React To Spring

Forum
Software Development Plan (Small Project)
Version <1.1>

Revision History

Date	Version	Description	Author
11/11/2024	1.1	Revised version of the initial one	Tuyen

Table of Contents

Software Development Plan (Small Project)

1. Introduction

1.1 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.

1.2 Scope

This *Software Development Plan* describes the overall plan to be used by the Forum project, including deployment of the product. The details of the individual iterations will be described in the Iteration Plans. The plans as outlined in this document are based upon the product requirements as defined in the *Vision Document*.

1.3 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 Organization — describes the organizational structure of the project team.

2. Project Overview

2.1 Project Purpose, Scope, and Objectives

This project is for those who wish to share their lives and connect with family, friends, and those around them. A place where they can share joy, sorrow, achievements, and even memorable daily moments. Additionally, it can be used by those who want to build a personal brand for their business.

2.2 Assumptions and Constraints

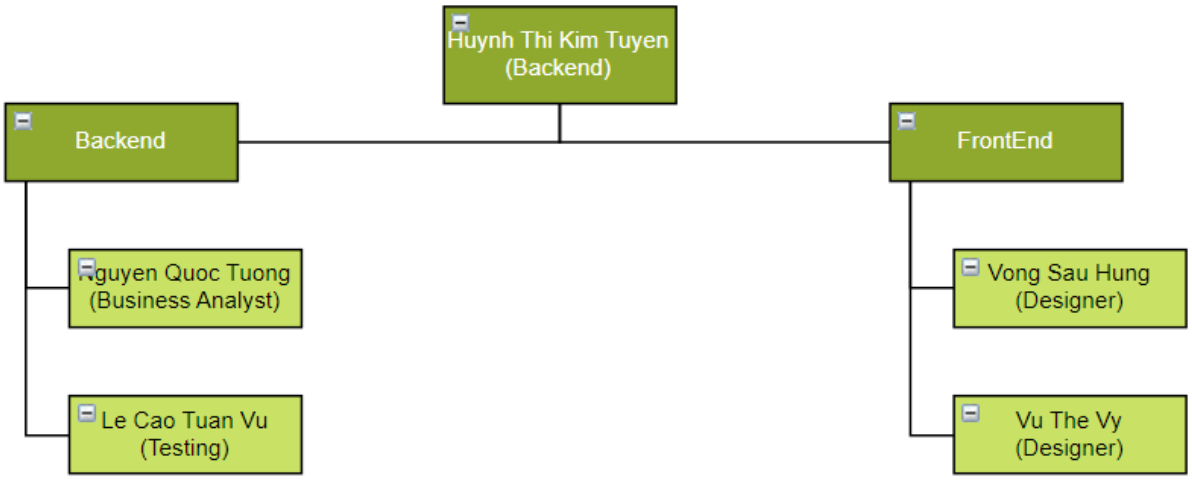
- Equipment : Every member needs to have a personal computer
- Schedule : The system is used for educational purpose which is a project of Introduction to Software Engineering course. Since the deadline is at the end of December, the system must be fully functional by this date.

2.3 Project Deliverables

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

3. Project Organization

3.1 Organizational Structure



3.2 Roles and Responsibilities

Person	Role
Huynh Thi Kim Tuyen	Developing server logic and databases. Manage the project, allocate tasks and monitor progress.
Nguyen Quoc Tuong	Developing server logic and databases. Analyze requirements, design workflow.
Vong Sau Hung	Frontend Developer Design GUI
Le Cao Tuan Vu	Developing server logic and databases. Ensure software quality
Vu The Vy	Frontend Developer Design GUI

4. Management Process

4.1 Project Estimates

- The estimated number of features is 12, with each feature requiring about 3 hours to develop (excluding time for learning, bug fixing, and documentation), resulting in an estimated project timeline of 3 months.
- If there are any changes in the requirements, a re-estimation will be conducted.

4.2 Project Plan

4.2.1 Phase Plan

Phase	No. of Iterations	Start	End
Inception	1	Week 1	Week 5
Elaboration	1	Week 6	Week 7
Construction	2	Week 8	Week 15

Phase	Description	Deliverable
Inception	Complete the requirements List all the use-cases	Project proposal, project plan, vision document
Elaboration	Analyze the requirements from the Inception phase. Develop architectural prototype	Class Diagram, UI prototype
Construction	Implement features Write test	Completed product

4.2.2 Iteration Objectives

- Iteration 1:
  - Gather and document all requirements for the project.
  - Develop the architectural design of the system, including class diagrams and the initial user interface (UI) prototype
- Iteration 2:
  - Complete the implementation of all features.
  - Conduct testing

4.2.3 Releases

- There are expected to be 2 releases:
  - 1.0 Demo: Complete all features
  - 2.0 Beta: Fix bugs, and this will be the final product

4.2.4 Project Schedule

Tasks	Begin	Finish
Phase 1	**03/10/2024**	**27/10/2024**
Project Plan SDP	**03/10/2024**	**09/10/2021**
-Introduction	03/10/2024	09/10/2024
-Project Overview	03/10/2024	09/10/2024
-Project Organization	03/10/2021	09/10/2024
-Management Process	03/10/2024	09/10/2024
Vision Document	**10/10/2024**	**30/10/2024**

-Introduction	10/10/2024	16/10/2024
-Positioning	10/10/2024	16/10/2024
-Stakeholder and User Description	16/10/2024	30/10/2024
-Product Overview	16/10/2024	27/10/2024
Phase 2	**28/10/2024**	**03/11/2024**
-Revise the Software Architecture Document	28/10/2024	01/11/2024
-UI prototype	28/10/2024	01/11/2024
-Setup the Development Environment	01/11/2024	03/11/2024
Phase 3	**04/11/2024**	**10/01/2025**
-Finished implementing all core features	04/11/2024	29/11/2024
-Implement other remaining features	29/11/2024	7/12/2024
-Evaluate and review the features	8/12/2024	14/12/2024
-Create Test plan	15/12/2024	18/12/2024
-Develop Test Cases	19/12/2024	20/12/2024
-Conduct test	21/12/2024	23/12/2025
-First Release (Demo)	24/12/2025	25/12/2025
-Detect and fix bugs	26/12/2025	28/12/2025
-Construction wrap-up meeting	29/12/2025	30/12/2025
-Final Release (Beta)	01/01/2025	02/01/2025

4.2.5 Project Resourcing

- We need traning how to apply Spring Framework on our project and some suitable Design Patterns.

4.3 Project Monitoring and Control

4.3.1 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.

4.3.2 Reporting and Measurement

Updated cost and schedule estimates, and metrics summary reports, will be generated at the end of each iteration.

The Minimal Set of Metrics, as described in the RUP Guidelines: Metrics, will be gathered on a weekly basis. These include:

Earned value for completed tasks. This is used to re-estimate the schedule and budget for the remainder of the project, and/or to identify need for scope changes.

Total defects open and closed – shown as a trend graph. This is used to help estimate the effort remaining to correct defects.

Acceptance test cases passing – shown as a trend graph. This is used to demonstrate progress to stakeholders.

In addition, overall costs will be monitored against the project budget.

4.3.3 Risk Management

Risk Ranking (High, Medium, Low)	Risk Description and Impact	Mitigation Strategy and/or Contingency Plan
High	- Group members have other course projects, which could lead to delays in meeting assigned deadlines. **- Impact level:** Very serious.	- If it seems that meeting the deadline is not possible, notify the group in advance so that contingency plans can be made. - There will also be support from other group members.
High	- Lack of project management skills, such as using Git flow or handling conflicts when working with Git. **- Impact level**: Can be overlooked.	- Resolve conflicts together as a team. - Improve project management skills. - Establish rules for everyone to follow when working on the project.
Medium	- Group members lack knowledge in FrontEnd and BackEnd, and are slow in approaching technologies, etc. **- Impact level:** Very serious.	- Experienced members in the group will guide and provide direction, as well as share reference materials. - Extend the learning time and reconsider the project to match the capabilities of the members.
Medium	- The project does not meet all the course requirements, including functionality, software documentation, and evaluation criteria. **- Impact level:** Very serious.	- Maintain regular communication with the course instructor to clearly understand the requirements. - Hold regular group meetings to discuss plans and direction.
Low	- A member may drop out or leave the course halfway. **- Impact level**: serious.	- Reduce the project's functionality to fit the remaining members' capacity. - Plan meetings to increase from 1 session per week to 2 sessions per week to ensure work quality.
Low	- Disagreements among group members. **- Impact level:** serious.	- The team leader will gather and organize group meetings to analyze everyone's opinions and decide on a common direction. - Always maintain a positive atmosphere, avoiding putting pressure on group members.

4.3.4 Configuration Management

Appropriate tools will be selected which provide a database of Change Requests and a controlled versioned repository of project artifacts.

All source code, test scripts, and data files are included in baselines. Documentation related to the source code is also included in the baseline, such as design documentation. All customer deliverable artifacts are included in the final baseline of the iteration, including executables.