



**NANYANG
TECHNOLOGICAL
UNIVERSITY**

SINGAPORE

Release Plan for TimeWise

Team 0
Lab group : SSP7

Group members	Roles
Mantri Raghav	Project Manager
Dwivedee Lakshyajeet	Development Lead
Harding James	Back-End Developer
Alex Bernini	Front-End Developer
Mittal Madhav	Release Manager
Xue Xueting	QA Engineer
Koh Hui Ling	QA Engineer
Lek Jie Ling	QA Manager

Approvals

Submitting Organization's Approving Authority:



Raghav Mantri

10/4/2020

Project Manager

Signature

Printed Name

Date

Title Name

Version History

Version #	Implemented By	Revision Date	Approved By	Approval Date	Reason
1	Mantri Raghav	7/4/2020	Dwivedee Lakshyajeet	7/4/2020	Completed Introduction, Overview, Assumptions, Constraints, Risks
1.2	Mantri Raghav	8/4/2020	Dwivedee Lakshyajeet	8/4/2020	Completed Release content, rationale, strategy and timeline
1.3	Mittal Madhav	8/4/2020	Dwivedee Lakshyajeet	8/4/2020	Completed the Risks, Glossary, Acronyms
1.4	Mantri Raghav	10/4/2020	Dwivedee Lakshyajeet	10/4/2020	Completed all other sections and reviewed document

Table of Contents

Introduction	5
Referenced Documents	5
Overview	5
Assumptions, Constraints, Risks	6
Assumptions	6
Constraints	6
Risks	7
Release Approach	9
Rationale	9
Release Strategy	9
Release Strategy - Release Content	10
Release Strategy - Release Schedule	10
Release Strategy - Release Impacts	11
Release Strategy - Release Notification	11
Glossary	12
Acronyms	12

Introduction

The Release Plan is a detailed specification of the features of TimeWise and the upcoming release versions and their date of availability. The aim of the release plan is to assist in determining the project priorities and to follow the progress and developments made. This plan will allow the team to identify criteria for the release versions based on urgency and significance, while at the same time helping them to plan their own continuous integration throughout the development life cycle.

Referenced Documents

Document Name	Issuance Date
TimeWise Project Proposal	7/2/2020
TimeWise Project Plan	9/3/2020
TimeWise System Requirement Specification	20/2/2020
TimeWise Quality Plan	20/2/2020

Overview

Productivity is crucial to college students. Organizing tasks and keeping oneself up-to-date with all the work and studies can help a student get good grades and make life easy and organised in college. Unfortunately, the current offering in the app stores fail to cater to the particular demands of the students.

TimeWise will be an application developed to meet the needs of students in NTU. The mobile application will allow students to automatically import the courses they have registered for the current semester through NTU's STARS system. This application will then show the user their classes for the next 7 days in a list format. The user can also add their own tasks and will be recommended 5 time slots in which they should add the task, keeping in mind their classes and current tasks.

Current solutions do not allow students to automatically import their weekly timetable. This forces students to manually enter their classes into their calendar or app. This is the problem which TimeWise seeks to address.

Keeping in mind software maintainability as well as ease of refactoring and programming, the application will be developed using the React Native framework which is compatible for both iOS and Android.

Assumptions, Constraints, Risks

Assumptions

Manpower & Team Size

1. There is no change in the number of team members required to complete the project and they are always present for meeting sessions.
2. All the stakeholders are always present for meetings and updates as well as provide constructive feedback.
3. Developers are always focused on the allocated task and knowledgeable about the necessary software development skills..
4. Developers are experienced enough to build the application.
5. The rest of the team are well aware of assigned tasks and would fulfil them to the best of their abilities.

Time

1. The time frame for the project is confirmed and finalized in the TimeWise Project Plan.
2. All tasks are completed on time.

Budget & Funding

1. The funding and materials required for the project will always be on time for the project tasks and fully accounted for.
2. There are no sudden budget cuts or additional hidden costs.

Development

1. There are no significant technological changes to be made to the development of the project as the stakeholders are very specific and meticulous about the necessary requirements.
2. The project scope does not change.

Constraints

Manpower & Team Size

1. The team size is small as it consists of only 8 team members.

Time

1. The entire project must be done by the expected project deadline of 16 April 2019 from the start date of 28 January 2018.

Budget & Funding

1. The budget given for this project is limited to \$83,000.00 and the project must be completed within this budget.
2. No additional resources will be provided hence resource management must be effectively managed.

Development

1. The project will be limited to the software development knowledge that the team has and how an application like TimeWise should be managed.

Risks

A detailed risk plan can be found in the 'Risk Management Plan' of the TIMEWISE documentation. Below is the risk management log to identify and manage risks accordingly.

	Type	Risk Description	Probability	Impact	Significance (Out of 10)	Strategies to Counter
1	People	Overworked Developers Sick / Quarantined Developers	Low	Low May lead to additional backlog and bad code because of reduced efficiency / morale of coders.	5	Mitigate Temporary replacements for the coders will be found, and the original coders will be given regular updates about the progress
2	Technology	Changes in NTU Website	Medium	High Leads to an obsolete backend	7	Mitigate Need to spend additional resources to redesign backend
3		Data Theft	Low	High Leads to loss of	6	Mitigate Client trust will be rebuilt using

				valuable user data, and this user trust		the help of professionals, and additional money will be invested for a higher security database.
4		Server Crashes	High	Low Due to excessive workload on servers, overheating and other reasons, there may be server crashes which may also lead to loss of user data.	6	Mitigate To avoid server crashes, servers capacity may be increased to ensure they are not overloaded, and the server farms must be ensured to be kept under right conditions to avoid problems like overheating
5	Requirement Changes	Complaint from users	High	High Leads to unhappy users, who may eventually switch over to another app	10	Accept The clientele complaints will be listened and catered to to avoid dissatisfied users
6		New competitors with better functionalities	High	High New competitors with better functionalities may lead to loss of users	10	Accept We must keep an eye out for new competitors and be ready to invest in adding new

						features for the app.
--	--	--	--	--	--	-----------------------

Release Approach

Rationale

We will adopt Agile Software Development (ASD) as our development lifecycle methodology. Agile methods of development are commonly described as “iterative” and “incremental” development. The main advantage of utilizing Agile Software Development lies in regular adaptation to changing circumstances.

Our release approach would also be in accordance with the SCRUM methodology in which we develop, test then release with approval from the relevant representatives.

Incremental improvements will be made to the system based on the feedback regarding the user's requirement and the delivery. Adjustments will also be made to ensure that any bugs and issues identified are resolved in the subsequent versions.

Release Strategy

The Release Strategy we employed is known as the Phased Function Rollout. It is a process where we will implement functionalities in separate modules that are part of the entire system independently until implementation of the system is complete. The incremental approach adopted allows the monitoring of the development progress and raise any issues that occur in the early implementation stages. Information learned from the early implementations of the system could be further applied to guide the rest of the process which results in fewer issues upon release.

A significant amount of time and resources have been invested to developing and implementing the core functionalities of TimeWise. In addition, we have predicted that this app would be widely employed by the majority of NTU students due to its versatility and ease of usage.

The Major Release includes the core and basic functionalities of the TimeWise which includes components like allowing users to import their timetable from STARS, add and remove tasks created from the coursework timetable and get top 5 recommended times.

For the Minor Releases, enhancement to the user experience and interactions of the system will be included.

Future Releases of the app will mainly consist of Minor Releases as well as Revisions.

Major Releases occur when there are changes to the STARS system by NTU or frameworks used which causes incompatibility issues.

Minor Releases involve significant bug fixes and minor improvements of the mobile application.

Revisions occur when there are little bugs or errors that require fixing.

1. Release Content

Based on the TimeWise Software Requirement Specification, the system provides use cases and the users' requirements listed below:

Use Cases

- a. The user must be able to login to the app using existing NTU account credentials
- b. The user must be able to import their respective school timetable after login
- c. The user must be able to add other tasks in the app
- d. The user must be able to remove tasks, be it custom or class tasks in the app
- e. The user must be able to change and update existing tasks in the app
- f. The user must be able to receive top 5 recommended times on task creation to select from as the time of a task
- g. The user must be able to receive a notification from the app upon time reached in the task

TimeWise Release Version 0.0.0 - Development Phase

The development of the TimeWise will include all the functionalities mentioned above except for g. which is "The user must be able to receive a notification from the app upon time reached in the task".

TimeWise Release Version 1.0.0 – Ready for Production / Official Release / Major Release

The official release of the TimeWise application. The feature g. which is "The user must be able to receive a notification from the app upon time reached in the task". Will also be included in this version. All required functionality based on the Use Cases defined by the developers has been tested and vetted by the QA Engineers and approved by the QA Manager. With this release, the users can use the application which allows us to gather valuable feedback and further improve on the features of TimeWise.

TimeWise Release Version 1.1.0 – 1.Y.Z – Continuous Maintenance and Improvements

The Minor Release / Patches of TimeWise will involve improving the current user interface. These improvements enhance the both user experience while using the system.

2. Release Schedule

The Major Release of TimeWise – Version 1.0.0 will be completed by 7th April 2020 as mentioned in the Project Plan Document, with the basic and core functionalities implemented. Any subsequent releases may take up to 1 month to complete. The subsequent releases will

focus on maintenance as well as implementing new functionalities based on the feedback gathered by the users.

3. Release Impacts

Every time there is a new release, the system impacts associated with the release and user processes that are to be modified will be identified. Any modifications to the system functionalities might affect the future development of the system, therefore it is required for project manager to oversee the release, as his or her responsibilities include monitoring the progress of the team members. The project manager has to be able to foresee any future consequences from each release.

4. Release Notification

When a release version has been generated, the respective stakeholders will be notified as stated below. Methods of how the stakeholder will be notified, what information will be included in each notification and prior to release, the timeframes for receipt of the information, will all be included.

Stakeholders	Method of Providing Notification	Information included in notification	Timeframes for receipt of notification
Users	Through system	What changes has been added	3 days prior to version release
Teammates	Emails	What changes are made, including the bugs to be fixed and also reasons why changes were made and user statistics.	Immediately once decision to implement the new release version

Glossary

	Term Name	Term Definition
1	Backend	The part of software that is typically responsible for storing and manipulating data.
2	Backlog	An accumulation of uncompleted work
3	Database	A structured set of data
4	Overheating	(here) when servers become physically too hot and start crashing
5	Server	A data server) is a software program/platform used to provide database services like storing, processing and securing data
6	Server Crash	A server crash is a serious computer failure that occurs when a software application stops functioning properly and exits
7	Server Farms	A collection of servers

Acronyms

	Acronym	Literal Translation
1	ASD	Agile Software Development
2	iOS	iPhone Operating System
3	NTU	Nanyang Technological University
4	QA	Quality Assurance
5	SRS	Software Requirement Specification