Music Player

**Software Development Plan (Small Project)**

**Version** 1.1

**Revision History**

| **Date** | **Version** | **Description** | **Author** |
| --- | --- | --- | --- |
| 11/06/2022 | 1.0 | Initial Version | Thiện Tiến, Anh Thư |
| 25/06/2022 | 1.1 | Update schedule and task assign after sprint planning 2 | Thiện Tiến, Anh Thư |

**Table of Contents**

[**Introduction**](#_30j0zll) **4**

[Purpose](#_yrhpyplkqi9p) 4

[Scope](#_8o6h0yw4i1x) 4

[Overview](#_sp0jv48ywr5d) 4

[**Project Overview**](#_d49hf817m9zm) **4**

[Project Purpose, Scope, and Objectives](#_3dy6vkm) 4

[Assumptions and Constraints](#_mtxetysof49y) 5

[Constraints:](#_hlkf9sfgb53x) 7

[Project Deliverables](#_yomdrofcr0bi) 7

[**Project Organization**](#_wsr8pouv86e0) **8**

[Organizational Structure](#_17dp8vu) 8

[Roles and Responsibilities](#_pgtmw1uegjw6) 8

[**Management Process**](#_eoeqht4gim52) **9**

[Project Estimates](#_lnxbz9) 9

[Project Plan](#_u7wm5wj2l46t) 9

[Project Monitoring and Control](#_xf6cyh8qsxi) 14

[Requirements Management](#_2l95n09k0eun) 14

[Reporting and Measurement](#_2bn6wsx) 14

[Risk Management](#_134xkljhk3be) 14

[Configuration Management](#_jxn5b47y1ga2) 15

**Software Development Plan (Small Project)**

# **Introduction**

The introduction of the **Software Development Plan** provides an overview of the entire document. It includes the purpose, scope, definitions, acronyms, abbreviations, references, and overview of this **Software Development Plan**.

## **Purpose**

The purpose of the *Software Development Plan* is to gather all information necessary to control the project. It describes the approach to the development of the software and is the top-level plan generated and used by managers to direct the development effort.

The following people use the *Software Development Plan*:

* The **project manager** uses it to plan the project schedule and resource needs, and to track progress against the schedule.
* **Project team members** use it to understand what they need to do, when they need to do it, and what other activities they are dependent upon.

## **Scope**

This *Software Development Plan* describes the overall plan to be used by the Music Player, including deployment of the product.

The plans as outlined in this document are based upon the product requirements as defined in the *Vision Document*.

## **Overview**

This *Software Development Plan* contains the following information:

Project Overview — provides a description of the project's purpose, scope, and objectives.  It also defines the deliverables that the project is expected to deliver.

Project Organisation — describes the organisational structure of the project team.

# **Project Overview**

## **Project Purpose, Scope, and Objectives**

Music is one of the best ways to relieve pressure in stressful modern society life. Then our idea is to create a music streaming platform using Android like Spotify/SoundCloud or Zing MP3.

Since the application is not commercial in nature, it will be free for users to use. There will be three main types of user:

* Audience: they can stream or download music, create their own playlists and add songs to favourites.
* Artist: they can upload new songs, album to app; view statistic of song, albums
* Admin: they can approve new songs from artists and manage accounts.

Application is developed and given users some buttons for simple operations such as play, stop, continue, repeat, shuffle. In addition, the music player will also add Light or Dark theme mode to enhance the user’s experience . Moreover, the application also allows users to share their favourite songs on social networks.

## **Assumptions and Constraints**

**Assumptions:**

* Delivery:
  + Project servers arrive configured as expected.
  + Achieve all the users (stakeholder) requirements.
  + Deliver software that meets the customer’s expectations.
  + Give users a high learnability application without too many complex features, enhance the interaction between the user and the media control so that the user can have better experience to achieve real pressure relief.
  + is delivered on time and on schedule
* Budget: Estimated cost of the project

Salary of the project developer:

* + UI Design: 50.000d ($2) for an hour.
  + Front-end: 20.000d ($1) for an hour.
  + Back-end: 50.000d ($2) for an hour.
  + Database: 35.000d (~$1.5) for an hour.

|  | **Function of the project** | **The performer** | **Implement Time** |
| --- | --- | --- | --- |
| 1 | Login and register accounts ( for audience, artist, admin) | UI design | 3 hour |
|  | Login and register accounts ( for audience, artist, admin) | Front\_end | 5 hour |
|  | Login and register accounts ( for audience, artist, admin) | Back\_End | 6 hour |
| 2 | Audience: stream music, download music | UI design | 3 hour |
|  | Audience: stream music, download music | Front\_End | 5 hour |
|  | Audience: stream music, download music | Back\_End | 6 hour |
| 3 | Audience: create own playlists | UI design | 3 hour |
|  | Audience: create own playlists | Front\_End | 5 hour |
|  | Audience: create own playlists | Back\_End | 6 hour |
| 4 | Audience: add song to favourite | UI design | 3 hour |
|  | Audience: add song to favourite | Front\_End | 5 hour |
|  | Audience: add song to favourite | Back\_End | 6 hour |
| 5 | Artist: upload new songs, albums | UI design | 3 hour |
|  | Artist: upload new songs, albums | Front\_End | 5 hour |
|  | Artist: upload new songs, albums | Back\_End | 6 hour |
| 6 | Artist: view statistics of songs, albums | UI design | 3 hour |
|  | Artist: view statistics of songs, albums | Front\_End | 5 hour |
|  | Artist: view statistics of songs, albums | Back\_End | 6 hour |
| 7 | Admin : approve songs | UI design | 3 hour |
|  | Admin : approve songs | Front\_End | 5 hour |
|  | Admin : approve songs | Back\_End | 6 hour |
| 8 | Admin: manage account | UI design | 3 hour |
|  | Admin: manage account | Front\_End | 5 hour |
|  | Admin: manage account | Back\_End | 6 hour |
| 9 | Others function when stream music ( play, pause, repeat, shuffle) | UI design | 3 hour |
|  | Others function when stream music ( play, pause, repeat, shuffle) | Front\_End | 5 hour |
|  | Others function when stream music ( play, pause, repeat, shuffle) | Back\_End | 6 hour |
| 10 | Search | UI design | 3 hour |
|  | Search | Front\_End | 5 hour |
|  | Search | Back\_End | 6 hour |
| 11 | Share | UI design | 3 hour |
|  | Share | Front\_End | 5 hour |
|  | Share | Back\_End | 6 hour |
| 12 | Save information of user account on system | Database | 20 hour |
| 13 | Save music on system | Database | 60 hour |

Cost of the project estimate = The Performance Salary \* Implement Time \* 2

UI Design Cost = 50.000\* 33 (hours) = 1.650.000d

Front\_End Cost = 20.000\* 55 (hours) =1.100.000d

Back\_End Cost = 50.000 \*66 (hours) = 3.300.000d

Database Cost = 35.000 \*80(hours)=2.800.000d

—--------------------------------------------------------

Project Cost: 8.850.000d\*2 =17.700.000d

* Schedule:
  + Each meeting occur at the beginning of weeks to assign jobs and occur at the end of weeks to assess jobs
  + Each member does at least 3 tasks a week.
  + The unexpected always happens. Always allow contingency in planning. If there is a delay, it is necessary to step up the work to catch up with the project progress soon.
  + Estimating the difficulty of problems and hence the cost of developing a solution is hard.
* Staff:
  + Keep the members' spirits upbeat
  + Happy, sociable communication between each member.
  + Take care of other staff's health.
  + Information must be exchanged on the status of work, design decisions and changes to previous decisions.

## Constraints:

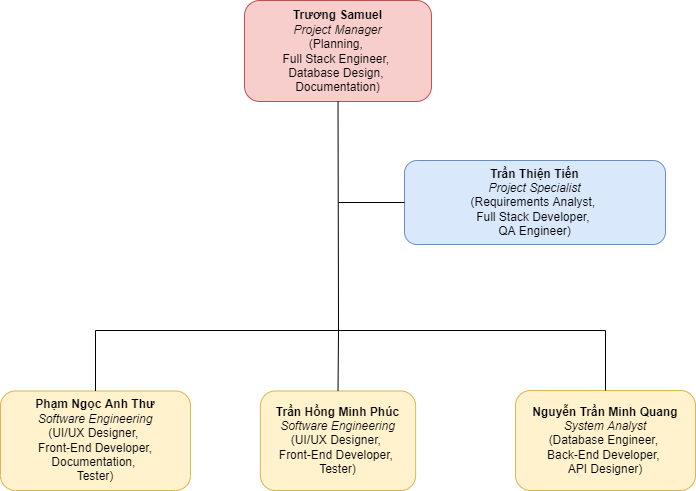
* Time: The time constraint refers to the project’s schedule for completion, including the deadline for each phase of the project and the date for rollout of the final deliverable.
* Scope: specific goals, deliverables, features, functions.
* Risks: Assess the risks that may affect project, monitor these risks and take action when problems arise.
* Staff: establish ways of working that lead to effective team performance

## **Project Deliverables**

Deliveries for each project phase are identified in the Development Case. Deliverables are deliverables are delivered towards the end of the iteration, as specified in section 4.2 at every Sprint.

# **Project Organization**

## **Organizational Structure**



## **Roles and Responsibilities**

| **Member** | **Role** |
| --- | --- |
| Trương Samuel, Project Manager | Planning, Full Stack Engineer, Database Design, Documentation |
| Thiện Tiến, Project Specialist | Requirements Analyst, Full Stack Engineer, QA Engineer |
| Anh Thư, Software Engineering | UI/UX Designer, Front-End Developer, Documentation, Tester |
| Minh Phúc, Software Engineering | UI/UX Designer, Front-End Developer, Tester |
| Minh Quang, System Analyst | Database Engineer, Back-End Developer, API Designer |

# **Management Process**

## **Project Estimates**

None

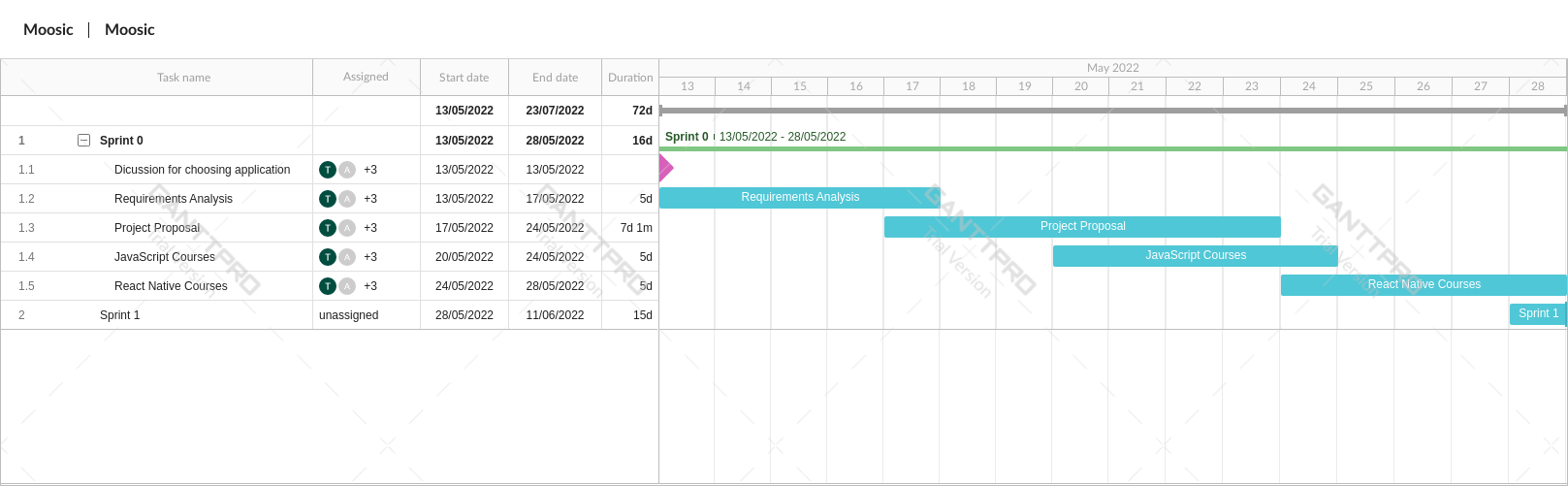
## **Project Plan**

**Sprint 0:**

* *Start Date:* 13/05/2022
* *End Date:* 28/05/2022
* *Deliverables:* A proposal - document that describes the project overview and key features.
* Tasks:

| **#** | **Task name** | **Purpose** | **Assigned** | **Start Date** | **Duration** |
| --- | --- | --- | --- | --- | --- |
| 1 | Discussion for choosing application | Understand the application will be developed in the course | All members | 13/05/2022 | 2 days |
| 2 | Requirement Analysis | Find out what popular apps are already on the market such as Spotify, ZingMP3 to list important requirements and features | All members | 15/05/2022 | 2 days |
| 3 | Project Proposal (PA0) | Introduction, Target User and Environments, Key features of application | All members | 17/05/2022 | 7 days |
| 4 | JavaScript Courses | Practice with the existing projects to get ideas when applying to group project | All members | 24/05/2022 | 4 days |
| 5 | React Native Courses | All members | 20/05/2022 | 4 days |

* *Gantt Chart for Sprint 0:*

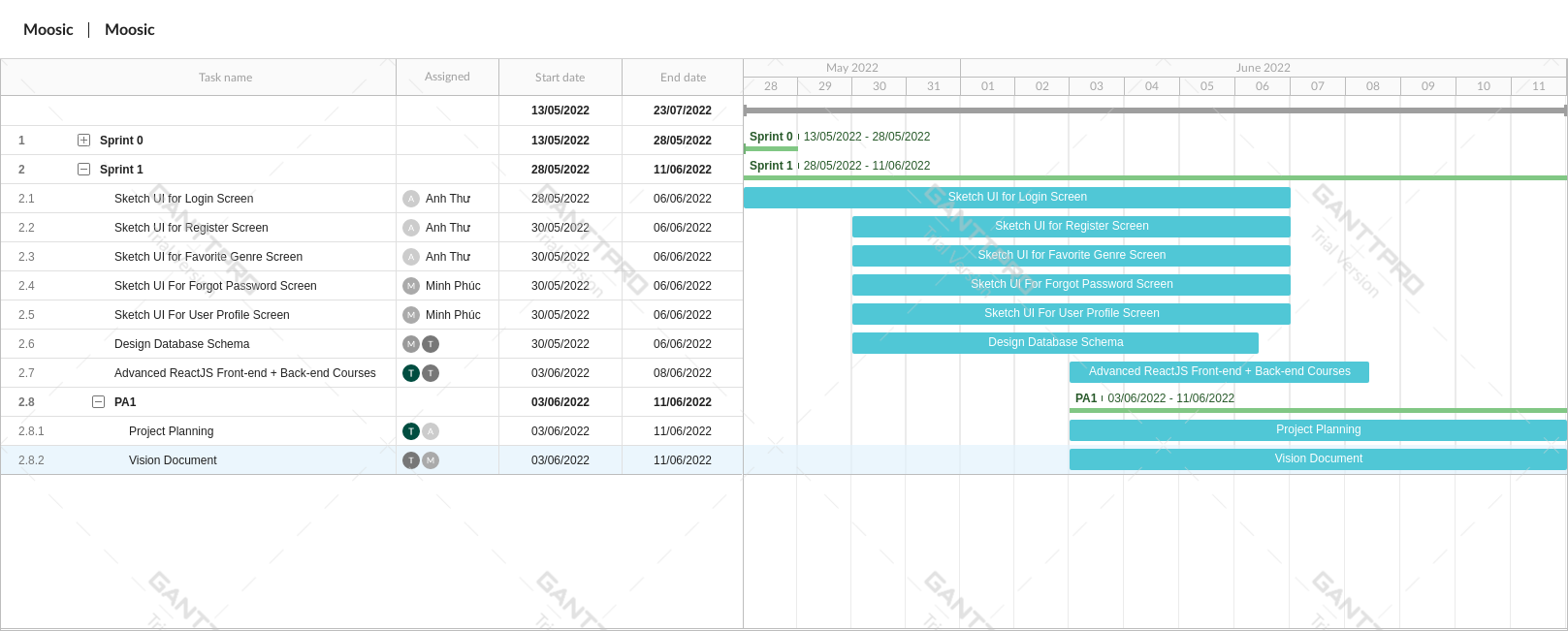


**Sprint 1:**

* *Start Date:* 28/05/2022
* *End date:* 11/06/2022
* *Deliverables:* A project plan what describes the requirements that will be implemented in the project and a vision document defines the high-level scope and purpose of project.
* Tasks:

| **#** | **Task name** | **Purpose** | **Assigned** | **Start Date** | **Duration** |
| --- | --- | --- | --- | --- | --- |
| 1 | **Sketch UI for:**  Login Screen  Register Screen  Favorite Genre Screen | Understand the color tone as well as the location of the appropriate buttons for the user to manipulate | Anh Thư | 30/05/2022 | 7 days |
| 2 | **Sketch UI for:**  Forgot Password Screen  User Profile Screen | Minh Phúc | 30/05/2022 | 7 days |
| 3 | Create Database Schema | Understand organization of data and the relationships between tables | Trương Samuel  Minh Quang | 30/05/2022 | 7 days |
| 4 | Advanced ReactJS Front-end + Back-end Courses | Understand how to interact between front-end and back-end | Trương Samuel  Thiện Tiến | 03/06/2022 | 5 days |
| 5 | Project Planning | PA1 | Thiện Tiến  Anh Thư | 03/06/2022 | 7 days |
| 6 | Vision Document | Trương Samuel  Minh Phúc | 03/06/2022 | 7 days |

* *Gantt Chart for Sprint 1:*

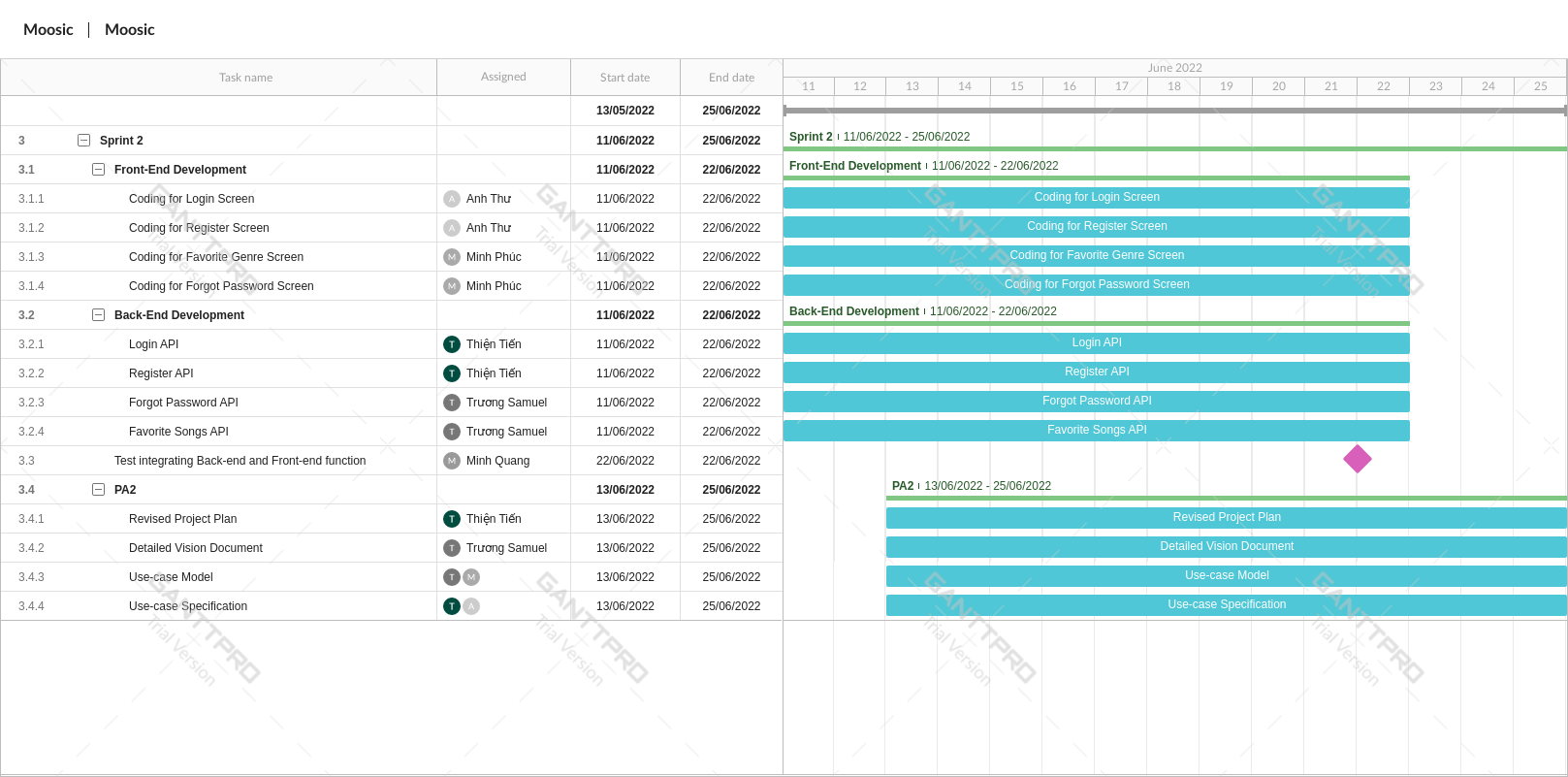


**Sprint 2:**

* *Start Date:* 11/06/2022
* *End Date:* 25/06/2022
* *Deliverables:* The document will bring reader the processes that occur within the system and more detail product functional and non-functional requirements, alternatives and competitors, user environments.
* Tasks:

| **#** | **Task name** | **Purpose** | **Assigned** | **Start Date** | **Duration** |
| --- | --- | --- | --- | --- | --- |
| 1 | Coding for Login, Register Screen | Convert from sketching to coding | Anh Thư | 11/06/2022 | 11 days |
| 2 | Coding for Favorite Songs, Forgot Password Screen | Minh Phúc | 11/06/2022 | 11 days |
| 3 | Login, Register API | Check user information with Database | Thiện Tiến | 11/06/2022 | 11 days |
| 4 | Forgot Password, Favourite Songs API | Trương Samuel | 11/06/2022 | 11 days |
| 5 | Integrating Back-End and Front-End function | ***Milestone: Combining Front-end and Back-end functionality for the first time*** | Minh Quang | 22/06/2022 | 4 days |
| 6 | Revised Project Plan | PA2 | Thiện Tiến | 13/06/2022 | 12 days |
| 7 | Detailed Vision Document | Trương Samuel | 13/06/2022 | 12 days |
| 8 | Use-case Model | Trương Samuel  Minh Phúc | 13/06/2022 | 12 days |
| 9 | Use-case Specification | Thiện Tiến  Anh Thư | 13/06/2022 | 12 days |

* *Gantt Chart for Sprint 2:*



**Sprint 3:**

* *Start Date:* 25/06/2022
* *End Date:* 09/07/2022
* *Deliverables:* the document is the deliverable container for the core architectural artifacts created during a project and for important related information. The class diagram includes classes, their main attributes, operations and relationships of the key components defined in the architecture.
* Tasks:

| **#** | **Task name** | **Purpose** | **Assigned** | **Start Date** | **Duration** |
| --- | --- | --- | --- | --- | --- |
| 1 | Sketch Playlist Screen | Sketch UI | Anh Thư | 25/06/2022 | 5 days |
| 2 | Sketch Music Screen | Minh Phúc | 25/06/2022 | 5 days |
| 3 | Code playlist screen | None | Anh Thư | 01/07/2022 | 8 days |
| 4 | Complete UI forgot password | Completed delayed task last week | Minh Phúc | 25/07/2022 | 4 days |
| 5 | Code music screen (Backend) | None | Trương Samuel | 27/06/2022 | 12 days |
| 6 | Upload music API | None | Minh Quang | 27/06/2022 | 12 days |
| 7 | Revised Use-case Model | PA3 | Thiện Tiến | 27/06/2022 | 12 days |
| 8 | Revised Use-case Specification | Thiện Tiến | 27/06/2022 | 12 days |
| 9 | Define Software Architecture | Trương Samuel  Minh Phúc | 27/06/2022 | 12 days |
| 10 | Class Diagram | Thiện Tiến  Anh Thư | 27/06/2022 | 12 days |

* *Gantt Chart for Sprint 3:*



**Sprint 4:**

* *Start Date:* 09/07/2022
* *End Date:* 23/07/2022
* *Deliverables:* The document describes sketching the user interface of the designing system for the main screen and 2 anothers for the key scenarios described in the use-case specifications. Besides, we also deliveried the beta version of the application to collect user reviews.
* Tasks:

| **#** | **Task name** | **Purpose** | **Assigned** | **Start Date** | **Duration** |
| --- | --- | --- | --- | --- | --- |
| 1 | Revise SAD | PA4 | To be determined at  Sprint Planning 3 | 11/07/2022 | 12 days |
| 2 | UI Prototype | 11/07/2022 | 12 days |
| 4 | Some tasks before first application release | To be determined at  Sprint Planning 3 | To be determined at  Sprint Planning 3 | |
| 3 | Application (version Beta) Release | For the instructor to evaluate and team will improve after receiving the comments from the lecturer before releasing the final product |

* *Gantt Chart for Sprint 4:* Will be added after Sprint Planning 2.

**Sprint 5:**

* *Start Date:* 23/07/2022
* *End Date:* To be announced later
* *Deliverables:* The final product that meets all listed requirements.
* Tasks:

| **#** | **Task name** | **Purpose** | **Assigned** | **Start Date** | **End Date** |
| --- | --- | --- | --- | --- | --- |
| 1 | Test Plan | PA6 | To be determined at  Sprint Planning 4 | 13/07/2022 | To be determined |
| 2 | Test Cases | 13/07/2022 |
| 4 | Test Report | 13/07/2022 |
| 3 | Project Presentation | 13/07/2022 |
| 4 | Application Final Release | Products for presentations | 13/07/2022 | Before 3 days at the end of Sprint 5 |

* *Gantt Chart for Sprint 5:* Will be added after Sprint Planning 3.

## **Project Monitoring and Control**

### *Requirements Management*

The requirements for this system are captured in the Vision document. Requested changes to requirements are captured in Change Requests, and are approved as part of the Configuration Management process.

### *Reporting and Measurement*

None

### *Risk Management*

Risks will be identified in the Inception Phase using the steps identified in the RUP for Small Projects activity “Identify and Assess Risks”. Project risk is evaluated at least once per iteration and documented in this table. The risks of the greatest magnitude are listed first in the table.

| **Risk Ranking (High, Medium, Low)** | **Risk Description and Impact** | **Mitigation Strategy and/or Contingency Plan** |
| --- | --- | --- |
| Catastrophic (High) | Capability of team members is not equal, hard for the team leader divides the work unevenly and does not match each person's ability. | Accept the risk, each member has to be responsible for their own given tasks. Each member actively searches for information and helps each other to complete the task on time. Regularly monitor the progress of weak team members |
| Serious (High) | Don’t have experience making software systems before. | Each members need to proactive in finding information, improving skills |
| Catastrophic (Low) | There is no general consensus among members | Resolve conflicts by voting and taking by the number |
| Tolerable (Low) | A member cant does anything due to health condition (paralyzed, comatose because of traffic accidents) | The supervisor knows the status of the staff to make choices: assign less tasks, tasks can be done later, put more people on that part .. to ensure the work does not stall when problems arise. health issue among members |
| Tolerable (Moderate) | Programming environment inconsistency. | According to the majority, those in the minority have to try to find a way to fix it, download support software, suitable working environment, use available tools, support websites, .. |
| Tolerable (High) | Weak time management, wait until the deadline is close to start focusing on work. | Supervisors must know how to remind, closely check and urge projects, people, and help to prevent further delays. |
| Tolerable (High) | The size of the software is underestimated. | Check the previous versions of the project, those available on the market for the most objective and comprehensive assessment, thereby reducing project underestimation |
| Serious (very low) | PC/laptop specs doesn’t meet the requirements for the specific programs | Searching for optimal, flexible alternatives using available software and programs |

### *Configuration Management*

None