

Software Requirement Specification

**Agile Software Project
Management System**
by
Team Coca-Cola

Unit Convenor: Prof. Patrick Then
Supervisor: Ms. Fu Swee Tee
Client: Ms. Phang Min Hui



1.1 Introduction:

● Purpose

- The purpose of this document is to provide the software requirement specifications of the Agile Software Project Management System. The creation of the system includes various functionality improvements, requirements given by client and also non-functional requirements that supports the agile software development lifecycle.

● Scope

- The scope of the project is to improve the Agile Software Management System developed in Software Engineering Project A. 3 different roles will get different improvement on their functionalities and UI of the system. The improvements will be added to all user roles includes **live chat function**, **viewing members' progress and the project progress**, **improved new layout of the Calendar page**, **auto language translation**, **hide important information in URLs using base64 encryption** and **receive email when resetting account password**. The system will also improve features that are accessible by project manager and project members such as improving the **Document Generator** to give user to customize their own version of document and to save documents in database so that they are editable before generating the documents. Besides, users can click on the notification that will redirect to live chat function and have a sub-form to choose either Project/Board/Task in time tracker function. In Progress page, users can also click Project image instead of just the word to view **Project details**, **edit project details** and have an **improved GitHub page UI** to improve user experience while interacting with the system. On the other hand, in Member's page, users can send message to other users and **view user status as either online or offline**. Apart from that, users can upload their own profile picture, display them and change their personal information. Lastly, for Project Managers only, they will be able **make announcement regarding projects to other project members**.

● ***Definitions, Acronyms and Abbreviations***

Below are definitions of term used within the domain of the Agile Software Project Management System. Definitions are further are covered in the next section (Business Rules)

Terminology	Definition
Dependencies	The software needed so the system will work without problems
Responsive	A design technique to ensure the software scales through different screen dimension and resolution
Sprint	Sprint is one timeboxed iteration of a continuous development cycle. Within a Sprint, planned amount of work has to be completed by the team and made ready for review.
Product backlog items	the Product Backlog is an ordered list of everything that is known to be needed in the product and constantly changes to identify what the product needs to be appropriate
Progress Tracking	A feature for all users to record and track their progress throughout certain period of time based on their actions in the task management and generate a graph or report to have an overview of their work progress.
Scrum Master	Scrum Master takes on the administrative, coaching and leadership roles that make Scrum development possible. In our system, project manager appointed with be the Scrum Master

2. Overall Description:

2.1 System Perspective

Agile Software Management System is an improved project management system that allows project manager to assign tasks to each project members and all of the task management, progress tracking, codes tracking & updates are done within the system. After each sprint, a complete documentation can be generated to summarized the progress for each project member. Project manager will also keep track of his/her project members' progress on the

project and give comments if there are any bugs found or corrections to be made. The improved codes will then be pushed to the repositories which automatically updates and integrates into the demo system. Project manager and members will be able to communicate and discuss by using the comment feature on the tasks assigned. The task management concept will be somewhat similar to Trello board with “To do”, “Doing”, “Testing” and “Done”.

2.2 New System Features

- █ All user roles
- █ Project Manager & Project Member
- █ Project Manager

All user roles:

- ❖ Live chat function
- ❖ View members' progress and progress of projects in Progress page
- ❖ Change the layout of the Calendar page
- ❖ Auto language translation
- ❖ Hide important information from URL using base64 encryption
- ❖ Receive email when resetting account password

Project Manager & Project Member:

- ❖ Give user to customize their own version of document in Document Generator
- ❖ Save documents in database for the document generator so that they are editable before generating the documents
- ❖ Redirect to live chat function by clicking on the notification
- ❖ Create sub-form to choose Project/Board/Task for time tracker function
- ❖ Project image is also clickable instead of just the word in Project Page
- ❖ Edit project details
- ❖ Improve GitHub page UI
- ❖ Add send message function in another column under Member's page
- ❖ Add user status: online or offline in another column under Member's page

- ❖ Upload and Display profile picture in another column under Member's page
- ❖ Allow user to change their personal information

Project Manager

- ❖ Make announcement regarding projects (Project Manager)

2.3 User Classes and Characteristics

➤ Client (Ms. Phang Min Hui)

- Ms. Phang is our client for this semester, also working in Karuna Sarawak company. Throughout this whole semester, she will guide us to make the system more powerful, useful and user friendly.

➤ Admin

- The admin will be able view everything that had been done by the user through the dashboard. They are controlling the back end of the system and manages both project manager and project member ensuring that they are doing their job properly therefore having the ability to do progress tracking and generating weekly reports based on the repository updates.

➤ Project Manager

- The project manager that created the project will have access to create new board and add, edit, delete requirements of the project. The created tasks are assigned to different project members. They will be in charge of the progress tracking of the project member ensuring that they are doing their job properly.

➤ Project Member

- Project members will be assigned with tasks and their job is to finish their tasks within the expected due dates. They will use the board to update their status of the task assigned for project manager to keep track of their progress. After their task is done, they will perform bug testing to make sure that everything is working properly. The completion date will then be updated after their tasks had been completed.

2.4 Design and implementation Constraints

Language

The primary language used in Agile Software Project Management System will be English. It has been the most popular language and widely used all around the world. Community with different cultural backgrounds will most likely be using English to communicate before getting to know more about each other.

The User Interface Design

The user interface will be able to scale across different screen resolutions (responsive design). Important information will still remain on the screen when the browser is resized and navigation bar is shrunk into a hamburger.

Security

The system must be secure so that the project that users are working will be keep safe. Only project member will be able to mark off tasks and some features like appoint project manager is only restricted for the project admin. Other users that are not involved in the particular project will not be able to gain access to it, this is to prevent information leakage.

2.5 Assumptions

1. All users must have internet connection to access to the system.
2. All users must have basic knowledge about Agile software development so that they will be able to navigate throughout the system.
3. Admin are more experience individuals whom will be capable to have the rights to delete user if necessary.
4. Project manager also have the feature to remove project member from project if necessary.
5. When user is removed from the system, their personal information will be totally removed. If they would like to participate in the project, they will need to create a new account.

2.6 Operating Environment

The web application can be operated on any device that have a web browser support.

3. System Features:

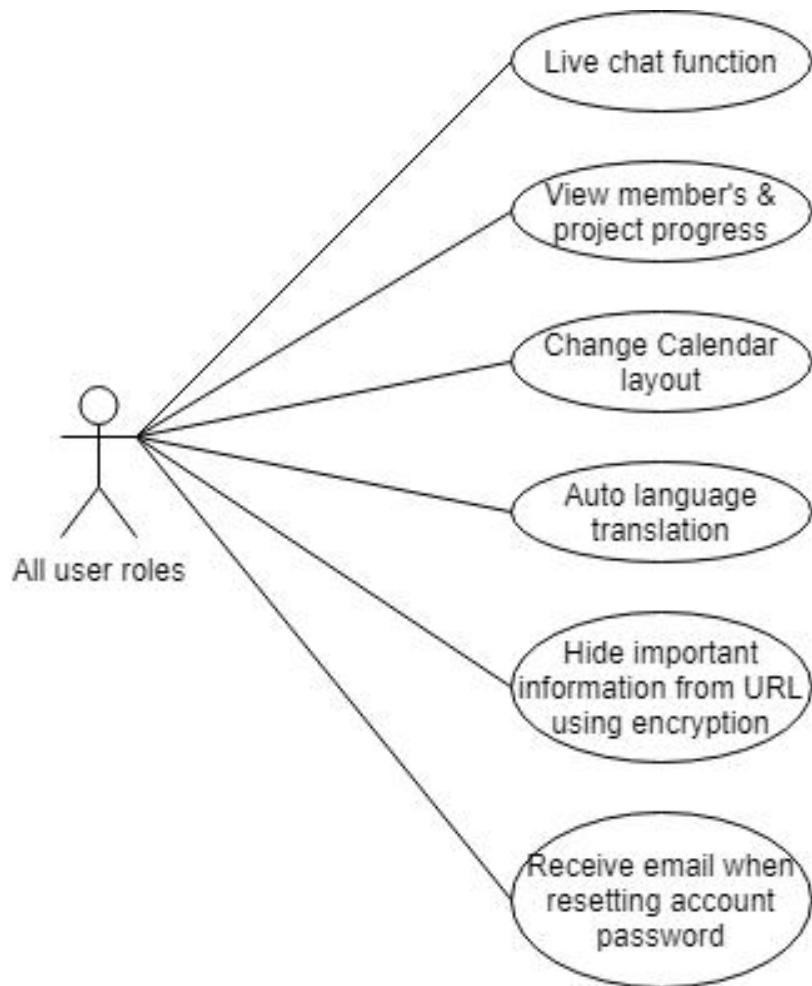
The functional requirements involved in the Agile Software Project Management System are listed down below:

- ❖ Live chat function (All user roles)
- ❖ View members' progress and progress of projects in Progress page (All user roles)
- ❖ Change the layout of the Calendar page (All user roles)
- ❖ Auto language translation (All user roles)
- ❖ Hide important information from URL using base64 encryption (All user roles)
- ❖ Receive email when resetting account password (All user roles)
- ❖ Give user to customize their own version of document in Document Generator (Project Manager, Project Member)
- ❖ Save documents in database for the document generator so that they are editable before generating the documents (Project Manager, Project Member)
- ❖ Redirect to live chat function by clicking on the notification (Project Manager, Project Member)
- ❖ Create sub-form to choose Project/Board/Task for time tracker function (Project Manager, Project Member)
- ❖ Project image is also clickable instead of just the word in Project Page (Project Manager, Project Member)
- ❖ Edit project details (Project Manager, Project Member)
- ❖ Improve GitHub page UI (Project Manager, Project Member)
- ❖ Add send message function in another column under Member's page (Project Manager, Project Member)
- ❖ Add user status: online or offline in another column under Member's page (Project Manager, Project Member)

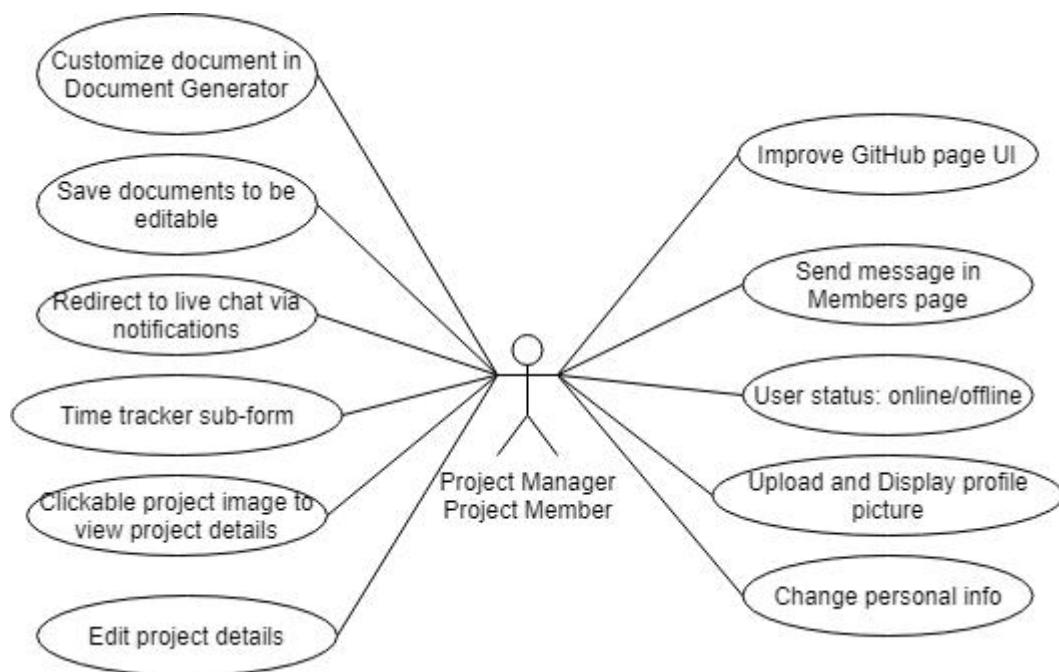
- ❖ Upload and Display profile picture in another column under Member's page (Project Manager, Project Member)
- ❖ Allow user to change their personal information (Project Manager, Project Member)
- ❖ Make announcement regarding projects (Project Manager)

3.1 Use Case Diagram

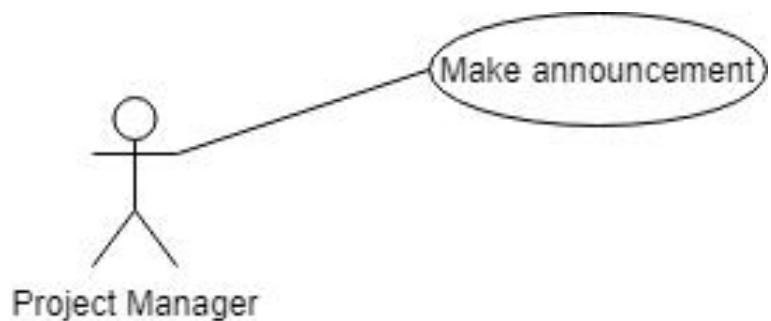
- All user role Use Case



- **Project Manager & Project Member Use Case**



- **Project Manager Use Case**



3.2 Use Cases

1. Use Case: Live chat function

Use Case Name	Live chat function
Actor(s)	All user roles
Purpose	To communicate with each other within the system
Precondition	The user had an existing account and is assigned to at least a project
Success Guarantee	The user successfully communicates with other users within the system
Sub-Tasks/Scenario	
1. User navigates to the live chat icon located at the top right corner of the system.	
2. User search for other users within the same project and sends a message.	
3. User successfully receives message from other users.	
Variant	
3a. The user is not in any of the project within the system.	
3a1. The system could not find any users to be contacted.	
3a2. Join a project and repeat step 2.	

2. Use Case: View members' progress and progress of projects in Progress page

Use Case Name	View members' progress and progress of projects in Progress page
Actor(s)	All user roles
Purpose	To view members' and project progress
Precondition	The user had worked on the project and is having progress data in the project
Success Guarantee	The user successfully view the progress of members and projects
Sub-Tasks/Scenario	
1. User navigates to the progress page of the system.	
2. User choose to view either members' progress or progress of the projects.	
Variant	
3a. The project does not have any progress/no members assigned to the project.	
3a1. The system displays error message "No progress found!".	
3a2. Start working on the project and repeat step 2.	

3. Use Case: Change calendar layout

Use Case Name	Change calendar layout
Actor(s)	All user roles
Purpose	To change the layout of the calendar page to a new one
Precondition	The user registered an account with the system
Success Guarantee	The user successfully viewed the new calendar layout

Sub-Tasks/Scenario
1. User navigates to the calendar page
2. User successfully viewed the new calendar layout
Variant
-

4. Use Case: Auto language translation

Use Case Name	Auto language translation
Actor(s)	All user roles
Purpose	To allow user to change the language of the system
Precondition	Having an internet connection
Success Guarantee	The language of the website successfully changed to the chosen one.
Sub-Tasks/Scenario	
1. Use navigate to the dropdown menu provided.	
2. User click on the dropdown menu to choose the languages provided.	
Variant	
-	

5. Use Case: Hide important information in URL using base64 encryption

Use Case Name	Hide important information in URL using base64 encryption
Actor(s)	All user roles
Purpose	To hide important information in URL to prevent information leakage from system
Precondition	User must create a project/board/task in the system
Success Guarantee	The URL link which shows id for project/board/task will be encrypted using base64 encryption and show random alphabets in the URL
Sub-Tasks/Scenario	
1. User click on the URL link and found there are board id or project id.	
2. User tried to change the project or board id in the URL to access other's person project information	
3. Error message pop out "Access Denied"	
Variant	
4a1. User does not have a project in the system.	
4a2. User are required to create a project or a board within a project.	
4a3. Repeat step 1.	

6. Use Case: Receive email when resetting account password

Use Case Name	Receive email when resetting account password
Actor(s)	All user roles
Purpose	To allow users to receive email when resetting account password
Precondition	The user had an existing account in the system
Success	The user successfully received email after resetting

Guarantee	account password
Sub-Tasks/Scenario	
1.	User clicks on “Forgot password” and reset their account password.
2.	User received email notifying that their password has been reset.
Variant	
-	

7. Use Case: Customize document in Document Generator

Use Case Name	Customize document in Document Generator
Actor(s)	Project Manager / Project Member
Purpose	To customize document of their choice
Precondition	The user had logged into the account in the system
Success Guarantee	The user successfully created their own custom documentation
Sub-Tasks/Scenario	
1.	User navigates to Document Generator.
2.	User choose to customize the document and successfully saves it inside their device.
Variant	
-	

8. Use Case: Save documents within the system

Use Case Name	Save documents within the system
Actor(s)	Project Manager / Project Member
Purpose	To save documents within the system to be editable in the next login
Precondition	The document and contents are created within the Document Generator
Success Guarantee	The user successfully saves the document in the system and accessible in the next login to edit
Sub-Tasks/Scenario	
1.	User navigates to the Document Generator and created a document.
2.	User clicks the “Save” button under the document.
3.	A popup message indicates that the document is successfully saved.
8.	The user navigates back to the saved document and continue editing after leaving the page.
Variant	
-	

9. Use Case: Redirect to live chat via notifications

Use Case Name	Redirect to live chat via notifications
Actor(s)	Project Manager / Project Member
Purpose	To redirect to live chat via notification
Precondition	A notification has been received in notification section
Success	The successfully redirected to the live chat via notification

Guarantee	
Sub-Tasks/Scenario	
1. User receives notification indicating that a new message is received.	
2. User clicks on the notification and the website redirect the user to the live chat.	
3. User replies the message in the live chat to the sender of the message.	
Variant	
-	

10. Use Case: Create sub-form to choose Project/Board/Task for time tracker function

Use Case Name	Create sub-form to choose Project/Board/Task for time tracker function
Actor(s)	Project Manager / Project Member
Purpose	To choose project/board/task from the sub-form to be time tracked
Precondition	User had created a project with board and tasks that can be time tracked
Success Guarantee	The project/board/task is successfully selected and perform time tracking
Sub-Tasks/Scenario	
1. User navigates to the time tracker and clicks on the add button.	
2. User select the project/board/task that they would like to track the time from the sub-form.	
3. User starts the time tracker to record the time spent on the project/board/task.	
Variant	
2a. No project is created. No selection available in the sub-form.	
2b. Create new project/board/task and repeat step 1.	

11. Use Case: Clickable project image

Use Case Name	Clickable project image
Actor(s)	Project Manager / Project Member
Purpose	To allow the project image to be clickable while viewing project details
Precondition	There are existing projects in the Project page
Success Guarantee	The project is successfully clicked on any area of the project image
Sub-Tasks/Scenario	
1. User navigates to the project page of the system.	
2. User clicks on the project image.	
3. User can now view the project details and tasks within it.	
Variant	
3a. No existing project available in the Project page.	
3b. Create a new project and repeat step 2.	

12. Use Case: Edit project details

Use Case Name	Edit project details
Actor(s)	Project Manager / Project Member
Purpose	To allow user to edit project details
Precondition	There are existing projects in the Project page
Success Guarantee	The project details are successfully edited
Sub-Tasks/Scenario	
1. User navigates to the project page of the system.	
2. User clicks on the “More” icon to edit project details.	
3. User edited project details and click on the “Save button”.	
4. Project details successfully changed.	
Variant	
3a. No existing project available in the Project page.	
3b. Create a new project and repeat step 2.	

13. Use Case: Improve GitHub page UI

Use Case Name	Improve GitHub page UI
Actor(s)	Project Manager / Project Member
Purpose	To improve the UI for GitHub page
Precondition	User must register an account and logged in to their account
Success Guarantee	User saw new UI for the page after logged in
Sub-Tasks/Scenario	
1. User logged in to their account and access to the system.	
2. User navigate to the GitHub page and saw new UI implemented.	
Variant	
-	

14. Use Case: Send message in Member's page

Use Case Name	Send message in members page directly
Actor(s)	Project Manager / Project Member
Purpose	To send message directly in members page
Precondition	The user has an existing account in the system
Success Guarantee	The user successfully send message in Members page
Sub-Tasks/Scenario	
1. User navigates to the Members page.	
2. User chooses a project member to be contacted and sends a message.	
Variant	
-	

15. Use Case: User status: online/offline

Use Case Name	User status: online/offline
Actor(s)	Project Manager / Project Member
Purpose	To view the status of user whether in online or offline mode
Precondition	The user has an existing account in the system and a created project with project member added
Success Guarantee	The user could saw the status of each team members
Sub-Tasks/Scenario	
1. User navigates to the Members page.	
2. User saw the status of each member in the same project.	
Variant	-

16. Use Case: Upload and Display profile picture

Use Case Name	Upload and Display profile picture
Actor(s)	Project Manager / Project Member
Purpose	To upload and display profile picture of the user
Precondition	The user had an existing account within the system
Success Guarantee	The user successfully uploaded and displayed their profile picture
Sub-Tasks/Scenario	
1. User navigates to the Profile page.	
2. User choose an image from their device and upload it to the system as profile picture.	
3. Profile picture is successfully uploaded and displayed next to their username.	
Variant	-

17. Use Case: Change personal information

Use Case Name	Change personal information
Actor(s)	Project Manager / Project Member
Purpose	To change user's personal information
Precondition	The user has an existing account within the system with their personal information recorded
Success Guarantee	The user successfully changed their personal information
Sub-Tasks/Scenario	
1. User clicks Profile page.	
2. User clicks on the "Edit" button, edits the personal information and clicks on the on the "Save" button.	
3. The user's personal information successfully updated.	
Variant	-

18. Use Case: Make announcement

Use Case Name	Make announcement
Actor(s)	Project Manager
Purpose	To make announcement to all other project members
Precondition	The user is a project manager and is logged into the account
Success Guarantee	The user made an announcement successfully
Sub-Tasks/Scenario	
1. User navigates to the announcement section.	
2. User enter the announcement message in the text box to be posted.	
3. User clicks on the “Post” button and uploads the announcement to the system.	
5. The announcement message appears in every other project members’ notification	
Variant	
-	

4. External Interface Requirements

4.1 User Interfaces

The Agile Software Project Management System applies certain rules of user interface design, such as simplicity, visibility, responsivity and consistency. The user interface will be designed to be user-friendly so that users can interact and works with the system without any technical Information Technology knowledge when using it. Simplicity principles will be applied on the system by removing all unnecessary elements from the system and having a clean layout and uniform color scheme to maintain the consistency of the system. Proper color scheme will be selected and used in the system for header, body and footer to show consistency and visibility of the system. The system will be designed to be responsive on the layout so that the system can be used on different device with different screen resolutions. Apart from that, proper heading will be provided for user in order for them to navigate to the correct screen on the system. Visible font size and font weight will be used in the system to maximize the consistency and visibility on the system.

4.2 Software Interfaces

As the Agile Software Project Management System is a web-based application, the system will be accessible through web browsers such as Internet Explorer, Safari, Mozilla Firefox, Google Chrome and other latest web browsers that supports web kit. In addition, the system can be accessed from any devices of different screen ratios and resolutions and also from any platforms including Windows, Android, iOS and Linux.

4.3 Communication Interfaces

Users will be using the Integrated Management System (IMS) through a PHP-based webpage which stores and extracts data from a MySQL-based database server over File Transfer Protocol (FTP). The browsers will be using Hyper Text Transfer Protocol (HTTP) and FTP as communication standards.

5. Non-functional Requirements

5.1 Quality Attributes

5.1.1 Usability

- The system will be having a simple user interface to ensure user-friendliness for users, where technical Information Technology knowledge are not required to operate the system.
- User Interface of the system will be designed according to the rules such as simplicity, visibility, responsivity and consistency.
- The design of the system is consistent throughout the whole system.
- The system will be designed to be responsive on the layout in order for user to operate the system in different devices that used different screen resolutions.
- Proper heading will be provided for user to navigate to the correct part in the system.
- User manual will be provided in another page of the system for users' guidelines on using the system and how the system works.

5.1.2 Reliability

- The system shows accurate data from the database such as time completion of task and timelines set by manager.
- The system shows analytical data and graph to user with a high level of accuracy by generating various type of reports according to users' needs.
- The system shows analytical data according to the specific input of the user such as the period of time to generate reports on that period of time.
- The system will be able to make comments and remind user on task completion.
- The system will be able to capture the timeline, sprints and milestones correctly.
- The system could perform testing for bug testing, reporting and fixing in order to notify user and generate report.

5.1.3 Security

- The system can only be used by the users that had registered an account on the system and when they are signed in.
- Users with different authority will be having different levels of access to perform different actions in the system.

5.1.4 Portability

- The system's web application will be compatible with cross platform browsers such as Microsoft Edge, Google Chrome, Safari and Mozilla Firefox.
- The system is able to be run from any portable devices with different operation system responsively.

5.1.5 Maintainability

- The system is able to be updated or modified based on different requirements.

- Additional updates and features can be implemented to the system if required in future.

5.1.6 Availability

- The system will have an uptime of more than 95% and it will be having a downtime of no more than an hour to fix or update in future.

6. Change Management Process

The following actions would be taken in the event that the clients propose a new set of requirements or request to amend the proposed requirements:

- Brainstorming will be carried out among the team in regards to the new set of requirements, after which the most feasible solution is determined.
- The proposal for all amended and new requirements will be documented and presented to the clients for approval.
- The team will proceed on implementing the changes upon the approval of the client.
- The changes will be integrated into the existing system.

8. References

- AltexSoft. 2020. *Non-Functional Requirements: Examples, Types, How To Approach.* [online] Available at: <<https://www.altexsoft.com/blog/non-functional-requirements/>> [Accessed 5 September 2020].
- Blog, D., 2020. *A Basic Non-Functional Requirements Checklist.* [online] Thoughts from the Systems front line.... Available at: <<https://dalbanger.wordpress.com/2014/01/08/a-basic-non-functional-requirements-checklist/>> [Accessed 5 September 2020].

Document Approval

This document is approved by:



Ms Phang Min Hui
(Project Client)

Fu Swee Tee



Digitally signed
by Fu Swee Tee
Date: 2021.03.19
08:00:39 +08'00'

Ms Fu Swee Tee
(Project Supervisor)



Lee Jia Jun
(Project Team Leader)