Hazard Analysis
MES-ERP

Team #26, Ethical Pals
Sufyan Motala
Rachid Khneisser
Housam Alamour
Omar Muhammad
Taaha Atif

Table 1: Revision History

Date	**Developer(s)**	**Change**
October 15, 2024	Omar Muhammad	Introduction, scope and system boundaries
October 18, 2024	Housam Alamour	Critical Assumptions, started on FMEA
October 25, 2024	Sufyan Motala	Safety and Security Requirements and reflection

Contents

1 Introduction

A hazard is defined as a condition or event that can result in harm, failure, or an undesirable outcome in a system. For engineers, hazards are risks that can affect the safety of a system, its functionality, or its operational integrity. For this project, a possible solution discussed was a web based app for phones, laptops, and computers. Specifically, in our project, a hazard can be identified as something that would degrade the operational integrity of the system, cause a crash in our system, expose a user to information that should not be available to them, or grant a user access to something they shouldn't have access to.

2 Scope and Purpose of Hazard Analysis

The scope of the hazard analysis will include any potential risk that will degrade operational integrity, cause crashes, or give users unauthorized access/information to system functionalities. The losses incurred from the listed scope would include unauthorized access to sensitive data, or exposure to restricted information. This risk could lead to a failure to maintain accurate data of the reimbursement requests. For example, if the backend of the app is edited without proper authorization, it will cause the McMaster Engineering Society to have false information and could make them miss out on reimbursement requests, which lowers the integrity of the app. A final loss would be degraded system functionality or complete system failure if the app cannot handle various inputs from users. If these risks occur frequently, the user base of the application will not be happy and we will risk them not wanting to use the app. By conducting this hazard analysis, we intend to limit all these risks, by limiting the risks we ensure the users of the application will remain content with it and will continue to use the app.

3 System Boundaries and Components

- **Component: Database**
  - Hazards:
    1. Unauthorized access leading to data breaches.
    2. Data corruption or loss during operations or updates.
    3. Insufficient backups causing data loss.
- **Component: Front End**
  - Hazards:
    1. Unhandled inputs leading to system crashes.
    2. Poor user experience with performance speed.
    3. Poor user experience from a lack of system feedback.

4. Browser compatibility issues causing incorrect display of content.
5. Browser compatibility issues causing crashes.

- **Component: Back End**
  - Hazards:
    1. System crashes due to unhandled exceptions or inputs.
    2. Failure to process data correctly, leading to incorrect outputs.
    3. Poor security implementation exposing APIs.
- **Component: Hardware/Server**
  - Hazards:
    1. Power failures or hardware malfunctions.
    2. Weak server side processing power causing increased latency.
    3. No potential backup server in case of emergency.
    4. Insufficient server capacity for handling peak loads.
- **Component: Authentication System**
  - Hazards:
    1. Unauthorized access due to weak authentication protocols.
    2. Mismanagement of user roles leading to incorrect access control.
- **Component: Reimbursement System (Input/Output)**
  - Hazards:
    1. Incorrect data submission (e.g. incorrect amounts, invalid receipts) causing delays or rejection of requests.
    2. Output discrepancies (e.g. incorrect approvals or incorrect reimbursement amounts) due to calculation or logic errors.
    3. Failure to notify the appropriate parties (clubs or administrators) regarding the status of the request, leading to confusion and delays.
    4. Data tampering during the approval process, allowing unauthorized changes to reimbursement requests.
    5. Lack of exception handling, leading to duplicate requests or other unintended circumstances.

4 Critical Assumptions

1. **Assumption 1: Reliable Internet Connectivity**
   The system assumes that all users, including student leaders, administrators, and MES staff, will have access to reliable and stable internet

connections when interacting with the platform. This assumption is important because the platform is designed to work through a web-based interface. This will require real-time data processing. Reliable internet connectivity is necessary for users to access features like submitting reimbursement requests, tracking the status of payments, and viewing financial reports without interruption. If the connection is unstable, users might experience delays or errors during data submission, which may lead to incomplete reimbursement requests or user frustration. Also, poor connectivity could stop the delivery of notifications, meaning users will not get critical notifications and updates. Addressing potential connectivity issues early through notifications in the interface, error-handling when there is no connection and potentially providing offline access options could mitigate risks associated with this assumption.

2. **Assumption 2: Server Availability and Performance**
   It is assumed that the McMaster Engineering Society (MES) will provide a dedicated server with sufficient resources, including CPU power, memory, and storage, to support the platform's operations. The server must handle the anticipated workload, real-time updates, and concurrent access by all administrator of the reimbursement program of the MES as well as administrators of the clubs during peak periods. The server's storage capacity should be enough to store financial data for at least 3 years. This will allow for comprehensive audit trails and historical data retrieval. This assumption is important because if the server cannot handle the project, this could result in slower response times, data processing delays, or system crashes. This would directly impact user experience and the efficiency of financial operations. If these are not met or if future demand necessitates, then there may be a need to upgrade, potentially increasing project costs and delaying deployment. Regular performance monitoring should be planned to ensure server capacity aligns with user demand over time.

3. **Assumption 3: User Compliance with Data Entry Standards**
   It is assumed that all users of the platform, including student group leaders and MES staff, will adhere to the established data entry standards when submitting information into the system. This includes entering accurate details for each reimbursement request, such as the amount, purpose, and necessary attachments like receipts. Sticking to these standards ensures that the data processed by the platform is accurate and complete, minimizing errors during financial reviews and report generation. If users do not comply with these standards, there is a risk of data entry errors leading to rejected requests, delays in processing, and additional administrative workload to correct mistakes. Proper user training and clear guidelines should be implemented so all users comply. Additionally, the platform should include input validation and error-checking features to reduce the likelihood of incorrect data entries.

4. **Assumption 4: User Access to Compatible Devices and Browsers**
   It is assumed that all users, including student group leaders, administrators, group members and MES staff, will have access to devices and browsers that are compatible with the new platform. The platform is designed to function optimally on modern operating systems, as well as on commonly used browsers. This assumption is critical because the system's user interface and performance rely on up-to-date browser features and operating system support to ensure a smooth user experience. If users attempt to access the platform on outdated devices or unsupported browsers, they may experience issues such as reduced functionality, slower response times, or display errors. To mitigate this risk, it will be essential to clearly communicate the platform's minimum system requirements to users and provide support for upgrading or accessing compatible devices where possible.

5 Failure Mode and Effect Analysis (FMEA)

TO BE EDITED

Failure Mode	**Cause**	**Effect**	**S**	**O**	**D**	**RPN**	**Recommended Actions**
Data loss during migration	Incorrect data mapping	Loss of historical records, audit issues	8	4	6	192	Use validation scripts, trial migrations, ensure backups.
Unauthorized access	Weak passwords or vulnerabilities	Data breach, legal issues	9	3	5	135	Strong passwords, security audits, multi-factor auth.
System downtime	Server overload	Submission delays, processing impact	7	5	4	140	Increase capacity, load balance, monitor performance.
Incorrect approvals	Manual errors	Financial discrepancies, dissatisfaction	6	4	7	168	Automate approvals, training, add review layer.

6 Safety and Security Requirements

1. There must be comprehensive logging and monitoring of access to reimbursement information, including tracking actions such as data access, modification, and deletion.
   Rationale: Monitoring access helps in identifying and responding to unauthorized access attempts or suspicious activities, providing an audit trail that enhances data security.

2. There must be role based access control to restrict access to reimbursement and financial data.
   Rationale: Only authorized personnel, such as administrators and designated financial officers, should have access to sensitive reimbursement information. This control ensures confidentiality and limits access to sensitive financial information.

3. There must be data encryption for sensitive information such as user credentials, financial data, and reimbursement requests.
   Rationale: Encryption protects sensitive data from unauthorized access and ensures that data is secure during transmission and storage.

7 Roadmap

7.1 Immediate Implementation

The following safety requirements will be implemented during the capstone project to ensure a functional and secure platform:

1. **Input Validation and Error Handling**
   - Ensure that user inputs such as reimbursement amounts and receipts are validated to prevent submission errors and system crashes.
2. **Authentication and Access Control**
   - Implement login system with role-based access control to restrict sensitive financial data and operations.
3. **Data Encryption and Security**
   - Encrypt sensitive data and ensure secure connection between client and server to protect against unauthorized access.
4. **Audit Logs**
   - Set up audit logs to track user actions such as submissions, approvals, and modifications.
5. **Server Monitoring and Reliability**
   - Implement server monitoring and backups to ensure system availability and prevent data loss.

7.2 Deferred for Future Implementation

The following will be considered for future updates beyond the capstone project timeline:

1. **Multi-factor Authentication**

- Future versions will consider adding MFA to add additional account security.

2. **Advanced Fraud Detection**

   - Implement machine learning algorithms to detect potentially fraudulent reimbursement requests.

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?
   Sufyan: What went well for me during this deliverable is that we were thorough enough in our SRS that I did not have much extra needed to be added to section 6 of the Hazard Analysis.
   Omar: What went well for me during this deliverable was the clarity towards the answers we needed to do. It was very clear what a hazard analysis document should include, that plus the help from the TA made it very easy to know all the potential hazards are project may include.
   Taaha: what went well while writing the deliverable was that all the questions were understandable and were able to be completed with minimal help from TA.
   Rachid: What went well for me this deliverable was that the hazard analysis questions were extremely clear. The SRS allowed me to have a strong understanding of the requirements of the project which allowed me to easily answer all the questions. **Housam**: I found it helpful to look back at the SRS, which provided a solid foundation for defining clear, relevant assumptions related to the project.

2. What pain points did you experience during this deliverable, and how did you resolve them?
   Sufyan: I did not have any pain points during this deliverable.
   Omar: A pain point I had for this deliverable was thinking of the scope of the hazard document, I was not too sure if I needed to include certain elements (building safety) for our project, but after some clarification with the TA it was easy to know.
   Taaha: No pain points during deliverable.
   Rachid: I had no pain points working on this deliverable. **Housam**: Setting boundaries for assumptions without repeating parts or being redundant was sometimes hard. I resolved this by reviewing examples and communicating with the team to keep each assumption focused.

3. Which of your listed risks had your team thought of before this deliverable, and which did you think of while doing this deliverable? For the latter ones (ones you thought of while doing the Hazard Analysis), how did they come about?
   Team: We did not give much thought to the confidentiality of the data nor how roles should have varying levels of access. We thought we had that all covered until our meeting with the TA where this discussion was brought to light.

4. Other than the risk of physical harm (some projects may not have any appreciable risks of this form), list at least 2 other types of risk in software products. Why are they important to consider?
   Team: One is data security risk. Sensitive data especially financial information in our case should not be leaked to the wrong people. There could be legal consequences if this were to happen. Another is system downtime. If the system is down, users will not be able to submit reimbursement requests, delays in financial processing will occur, and this will lead to user dissatisfaction.