<Team 104>

<FitUS>
Software Development Plan

Version <1.2>

Revision History

Date	Version	Description	Author
<15/06/23>	<1.0>	<The initial version of the Project Plan>	<Team 104>
<27/06/23>	<1.1>	<The revised version of the Project Plan, including: 1. Changes to the project plan in 4.2.1, specifically the Gantt chart; 2. Added and corrected information throughout the document>	<Team 104>
<02/09/23>	<1.2>	<Adjust the scope in the section Project Overview (BMI)>	<Team 104>

Table of Contents

Software Development Plan

1. Introduction

The introduction of a **Software Development Plan** provides an overview of the document which builds a plan as a structural model**.** This document includes a purpose, scope, definitions, acronyms, abbreviations, references and overview of this project. Based on a project purpose and scope, this **Software Development Plan** synthesizes not only organization of role in project team, but also an abstract process in each phase of building project.

2. Project Overview

2.1 Project Purpose, Scope, and Objectives

The purpose of this project is to admonish users to usually take care of their health based on features of this web application, which provide substantial parameters to easily manage their weight. Nowadays, people are usually overwhelmed with their work and daily life. As a consequence, health is an aspect that most people overlook. Thus, the demands for health checking are crucial, and so is FitUS.

All individuals are the target audience since they have a significant need for health monitoring and attractiveness management. Users must be familiar with the fundamentals of utilizing website apps (sign in, create an account, etc.) in order to use this service.

This service can consump in other applications such as Web browsers (Safari, Chrome, Microsoft Edge, Opera, Firefox, etc.). In the development phase, this project will be programmed by using programming languages such as in Front-end included CSS, HTML, JavaScript/ TypeScript and in Back-end included ExpressJS (framework), JavaScript, MySQL/ Mongoose.

In this project, the objectives of it exist in both substance and mental for customers. It provides advice, an opinion and the statistics for users to have a different view about their health, about their fitness. Throughout the period using this application, customers will recognize the modification of their strength, their physical appearance which many statics and predictions this application provides.

Base on the project's purpose and objectives, the scopes of this project are:

- Health Record: Preserving a set of health statistics provided by the users.
- Health Monitoring: Illustrating charts and graphs representing users' health conditions.
- Edit health records: Adding and updating personal health information.
- View BMI value: Using your height and weight to determine the overall body weight is healthy or not.
- Food portion consultant: Providing users with a list of foods. Therefore, helping the users achieve their ideal physique.

2.2 Assumptions and Constraints

There are five individuals on the team building this project, and no more will be added while it is in progress.

Knowledge service in this project: Basic knowledge about Object-oriented programming, Frontend: Javascript/ HTML/ CSS, Backend: ExpressJS (framework), JavaScript, MySQL/ MongoDB.

This project has a zero-budget.

Meeting schedule:

- Planning: Every first Monday of every sprint at 20:00.
- Weekly Scrum: Every Wednesday or Thursday at 20:00.
- Review: Every last Saturday of every sprint at 15:00.

Each role of each person can be changed during the project.

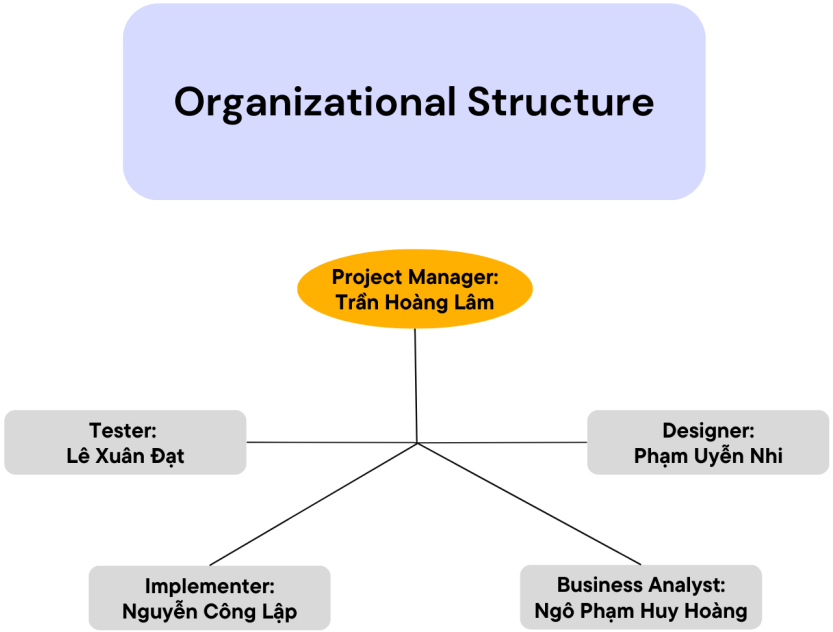
2.3 Project Deliverables

Below is the list of deliverables which will be produced during the project:

- **Project Management Artifacts:**
  - Software Development Plan: target delivery date: at the end of Sprint 1 and 2.
- **Requirements Artifacts:**
  - Vision Document: target delivery date: at the end of Sprint 1 and 2.
  - Use-case Specification: target delivery date: at the end of Sprint 2 and 3.
- **Analysis & Design Artifacts:**
  - Software Architecture Document: target delivery date: at the end of Sprint 3 and 4.
  - User Interface Prototype: target delivery date: at the end of Sprint 4.
- **Test Artifacts:**
  - Test Plan, Test Cases and Test Evaluation Summary: target delivery date: at the end of Sprint 5 and 6.
- **Product Releases**: the first version will be released at the end of Sprint 5 and the second will be released at the end of Sprint 6.

3. Project Organization

3.1 Organizational Structure



3.2 Roles and Responsibilities

Person	Role

Tran Hoang Lam	Project Manager
Pham Uyen Nhi	Designer
Nguyen Cong Lap	Implementer
Ngo Pham Huy Hoang	Business Analyst
Le Xuan Dat	Tester

Person	**Role**	**Responsibility**
Tran Hoang Lam	Project Manager	- Write a project plan - Organize meetings - Summarize point of project assignment
Pham Uyen Nhi	Designer	- Design a system (including UI, UX and database)
Nguyen Cong Lap	Implementer	- Write source code - Review source code
Ngo Pham Huy Hoang	Business Analyst	- Communicate with team about requirements - Analysis requirements
Le Xuan Dat	Tester	- Report performance of project - Write test plan and test case

4. Management Process

4.1 Project Estimates

The project will be released at the end of August. The project estimates are based on the requirements in each project assignment given by the TAs. The project will be estimated again when releasing the first version of the application.

4.2 Project Plan

4.2.1 Phase and Iteration Plan

The project consists of 4 phases: Inception, Elaboration and Construction, Transition

- **Inception**: Week 1,2 (~Sprint 1):
  - Starting date: 11/06/2023
  - Ending date: 24/06/2023
  - Overall objectives: drafting initial versions of Project Plan and Vision Document.
- **Elaboration**: Week 3-6 (~Sprint 2,3):
  - Starting date: 25/06/2023
  - Ending date: 22/07/2023
  - Overall objectives:
    - Releasing revised versions of Project Plan and Vision Document.
    - Use-case document and use-case model.
    - Design document: defining software architecture in Software Architecture Document with class diagrams, designing the UI.

        - Preparing the test plan of the product.
- **Construction**: Week 7-10 (~Sprint 4,5):
    - Starting date: 23/07/2023
    - Ending date: 19/08/2023
    - Overall objectives:
        - Releasing revised version of Software Architecture Document.
        - Designing test cases and summarizing test results in the test report.
        - Releasing the first version of the product.
- **Transition**: Week 11,12 (~Sprint 6):
    - Starting date: 20/08/2023
    - Ending date: 02/09/2023
    - Overall objectives:
        - Filling in test summary report.
        - Implementing automated testing.
        - Releasing the second (final) version of the product.
        - Project presentation and product demo.

For detailed project plan and task assignment, the team's Gantt Chart can be found here.

4.2.2 Releases

Throughout the project, there will be a total of 2 releases:

- The first one is planned to be released after the 5th Sprint (PA5) on August 19th, 2023
- The second one is planned to be released after the testing period in the 6th Sprint (PA6) on September 2nd, 2023. This is also the project's final release.

4.2.3 Project Schedule

Milestones	Finish Date
Finish Phase 1: Inception	24/06/2023
Finish Phase 2: Elaboration	22/07/2023
Finish Phase 3: Construction	19/08/2023
Finish Phase 4: Transition	02/09/2023
Release the first version	19/08/2023
Release the final version	02/09/2023

4.3 Project Monitoring and Control

4.3.1 Reporting

- Discord will be used for discussing the project (what should we do? programming language? ideas? etc) and reporting the project status weekly.
- Facebook/Slack will be used to ask some project-related questions to the TAs.
- Messenger will be used for frequently informal chatting.
- Trello will be used for reporting the state of completeness of the tasks assigned to each member (checklist/activity sections).

4.3.2 Risk Management

Risk ID	***Risk Description***	***Probability***	***Impact***	***Priority***	***Mitigation Strategy or Contingency Plan***
001	Lack of coding knowledge	High	Being not able to implement the product	High	Attend web programming courses regularly. Focus on an easier path of implementing the product.
002	Hardware unavailability	Low	Cannot run/test the product	High	Find a more lightweight way of implementing the product that doesn't require too many hardware resources. If the current implementation is already optimal, it is a must to upgrade the hardware system.
003	Absent/Unavailable members	Medium	Being not able to complete tasks on time, or catch up with group's plan and works	High	Assign tasks of the absent member to the remaining members. Apply stricter ways for scoring task's completeness of each member if the situation gets worse.
004	Late submission of prerequisite tasks	High	Tasks that depend on prerequisite tasks can also be submitted late due to longer waiting time	High	Regularly remind every member of their assigned task and deadlines.
005	Lost files	Low	Lose all the progress of the current work	Medium	Backup files in different places with as many copies as possible. Save files every time a member finishes their parts.
006	Unexpected change in requirements	Medium	Some tasks can be interrupted, or canceled due to new requirements	Medium	Quickly adapt to the changes, find solutions for the requirements and make changes to the tasks assigned to each member.
007	Time required to develop the software is underestimated	Low	Late delivery of artifacts	Medium	Re-estimate the time needed and update the project plan/project schedule. If it is too late to change, extend team members' work time per day to catch up with the progress.

4.3.3 Configuration Management

- Google Drive: storing and sharing documents (like Software Development Plan document, Vision document, …) and project reports.
- GitHub: managing source code and related files.
- Discord: storing temporary code, ideas and conversations.

- Trello: attaching links to documents that need to be worked on, also storing submission files of each member.