
Morello

Morello
Software Development Plan (Small Project)

Version 1.0

Morello	Version: 1.0
Software Development Plan (Small Project)	Date: 04/11/2023
<document identifier>	

Revision History

Date	Version	Description	Author
04/11/2023	1.0	First draft	Morello

Morello	Version: 1.0
Software Development Plan (Small Project)	Date: 04/11/2023
<document identifier>	

Table of Contents

1. Introduction	4
1.1 Purpose	4
1.2 Scope	4
1.3 Overview	4
2. Project Overview	4
2.1 Project Purpose, Scope, and Objectives	4
Purpose	4
Scope	4
Objectives	4
2.2 Assumptions and Constraints	4
2.3 Project Deliverables	4
3. Project Organization	5
3.1 Organizational Structure	5
3.2 Roles and Responsibilities	5
4. Management Process	6
4.1 Project Estimates	6
4.2 Project Plan	6
4.2.1 Phase Plan	6
4.2.2 Iteration Objectives	6
4.2.3 Releases	7
4.2.4 Project Schedule	7
4.2.5 Project Resourcing	7
4.3 Project Monitoring and Control	8
4.3.1 Requirements Management	8
4.3.2 Reporting and Measurement	8
4.3.3 Risk Management	8
4.3.4 Configuration Management	9

Morello	Version: 1.0
Software Development Plan (Small Project)	Date: 04/11/2023
<document identifier>	

Software Development Plan (Small Project)

1. Introduction

1.1 Purpose

This *Software Development Plan* defines the plan for developing project Morello, in particular: the constraints (budget, number of members, development duration), the role and responsibilities of every member, project phases and iterations, and risk management plan.

1.2 Scope

This *Software Development Plan* describes the overall plan across the development duration of project Morello.

1.3 Overview

This *Software Development Plan* contains the following information:

Project Overview — briefs about the purpose, scope, and objectives of project Morello. It also defines the deliverables that the project is expected to deliver.

Project Organization — describes the organizational structure of the project team and the role/responsibilities of each member.

Management Process — describes the planned schedule of project Morello, including the phases and iterations. Also identifies the risks and the plan to manage them.

2. Project Overview

2.1 Project Purpose, Scope, and Objectives

Purpose

- The project aims to solve the difficulty arising from managing funds of small local groups.

Scope

- We mainly aim to solve the management of the funds of a classroom or small local organization where members of the group can meet face-to-face on a regular basis.

Objectives

- Intuitive UI where normal members can use the app at first glance.
- Effective management of funds for high-level members such as group owner and moderator.
- Transparency over the cash flow within a fund.

2.2 Assumptions and Constraints

- The project lasts for a fixed period of 12 weeks.
- There are 5 members working on the project without any modifications throughout its lifetime.
- The project is zero-budget.
- No special types of equipment are required for the project other than personal computers that can connect to the Internet.

2.3 Project Deliverables

- Use-case diagrams: Nov 27, 2023
- Mobile application UI design: Nov 27, 2023
- Development schedule: Nov 6, 2023
- Data model diagram: Nov 27, 2023
- Project Plan (this document): Nov 6, 2023

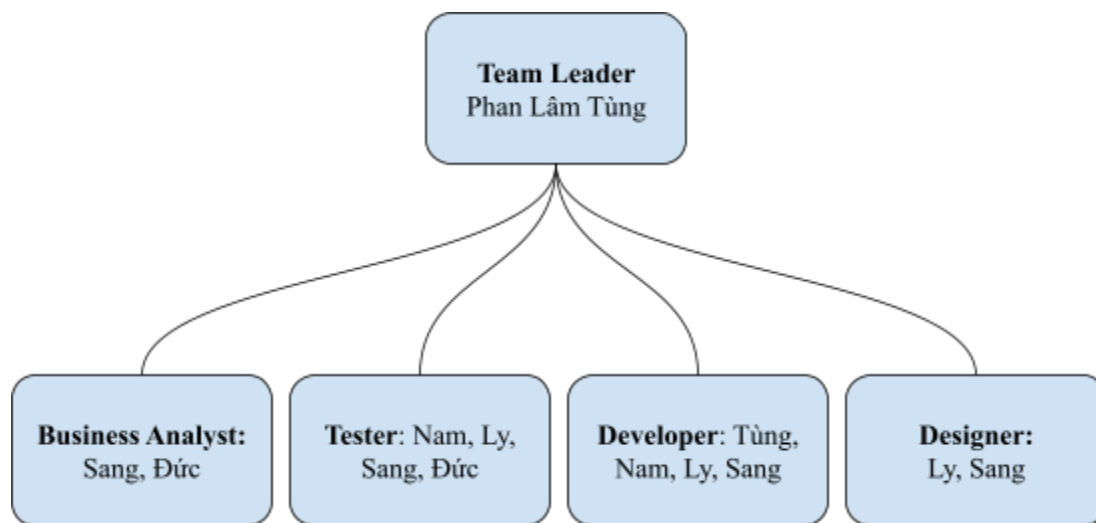
Morello	Version: 1.0
Software Development Plan (Small Project)	Date: 04/11/2023
<document identifier>	

- Project Vision: Nov 6, 2023
- Demo executable: Dec 18, 2023
- Release executable: Jan 3, 2024
- Tests suite: Dec 18, 2023 `

3. Project Organization

3.1 Organizational Structure

There are 5 roles in this project: team leader, business analyst, tester, developer and designer. The team leader will have the responsibility of supervising the whole development process.



3.2 Roles and Responsibilities

Person	Role
Phan Lam Tung, Senior Engineer Manager, Principal Software Engineer	Responsible for supervising the development process, which includes: tasks management and assignments, process tracking and monitoring. May provide assistance to other roles.
Nguyen Nhu Hoai Nam, Developer, Tester	Responsible for researching, implementing and testing new features. Also responsible for analyzing feature requirements and giving help where I can.
Nguyen Thanh Thao Ly, Developer, Tester, Designer	Responsible for designing the user interface, implementing, and testing them. I also breathe life into ideas and functionalities. May provide assistance to other developers/testers.
Pham Van Tan Sang, Developer, Tester, Business Analyst, Designer	Responsible for designing, developing, and testing the software and managing databases. Also responsible for gathering requirements and ensuring our software aligns with business goals and user needs.

Morello	Version: 1.0
Software Development Plan (Small Project)	Date: 04/11/2023
<document identifier>	

Hoang Thien Duc, Business Analyst, Tester	Responsible for collecting user requirements, doing requirement analysis, as well as managing the requirements. Also responsible for testing software components and user interfaces.
--	---

4. Management Process

4.1 Project Estimates

4.2 Project Plan

4.2.1 Phase Plan

Phase	Description	Start date	End date
Inception	Project planning, basic project vision.	Oct 30, 2023	Nov 6, 2023
Elaboration	User stories, use-case diagram, UI design, data model design, API documentation, software architecture, refining project vision and project development plan	Nov 7, 2023	Nov 27, 2023
Construction	Front-end and back-end basic functionality. Demo	Nov 28, 2023	Dec 18, 2023
Transition	Testing and preparing the software for production.	Dec 19, 2023	Jan 2, 2024

4.2.2 Iteration Objectives

There will be 6 iterations/sprints, corresponding to each PA:

Iterations/ Sprints	Objectives	Start date	End date
PA0	Finish the Project Proposal. Set up necessary tools for development.	Oct 9, 2023	Oct 22, 2023
PA1	Draft out the Project Plan and Vision Document.	Oct 23, 2023	Nov 5, 2023
PA2	Finish the Project Plan and Vision Document. Sketch out Use-Case Model, Use-Case Specs.	Nov 6, 2023	Nov 19, 2023
PA3	Draft out the Software Architecture Document. Finish Use-Case Model, Use-Case Specs.	Nov 20, 2023	Dec 3, 2023
PA4	Revise Software Architecture Document. Draw Deployment Diagram in UML. Create UI Prototype.	Dec 4, 2023	Dec 17, 2023
PA5	Prepare and finish Test Plan, Test Report, and Project Presentation.	Dec 18, 2023	Dec 31, 2023

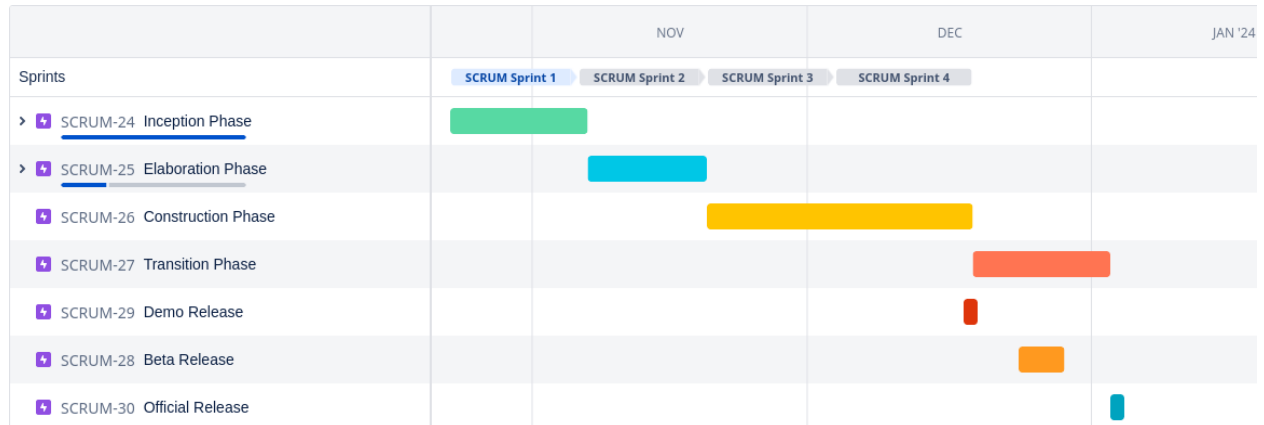
4.2.3 Releases

- At the end of the Construction phase, a demo of the software will be ready (Dec 18, 2023).

Morello	Version: 1.0
Software Development Plan (Small Project)	Date: 04/11/2023
<document identifier>	

- During the Transition phase, a beta version of the software will be ready (est. Dec 24 - Dec 28).
- At the end of the Transition phase, the software will be officially released (Jan 2, 2024).

4.2.4 Project Schedule



4.2.5 Project Resourcing

Required roles:

- Developers:
 - Quantity: 4 people.
 - Skills: Adept in making Android apps with Jetpack Compose and using Flask to create efficient and conformant RESTful API.
- Designers:
 - Quantity: 2 people.
 - Skills: Adept in using Figma, be able to create interfaces that are intuitive and easy to use, while also ensuring implementability.
- Testers:
 - Quantity: 4 people.
 - Skills: Be able to create system-wide testing plans and write test cases for many software components.
- Business Analysts:
 - Quantity: 2 people.
 - Skills: Be able to document in detail business requirements and communicate effectively between stakeholders and the team.
- Team Leader:
 - Quantity: 1 person.
 - Skills: Efficient in coordinating members to meet product requirements and project timelines.

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, which 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.

- Updated Cost and Schedule Estimates:
 - Ensure accurate tracking of project budget and schedule.

Morello	Version: 1.0
Software Development Plan (Small Project)	Date: 04/11/2023
<document identifier>	

- Identify the need for any scope changes or adjustments.
 - Allow for timely resource allocation and risk management.
- Key Metrics:
 - Earned Value for Completed Tasks: used to re-estimate the project schedule and budget for the remaining work.
 - Total Defects (Open and Closed): We will maintain a trend graph showcasing the number of defects found and resolved, mainly used for estimating the effort required to address any open defects and ensuring the quality of the project.
 - Acceptance Test Cases Passing: Helps to demonstrate our progress to our key stakeholders, specifically our course teacher, showing how well the project aligns with the requirements and validates that the project is on the right track to meet its goals.
- We will also diligently monitor overall project costs to ensure they align with the allocated budget, ensuring that the project stays within its financial constraints.

4.3.3 Risk Management

ID	Risk Ranking (High, Medium, Low)	Risk Description and Impact	Mitigation Strategy and/or Contingency Plan
1	High	Scope Creep: Frequent changes in project requirements.	Clearly define project scope, use change control process, and have a well-documented requirements management plan.
2	High	Budget Overrun: Insufficient budget allocation.	Regularly track project expenses, revise budget estimates, and seek additional funding if necessary.
3	High	Losing a crucial team member.	Cross-train team members, maintain thorough documentation and have contingency plans in place for succession.
4	High	Data Security Breach: Unauthorized access to sensitive data.	Implement robust security measures, encryption, and regular security audits to protect data.
5	Medium	Technology Risk: Unfamiliar technology stack leading to delays.	Conduct technology research, training, and maintain close collaboration with experienced mentors or consultants.
6	Medium	Unrealistic Deadlines: Aggressive project timelines.	Reassess project schedules, negotiate with stakeholders for more realistic deadlines, and prioritize critical features.
7	Medium	Quality Assurance: Frequent software defects and bugs, leading to deadlines not being met.	Implement robust testing procedures, use automated testing tools, and conduct code reviews.
8	Medium	The User Interface and User Experience are not friendly.	Learn about UI/UX designing and conduct user acceptance testing to make sure the product is easy to use.
9	Medium	Cybersecurity Threats: Threats like malware, hacking, or phishing.	Implement cybersecurity measures, keep software up-to-date, and conduct security assessments.
10	Low	Communication Issues: Lack of clear communication among	Implement regular meetings, use collaborative tools, and encourage open communication within the team.

Morello	Version: 1.0
Software Development Plan (Small Project)	Date: 04/11/2023
<document identifier>	

ID	Risk Ranking (High, Medium, Low)	Risk Description and Impact	Mitigation Strategy and/or Contingency Plan
		team members.	
11	Low	Natural Disasters: Events such as earthquakes, floods, or storms.	Implement data backup and recovery plans, relocate critical hardware to safer locations, and maintain emergency response procedures.
12	Low	Insufficient Documentation: Poor documentation practices.	Study and enforce documentation standards, provide training on documentation, and maintain version control for project documentation.
13	Low	Hardware Failures: Failure of critical hardware components.	Regular maintenance, redundancy, and replacement plans for critical hardware components.

4.3.4 Configuration Management

- **Jira:** Change Request and Issue Tracking
 - Purpose: Jira serves as our central hub for tracking Change Requests, issues, and project tasks.
 - Functionality: With Jira, we can log, prioritize, and monitor all Change Requests and issues related to our software development. It enables clear visibility into the progress of each request and ensures accountability.
- **Git and GitHub:** Version Control for Source Code
 - Purpose: Git and GitHub are the version control systems for managing source code.
 - Functionality: All source code is stored and versioned in Git repositories on GitHub. This allows us to keep a comprehensive history of code changes, collaborate seamlessly, and confidently manage different code branches and releases.
- **Google Drive:** Repository for Project Artifacts
 - Purpose: Google Drive provides a controlled repository for project artifacts, including documentation and non-code files.
 - Functionality: We use Google Drive to store and organize all non-code project artifacts, such as design documentation, user manuals, and specifications.
- **Figma:** Collaborative Design
 - Purpose: Figma is our platform for creating and maintaining design documentation.
 - Functionality: Design documentation is essential for our user interface and experience design. Figma facilitates collaborative design work and document versioning, ensuring consistency in our product's visual elements.