small project with a low budget, low manpower and a tight schedule, a rigorous and highly structured approach to its development, with a clearly defined start and end is preferred.

The Waterfall model provides all of the above. In addition, it is simple and easy to understand, use and manage, thus preventing any wastage of time caused by a lack of understanding in the project schedule. The clearly defined stages and tasks provide the Planit team with a rigorous project schedule and clear responsibilities for each of the team members. This ensures efficient development of Planit and full utilization of manpower.

5.2. RELEASE STRATEGY

The release strategy the Planit team has adopted is the **phased function rollout**. This involves the incremental development, implementation and release of separate modules and functions in the Planit application to be combined over time. Similar to the Waterfall development model, the implementation of different modules and functions in the Planit application are separated into different phases. This ensures that all potential implementation issues are fixed before the release of the next version and prevents developers from having to deal with multiple implementation issues at the same time.

The version control software for Planit is Git. For each release, developers will take the following steps:

1. Create release branch
2. Checkout release branch
3. Build and test release branch
4. Create release distribution file
5. Test distribution contents
6. Tag release
7. Hand-off distribution file to QA

They will also ensure that the executable program delivered to the test team or to customers is complete, repeatable, informative, schedulable and portable.

5.2.1. RELEASE CONTENT

Based on Planit's System Requirements Specification and Use Cases, the following functionalities are to be expected of the Planit application:

Use Case	Functionality	Description
1	Register User	User registers for a Planit account.
2	User Login	User logs into their existing Planit account.
3	Delete User	Admin deletes an existing User from the system.
4	View Plant	User accesses and views information related to their plant's growth.
5	Add Plant	User adds a new plant.
6	Modify Plant	User modifies an existing plant.
7	Delete Plant	User deletes an existing plant.
8	Add Special Instructions	Application provides users with plant growing instructions specific to their plant's species.
9	Display Warning Tasks	System displays warning tasks for each plant.
10	Modify Warnings	Admin modifies the warning task message(s) sent to users.
11	Display Routine Tasks	System displays outstanding tasks to the user.
12	Complete Task	User marks their outstanding tasks as completed.
13	Search queries	User enters a search query and the system displays external search results to the user. User will be redirected to an external website upon clicking any link.

The following table identifies each specific release and the functionalities in each release together with the rationale behind the releases.

Release Version	System Functionalities	Description/Rationale
0.0.0	Use Cases 1-7	**Released for internal testing only.** This release allows developers to test the essential functions of the Planit application to ensure that there are no urgent or high-priority bugs that affect the fundamental functionalities of Planit. Once testing is completed, further addition of secondary functions can take place.
0.0.1	Use Cases 1-13	**Released for internal testing only.** All secondary functions are added. This release allows developers to test all functions of the Planit application to ensure that it is ready for release to users.
1.0.0	Use Cases 1-13	**Released for all users** This release includes all functionalities of Planit. With this release, Planit will be available to all public users. During this release, developers will collect usage information and user feedback and work to improve future releases.
1.0.X	Use Cases 1-13	**Released for all users** These releases include all functionalities of Planit. The releases are revisions to the 1.0.0 after bug fixes or small quality-of-life changes (e.g. change in font, slight adjustments in UI, etc).

Beyond version 1.0.X, minor releases (1.X.X) will similarly go through the cycle of internal testing, public release and revisions.

5.2.2. RELEASE SCHEDULE

To ensure good user experience, Planit will adopt regular monthly revisions to ensure that bugs and other user feedback are addressed in a timely manner.

Release Version	Date of Release
0.0.0	15 Oct 21
0.0.1	22 Oct 21
1.0.0	25 Oct 21
1.0.X	Recurring monthly revisions and patches

5.2.3. RELEASE IMPACTS

Each release will affect various components of Planit. The impacts can be classified under: business impacts, business process impacts, system/interface impact and impact to end users.

Release	System Impact	Business/Business Process Impact	Impact to End Users	Benefits/ Objectives/Goals
0.0.0	Modification of source code in the main branch according to internal test results.	Essential functions of Planit are now ready.	-	Essential functionalities of Planit are ready for release.
0.0.1	Modification of source code in the main branch according to internal test results.	Non-essential functions of Planit are now ready.	-	Non-essential functions of Planit are ready for release. Planit is ready for public release.
1.0.0	Main branch is now ready for new changes	Product rolled out to users	Product rolled out to users	First major release to users. User feedback and usage statistics can now be collected.
1.0.X	Main branch updated	Improve user experience through bug fixes	Better user experience now that bugs are fixed	Remove bugs and introduce minor updates and changes to improve user experience

5.2.4. RELEASE NOTIFICATION

When a new release is prepared, the various stakeholders and users of Planit must be informed. This ensures that stakeholders are up to date with the project's progress and users are informed of the improvements made. The table below lists the release notification methods:

Stakeholder	**Method of Notification**	**Information Required**	**Timeframe**
Investors	Email notification	A summary and recap of all changes made in the new release to ensure investors are up-to-date on the project progress. Details such as budget, user impressions and application ratings should be provided when asked.	Immediately after the release is approved.
Users	Email notification Website notification bar	A quick summary of all the changes available in the new release, the date and time of the release as well as how to update to the newest version.	Email should be sent 1 day prior to the release of the new version. The website notification bar should be added 3 days prior to the release of the new version.
Planit Team	Online conference	A summary and recap of all changes made in the new release to ensure the entire team is up-to-date on the adjustments that were made or not made.	Immediately after the release is approved

6. GLOSSARY

Express	A Node.js web application framework that provides a robust set of features for web and mobile applications
MongoDB	A document-oriented database program
Release	A version of the application that is made available for internal testing or release to users
Use Case	A written description of the tasks that users can complete through the Planit application.

7. APPENDICES

Project Proposal: https://github.com/neoyongtai/3002_project/blob/Deliverables/Planit%20Proposal.pdf/

Risk Management Plan:
https://github.com/neoyongtai/3002_project/blob/Deliverables/Risk_Management_Plan_Planit.pdf/

System Requirements Specification:
https://github.com/neoyongtai/3002_project/blob/Deliverables/System%20Requirement%20Specification%20.docx.pdf/

Use Cases:
https://github.com/neoyongtai/3002_project/blob/Deliverables/Planit%20Use%20Case%20Descriptions.pdf