LemmeDoItForU

LemmeCook
Software Development Plan
Version <1.1>

Revision History

Date	Version	Description	Author
<29/05/2024>	<1.0>	First version of Software Development Plan	LemmeDoItForU Team
<08/07/2024>	<1.1>	Revised version of Software Development Plan	LemmeDoItForU Team

Table of Contents

Software Development Plan

1. Introduction

A **Software Development Plan** (**SDP**) is a concise roadmap that outlines the objectives, processes, and resources necessary for a software project. It includes key elements such as project scope, requirements, architecture, schedule, and risk management strategies. The **SDP** ensures that all stakeholders are aligned, resources are allocated efficiently, and potential challenges are proactively addressed. By providing a clear and structured approach, the **SDP** enhances communication and increases the likelihood of delivering a high-quality software product on time and within budget.

2. Project Overview

2.1 Project Purpose, Scope, and Objectives

2.1.1 Purpose

The purpose of this project is to create an application where users can get anything related to food such as: recipes of specific dishes, suggestions of what to eat for a specific purpose (losing weight, healthy diet, etc.), suggestions of what to cook in different situations (party, family gathering, etc.), etc.; and to make revenue from this application.

2.1.2 Scope

A mobile app for cooking enthusiasts.

2.1.3 Objectives

A useful and attractive application for users with an interest in things that related to cooking, eating and food.

2.2 Assumptions and Constraints

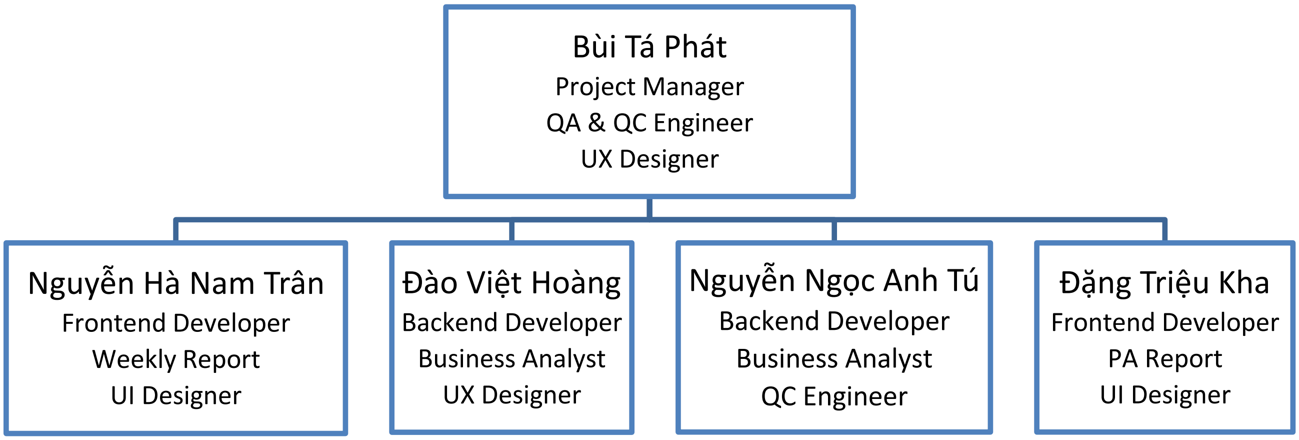
- Project has fixed schedule of 12 weeks.
- Budget: 1000$.
- Each group member is only allowed to come into the meeting late for a maximum of 15 minutes.
- Each group member must follow Coding convention for TypeScript.
- Schedule: Meetings take place on Wednesdays (offline), Saturday or Sunday (online).
- Staff:
  - Backend Developers: 2 persons
  - Frontend Developers: 2 persons
- Equipment:
  - 2 mobile phones (for frontend developers)
  - 5 laptops

2.3 Project Deliverables

Deliverables for each project phase are identified in the Development Case. Deliverables are delivered towards the end of the interaction, as specified in section 4.2.4 Project Schedule.

3. Project Organization

3.1 Organizational Structure



3.2 Roles and Responsibilities

Person (Name, role)	**Responsibility**
Bùi Tá Phát *Project Manager, QA & QC Engineer, UX Designer*	Manage projects, assign work to members, and provide coding conventions, also responsible for tracking bugs and debugging.
Nguyễn Hà Nam Trân *Frontend Developer, Weekly Report, UI Designer*	Write weekly report, responsible for front-end part and UI of mobile app.
Đào Việt Hoàng *Backend Developer, Business Analyst, UX Designer*	Collect customer requirements and connect the development team with the customer, also responsible for back-end part and UX of mobile app.
Nguyễn Ngọc Anh Tú *Backend Developer, Business Analyst, QC Engineer*	Collect customer requirements and connect the development team with the customer, also responsible for developing back-end of mobile app, testing all features, functions and errors occurred during development.
Đặng Triệu Kha *Frontend Developer, PA Report, UI Designer*	Responsible for writing the PA report, developing front-end part and UI of mobile app.

4. Management Process

4.1 Project Estimates

- Knowledge and skill’s cost: ~1000$ (~200$/member)
- Human resources cost: ~12 labor weeks/member

4.2 Project Plan

Phase and Iteration Plan	*Sprints*	*Start-End*	*Task*	*Result*
Inception	Sprint 1	20/05/2024 - 27/05/2024	Idea for the project. Set up environment for the project. Define roles for project members. Write weekly report. Project proposal.	PA0. Weekly Report 1.
Elaboration	Sprint 2	27/05/2024 - 10/06/2024	Front-end training: HTML, CSS, TypeScript, Expo. Back-end training: NodeJS, MongoDB, TypeScript, Expo. Vision document. Project plan. Write weekly report and PA.	PA1. Weekly Report 2. Weekly Report 3. Have basic knowledge for building a project.
	Sprint 3	10/06/2024 - 24/06/2024	Continue learning and training technology. Use-case model. Use-case specification. Detailed vision document and project plan. Market research. Design class diagram. Design UI. Write weekly report.	Weekly Report 4. Weekly Report 5. Design of project on Figma: Homepage, Chatbot's chat box, etc. A basic class diagram. Have enough skills for building a project
	Sprint 4	24/06/2024 - 15/07/2024	Revised project plan and vision document. Use-case model. Use-case specification. Front-end: Coding some mini project to gain experience. After that, start coding UI with services from back-end. Back-end: Coding API/services (mainly database) for calling from front-end. Write weekly report and PA.	PA2. Weekly Report 6. Weekly Report 7. An improved class diagram. An end-product with basic I/O and database processing. Have enough skills for building a project.
Construction	Sprint 5	15/07/2024 - 29/07/2024	Design class diagram. Improve UI/UX, logic, validate and animation. Preparing test plan and test cases.	PA3 Weekly Report 8. Weekly Report 9.

			Continue develop front-end's detail and back-end's I/O and data processing. Provide first end-product releases. Write weekly report and PA.	A completed class diagram. A releasable product.
	Sprint 6	29/07/2024 - 12/08/2024	Continue improve the project for passing all tests. UI prototype. Prepare more test plan and test cases. Write weekly report and PA. Provide second end-product releases. Prepare the presentation.	PA4. Weekly Report 10. Weekly Report 11. An improved product.
	Sprint 7	12/08/2024 - 26/08/2024	Prototyping and Testing. Provide final end-product releases if possible. Project presentation and demo. Write PA.	PA5. Weekly report 12. Weekly report 13. Presentation project. Demo. Final submission. Final product.

4.2.1 *Releases*

Version	Target release date	Description
First version.	29/07/2024	A product with all features of high and medium priority.
Second version.	12/08/2024	A product with all features and pass all basic tests.
Final version.	26/08/2024	A completed product.

4.2.2 *Project Schedule*

Name	Phase	Target date
Role designation and project proposal	Sprint 1	27/05/2024
Training and vision document	Sprint 2	10/06/2024
Detailed vision document and project plan, *design, and class diagram*	Sprint 3	24/06/2024
Extensive Training and detailed class diagram	Sprint 4	15/07/2024

Building project and testing preparation	Sprint 5	29/07/2024
Building prototype and preparing presentation	Sprint 6	12/08/2024
Prototyping and testing, and presentation	Sprint 7	26/08/2024

4.3 Project Monitoring and Control

4.3.1 Reporting

- Weekly meeting online on slack or offline
- Weekly status report
- Informal chats

4.3.2 Risk Management

Risk ID	*Risk Description*	*Probability*	*Impact*	*Priority*	*Mitigation Strategy or Contingency Plan*
RT1	Technology Integration Issues	Medium	High	High	Conduct thorough testing of all integrated systems before deployment. Ensure regular updates and compatibility checks.
RT2	Software Bugs and Glitches	High	Medium	High	Implement thorough testing, including unit, integration, and system tests. Establish a protocol for quick bug fixes and regular software updates.
RH1	Timeline Delays	Medium	High	Medium	Develop a detailed project schedule with buffer times, regularly track progress against milestones, and adjust schedules as needed.
RH2	Team member is busy or doesn’t have time to work	Medium	Low	Medium	All members of team discuss and divide jobs again to minimize the bad affect to the project

4.3.3 Configuration Management

- Git, GitHub: for distributed version control and source code, related files management.
- Trello: for creating and managing task boards.
- Slack, Messenger: for discussion and planning for the project.
- Figma: for design user interface