Software Project

Management Plan

Group 2

CompareCart

Software Engineering
Fall 2024

CS673 A1

Siming Zhao
Zichen Wang
Yifei Wang
Jingwei Ma
Sifat Singh Khalsa

Software Project Management Plan (SPMP)

1. Purpose
The SPMP outlines the approach, responsibilities, and timeline for managing and delivering the project. This document ensures that the project is executed efficiently with clear guidelines on task delegation, timelines, resource management, risk mitigation, and communication.

2. Project Overview
The project aims to develop a web-based application, with detailed planning and tracking of activities to ensure successful delivery. The following sections describe the scope, team roles, project schedules, and risk management strategies for the project.

3. Project Scope
The scope of this project includes:

- Design and implementation of the web-app and its features.
- Development of user and technical documentation.
- Testing, deployment, and continuous improvement of the web-app.
- Effective collaboration and version control using Git.

4. Project Organization

4.1 Roles and Responsibilities

- **Project Manager**: Oversees the project schedule, coordinates resources, and ensures that the project remains on track.
- **Development Team**: Responsible for coding, unit testing, and technical documentation.
- **Quality Assurance (QA) Team**: Ensures all deliverables meet quality standards through testing and review.
- **Product Owner**: Reviews project features, provides feedback, and signs off on project milestones.
- **Stakeholders**: Provide feedback on project goals and features, helping guide the direction of the project.

4.2 Communication Plan

- **Git**: Used for source code management and collaboration.
- **WhatsApp**: Facilitates immediate communication among team members.
- **Meetings**: Weekly status meetings will be held to review progress, discuss blockers, and realign priorities.

5. Work Breakdown Structure (WBS)

- **Task 1**: Requirements gathering and documentation.
- **Task 2**: Design architecture and database structure.
- **Task 3**: Development of web-app (divided into modules).
- **Task 4**: Testing and quality assurance.
- **Task 5**: User documentation.
- **Task 6**: Build and release management.
- **Task 7**: Project closure and final deliverables.

6. Schedule and Milestones

6.1 Milestones

- **M1**: Requirements finalized and documented.
- **M2**: Design phase completion.
- **M3**: First functional prototype ready.
- **M4**: Initial testing completed.
- **M5**: User documentation prepared.
- **M6**: Final build and release approved.
- **M7**: Project closure.

6.2 Schedule

- A detailed Gantt chart will be used to outline the timeline for the tasks mentioned in the WBS. Regular progress reviews will be conducted to ensure adherence to deadlines.

7. Resource Management

7.1 Team Allocation

- Team members will be allocated based on their expertise to each module of the project.
- The project lead will ensure that resources are distributed effectively to prevent any bottlenecks in development.

7.2 Tools

- **Git**: Code versioning and collaboration.
- **Google Drive**: Documentation management.

8. Risk Management

8.1 Risk Identification

Risks identified in the SCMP include:

- Developmental delays due to technical complexity.
- Unintended interactions between modules.
- Team member unavailability.
- Data loss.
- Inefficient development processes.
- Lack of experience among team members.

8.2 Risk Mitigation

- **Developmental delays**: Early identification of technical challenges and proactive research will minimize delays.
- **Interaction effect**: Modular programming will be emphasized to reduce dependencies and interaction issues between modules.
- **Team member unavailability**: Contingency plans will include flexible schedules and backup resources.
- **Data Loss**: Frequent backups will be taken, and access controls will be enforced.

- **Inefficient development process**: Clear division of responsibilities, small working groups, and regular check-ins will improve communication and efficiency.
- **Lack of experience**: Provide learning resources such as tutorials, mentorship, and team collaboration opportunities.

9. Monitoring and Control

9.1 Progress Tracking

- Progress will be tracked via Jira, with task assignment and ticket updates ensuring visibility for all team members.
- Weekly status meetings will monitor progress and address any potential blockers.

9.2 Quality Control

- The QA team will review each build to ensure that the application meets the predefined quality standards.
- Pull requests will be used for code review and quality assurance before merging.

10. Reviews and Approvals

- Regular code reviews will be conducted before merging changes into the main branch.
- Each release will undergo a final review by the product owner and the QA team before being approved for production.

11. Project Closure

Upon successful completion of the project, a retrospective will be conducted to evaluate the project's outcomes, identify lessons learned, and archive the project documentation.