
Morello

**Morello
Vision Document**

Version 1.0

Morello	Version: 1.0
Vision Document	Date: 04/11/2023
<document identifier>	

Revision History

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

Morello	Version: 1.0
Vision Document	Date: 04/11/2023
<document identifier>	

Table of Contents

1.	Introduction	3
1.1	References	3
2.	Positioning	3
2.1	Problem Statement	3
2.2	Product Position Statement	3
3.	Stakeholder and User Descriptions	3
3.1	Stakeholder Summary	3
3.2	User Summary	3
3.3	User Environment	3
3.4	Summary of Key Stakeholder or User Needs	3
3.5	Alternatives and Competition	3
4.	Product Overview	3
4.1	Product Perspective	3
4.2	Assumptions and Dependencies	3
5.	Product Features	3
6.	Other Product Requirements	3

Morello	Version: 1.0
Vision Document	Date: 04/11/2023
<document identifier>	

Vision (Small Project)

1. Introduction

In Vietnam, many school classes handle their funds in a traditional way, usually with a class treasurer, who keeps track of the money. They often do this using a pen-and-paper method, which is not an ideal solution since it is very tedious when changes need to be made, an example is when someone switches to a different group. Moreover, this approach is prone to error because most of the calculations are manually calculated.

To address this problem, we want to create a mobile app that helps fund managers to easily keep track of members' payment status, while at the same time allowing class members to see how the money is being spent. Not only that, this app can also be helpful for small to medium groups that face the same challenges in general.

2. Positioning

2.1 Problem Statement

The problem of	Track payment status for each member. Update payment records. Monitor the fund's money flow. Send timely reminders to members to complete their payments. Ensure members are informed about fund usage.
affects	Students in a class, or in general, members in small to medium groups.
the impact of which is	Fund managers can easily make mistakes, which can lead to inconsistency between the actual funds and the fund records. Members have little information about how the fund was used. Complications when trying to update the group's members. Members are not reminded of payment dates, which may lead to even more complications since the manager has to temporarily compensate for the late payment.
a successful solution would be	Allowing fund admins to manage their funds through a mobile-first interface that prioritizes simplicity and effectiveness to ensure easy fund management. Enhance the UX for both the managers and the members.

2.2 Product Position Statement

For	fund members.
Who	wants to manage a fund effectively and transparently.
The (product name)	Morello - Fund Managing App
That	has an intuitive and user-friendly interface with minimal installation; allows efficient collection of funds where all payments are recorded in the balance sheet; the app also has the ability to notify users of incoming payments.
Unlike	other pen-and-paper methods.

Morello	Version: 1.0
Vision Document	Date: 04/11/2023
<document identifier>	

Our product	provides a seamless interface that operates on any Android mobile device. Requires minimal installation and is easy to use, in contrast to other tedious and error-prone methods.
-------------	---

3. Stakeholder and User Descriptions

3.1 Stakeholder Summary

Name	Description	Responsibilities
Lecturers and Teaching Assistants	The people who teach and manage students in the CS300 course.	<ul style="list-style-type: none"> - Hand out lecture slides, homework, projects, ... in the CS300 course. - Test students' knowledge through examinations. - Assess students' projects and examine results.
Students	The people who attend the CS300 course to acquire knowledge and credits to graduate.	<ul style="list-style-type: none"> - Submit homework and Project Assignments (PAs). - Take part in examinations. - Attend lab sessions to report the project's progress.

3.2 User Summary

Name	Description	Responsibilities	Stakeholder
Group Leader	They can create and manage groups with moderators and members.	<ul style="list-style-type: none"> - Responsible for the actual fund of the group. - Issue <i>income</i> requests. - Record <i>expense</i> entries. - Collect and verify members' payments for an <i>income</i>. 	
Group Moderators (fund collectors)	They help admins in collecting and verifying members' payments.	<ul style="list-style-type: none"> - Collect and verify members' payments for an <i>income</i>. 	
Members	They can view all the activities in the group (income/expenses) and the group's balance.	<ul style="list-style-type: none"> - Complete income requests. - Help identify inconsistencies in the group's log. 	
System admins	They help aid in customer support.	<ul style="list-style-type: none"> - Resolve issues regarding special cases with users' accounts. - Monitor and report irregular activities in the app. - Handle users' feedback and feature requests. 	

3.3 User Environment

Morello	Version: 1.0
Vision Document	Date: 04/11/2023
<document identifier>	

Task	Number of people	Task Cycle Duration	Duration	Environmental Constraints
Create Group Leader Account	1 (Group Leader)	Once	3-5 minutes	Online Mobile
Create Moderators' Accounts	1 (Moderator)	Once	3-5 minutes	Online Mobile
View members list	1	Occasionally	1-3 minutes	Mobile
View fees	1	Regularly ~Once per day	1-3 minutes	Mobile
Remind payments (Notifications)	Depends on group size	On demand or Once a week	< 1 minutes	In person Online Mobile
Money collecting	3-5 (Moderators)	On-demand (when there's an income entry)	< 1 minute	In person (cash) Online Mobile (e-wallets)
Import member data	1 (Group Leader)	On demand	3-5 minutes (if the students list is available)	Online Mobile
Record description for income/expense entries	1 (Group Leader)	On demand (every time an income/expense entry is added or edited)	< 2 minutes	Online Mobile
View members that are overdue	1 (Group Leader/Moderator)	Regular	< 1 minute	Online Mobile
Update members' payment status	1 (Group Leader/Moderator)	Regular (every time a Moderator collects money from members)	< 1 minute	Online Mobile
View moderators list	1 (Group Leader)	On demand	< 1 minute	Mobile
Add/remove members	1 (Group Leader)	On demand ~ Rarely	< 2 minutes	Online Mobile
Add/remove moderator	1 (Group Leader)	On demand ~ Rarely	< 2 minutes	Online Mobile

Morello	Version: 1.0
Vision Document	Date: 04/11/2023
<document identifier>	

3.4 Summary of Key Stakeholder or User Needs

Need	Priority	Concerns	Current Solution	Proposed Solutions
Remind members to pay	High	Users expect that there is a way for them to remind others to pay.	Either by telling them in person, or through an online medium like Discord.	The app will send notifications to the members' mobile phones
Record discounts, and also the according expenses of each person after receiving the discount	Medium	Users expect the discounts should be evenly distributed between the members, without the tediousness of calculating it.	Calculate by hand, or by using a calculator, then send the messages to others via an online medium such as Discord	Automated using a feature in the app
Record expenses and income entries for the whole group.	High	Users expect that there is a way to see all payment records in the past.	Mostly rely entirely on the leader's or members' memory, don't record at all.	Automated using a feature in the app

3.5 Alternatives and Competition

- **Pen-and-paper method:** This method is traditional and is widely used, typically by class moderators, to manage their class's fund. However, because this method is operated entirely by humans, it is prone to calculation errors, missed deadlines, and countless other mistakes that could be avoided through automation. Transparency is also a concern, as papers holding the records are typically kept only by a few moderators.
- **Spreadsheets (Excel, Google Sheets):** This method allows advanced customizations that cater to the user's specific needs. Spreadsheets from online services (Excel, Google Sheets) can also be shared with all members to view or edit. However, because spreadsheets are a general-purpose tool, they can be very difficult to use. The user should be highly efficient in Excel or Google Sheets to utilize its full potential.
- **Splitwise - app to track shared expenses:** Splitwise allows users to join groups, split bills among group members, view group expenses, and track the amount owed between members. Friendly UI/UX gains Splitwise an edge over its competitors. However, Splitwise's functionality is quite limited. *Firstly*, the app requires an expense to exist before any income can be recorded. This is inconvenient in the case of a class fund, when its purpose (such as buying chalk, emergency needs or other supplies) is not immediately apparent at the time the money is collected. *Secondly*, this app lacks support for admin/moderator privileges, which might be problematic for large groups (such as a class). *Thirdly*, Splitwise requires every member to sign up with an email address formally. The process is, at best, cumbersome for its users, and at worst, an unnecessary privacy concern.

4. Product Overview

4.1 Product Perspective

- This product is independent; its database is going to be created by our own group.
- We plan on developing this app to replace certain existing systems such as Excel, Google Sheets or apps such as Splitwise when it comes to funding problems.

4.2 Assumptions and Dependencies

- **Assumption:**
 - The product is run on an Android device.

Morello	Version: 1.0
Vision Document	Date: 04/11/2023
<document identifier>	

- **Dependencies:**
 - The product obeys the regulations in a country and the moral rules.

5. Product Features

Features	Description
Create a group	The Group Leader can create a group and add moderators to that group. The Group Leader can add members to that group. Each member is identified with a unique name which is set by the group leader.
Manage balance records	The Group Leader/Moderators can add/delete/edit income/expense/bill records, including the record's name, description, deadline, and involving members.
Manage the payment status of members	The Group Leader/Moderators can view the payment status of every member. The status could be <i>paid/not paid/late</i> . The Group Leader/Moderators can change a member's status from <i>not paid</i> to <i>paid</i> and vice versa.
View his/her payment status	A member can view their most recent payments, including essential information like payment description and payment deadlines
Subscribe to a member	A device can subscribe to some members to receive notifications for that member.
Send feedback to the system admin	A registered member can send feedback to the system admin, either to report a bug or to give necessary
View balance and record history of the group	A member can view the current balance of the group. Including the descriptions of each income or expense entry.
Notifications of new records	A member will receive notifications when the group leader/moderators create a new income/expense/bill record.
Notifications to remind members of coming/late payments	A member will receive notifications when a payment deadline is coming, or when their payment is late. The Group Leader/Moderators can send this type of notification manually.

6. Non-Functional Requirements

- **Usability:** The app must have an intuitive interface that conforms to the Android Design Guidelines so that users can quickly and easily access all features. Documentation for users must be accurate and easy to understand.
- **Reliability:** The app must consistently and accurately display income/expense records. Data must not be lost or corrupted during regular usage. Features must function as intended.
- **Performance:** The app must respond quickly to user actions. Free of performance issues: scroll janks, high startup latency, hiccup transitions.
- **Maintainability:** The codebase must be well-organized and documented. Regular updates must not affect the app's core functionality. Bugs must be identified and fixed quickly without affecting the app's core functionality.
- **Scalability:** The app must be able to handle a growing number of users and data without violating other requirements.