Problem Statement and Goals
Software Engineering

Team #5, Money Making Mauraders
Zhenia Sigayev
Justin Ho
Thomas Wang
Michael Shi
Johnny Qu

Table 1: Revision History

Date	Developer(s)	Change
09-22	Thomas Wang	Added Section 1
09-22	Justin Ho	Problem 1.1-1.4
09-22	Zhenia Sigayev	Added Sections 2-5
...	...	...

1   Problem Statement

The engineering department currently uses spreadsheets and manual processes to track club finances. This approach often results in errors from lost reciepts, duplicated claims, and large amount of claims at the end of the year. A centralized financial tracking software system is needed to streamline and automate this process.

The software will take clear inputs such as budgets, expenses, transactions, receipts, and any other financial information. The outputs will include reports, summaries, and visual dashboards. This system will be kept simple so it is easy to use and maintain for future student groups.

Stakeholders include student group leadership teams who manage budget, administrative staff who monitor spending and issue reimbursements, and faculty advisors. The environment the software will run on is the department's existing digital infrastructure with the software being in the form of a web platform

The importance of the problem lies in improving financial tracking efficiency, reducing the reporting time and support better financial decision making for

student activities. The goals of the project are measurable: minimize errors in transaction logging and reciept submissions, encourage continuous submissions over the school year rather than a large quantity at the end of the year, and provide clear visualization that makes budgets and spending easy to interpret.

1.1 Problem

- MES reimbursment process is manual and slow to complete
- expense submissions can come from multiple places making them difficult to access
- Hard to keep track of payments and receipts

1.2 Inputs and Outputs

Inputs:

- **User Information**
  - Authentication credentials
  - User role (club member, treasurer, MES admin)
  - Club affiliation

- **Financial Data**
  - Reimbursement amount
  - Transaction date
  - Expense category/purpose
  - Payment method used

- **Documentation**
  - Digital receipts (PDF, images, emails)
  - Physical receipt scans/photos
  - Supporting documentation (event details, approval emails)

- **Budget Information**
  - Club budget allocations
  - Funding category limits

Outputs:

- **User Interface**
  - Personalized dashboards by role
  - Reimbursement request history

- Status tracking for pending requests

- **Reporting**

  - Financial summaries by club/category
  - Budget utilization metrics
  - Transaction logs for auditing
  - Exportable reports (CSV, PDF)

- **Notifications**

  - Status change alerts (submitted, approved, rejected)
  - Reminder notifications for pending actions
  - Budget threshold warnings

- **Administrative Tools**

  - Approval workflows
  - Batch processing capabilities
  - System configuration options

1.3 Stakeholders

- Group leadership teams in charge of managing MES budget

- Administrative staff in charge of monitoring spending provide reimbursements

- Faculty advisors who determine what expenses are eligible for reimbursement

1.4 Environment

Software and Technology Stack:

- **Languages:**

  - TypeScript (primary)
  - CSS

- **Runtime:**

  - Node.js

- **Database:**

  - MongoDB Community Server

- **Core Frameworks:**

- Next.js (React-based framework for server-side rendering, routing, static site generation)
- React (UI library)
- Tailwind CSS (utility-first CSS framework)
- NextUI (UI component library)

- **Major Libraries:**

  - @mui/material (Material UI for React)
  - @tanstack/react-query (Data fetching, state management)
  - axios (HTTP client)
  - next-auth (authentication)

- **Other Configuration/Plugin Files:**

  - next-sitemap (sitemap generation)
  - pnpm/yarn (dependency management)

2 Goals

- **Centralize financial submission process** — Create a single platform where all expense submissions, approvals, and tracking occur, eliminating the need for multiple submission channels.

- **Enhance budget visibility** — Provide real-time budget utilization data that is accessible to all stakeholders with appropriate permissions.

- **Reduce processing time** — Decrease the average number of actions needed to process a reimbursement request to at most 10.

- **Improve receipt management** — Achieve 100% receipt retention through digital storage, preventing lost or misplaced documentation.

- **Minimize manual data entry** — Automate at least 80% of the data entry currently performed manually by treasurers and administrators.

3 Stretch Goals

- **Mobile-friendly interface** — Create a responsive design that works seamlessly on smartphones, allowing receipt uploads directly from mobile devices.

- **Bulk processing features** — Enable administrators to process bulk reimbursement requests simultaneously to further improve efficiency.

- **Distribute submissions throughout the year** — Reduce end-of-year submission spikes by implementing reminder features that encourage regular submissions.

- **Automated receipt data extraction** — Have at least 10 MES clubs submit at least 1 reimbursement receipt using our new MES financial tracker before end of April.

- **Advanced analytics dashboard** — Provide trend analysis and forecasting tools to help clubs plan future spending based on historical data.

4   Extras

- **User Instructional Video** — Create a short instructional video demonstrating how to use the MES financial tracker, including steps for submitting expenses, tracking reimbursements, and generating reports.

- **Wireframe Report** — Develop a comprehensive wireframe report outlining the user interface and user experience design for all key pages of the MES financial tracker.

Appendix — Reflection

The purpose of reflection questions is to give you a chance to assess your own learning and that of your group as a whole, and to find ways to improve in the future. Reflection is an important part of the learning process. Reflection is also an essential component of a successful software development process.

Reflections are most interesting and useful when they're honest, even if the stories they tell are imperfect. You will be marked based on your depth of thought and analysis, and not based on the content of the reflections themselves. Thus, for full marks we encourage you to answer openly and honestly and to avoid simply writing "what you think the evaluator wants to hear."

Please answer the following questions. Some questions can be answered on the team level, but where appropriate, each team member should write their own response:

1. What went well while writing this deliverable?

   One major process that went well is that we effectively collected information from the relevant MES stakeholders very thoroughly and appropriately. As a result, we were able to clearly define:

   (a) Pain points in the problem that the MES and clubs are facing

   (b) Goals and Requirements needed in the solution to be effective

   (c) Stretch goals that would be nice-to-haves for the MES reimbursement team

2. What pain points did you experience during this deliverable, and how did you resolve them?

   One of the great pain points that our group has faced is that we did not have very strong team alignment during this deliverable, which meant we were figuring things out a bit too late in the process. We resolved this by ensuring that we had more meetings to check in with each other and make sure we were all on the same page moving forward. Additionally, the disorganized nature of not knowing where exactly the deliverable requirements were made it challenging to stay on track.

3. How did you and your team adjust the scope of your goals to ensure they are suitable for a Capstone project (not overly ambitious but also of appropriate complexity for a senior design project)?

   Our team approached scope adjustment by conducting meetings with MES developers to identify their most critical pain points. This helped us prioritize essential features over nice-to-haves. We then established clear primary goals focused on core functionality (centralized submissions, budget visibility) while designating more complex features (like automated data extraction) as stretch goals. This tiered approach ensures we deliver a complete, valuable solution even if resource constraints prevent implementing every feature. Additionally, we decided on an incremental

6

development approach with regular milestones rather than attempting a monolithic build, allowing us to demonstrate concrete progress throughout the project timeline while maintaining appropriate complexity for a senior design project. Finally, we figured out early risks such as where development would occur (e.g., in the MES monolith repo, our own own repository, which we would then submodule into the MES repo, etc.) and planned accordingly to mitigate these risks.