Hazard Analysis
Farming Matters

Team #14, The Farmers
Brandon Duong
Andrew Balmakund
Mihail Serafimovski
Mohammad Harun
Namit Chopra

Table 1: Revision History

Date	**Developer(s)**	**Change**
10/19/2022	Namit Chopra, Brandon Duong Andrew Balmakund, Mohammad Harun Mihail Serafimovski	Finished first version
04/03/2023	Brandon Duong	Failure modes for Database more specific, changed the scope to not include hardware or game logic, added severity column to FMEA, added rational to requirements

Contents

1 Introduction **1**

2 Scope and Purpose of Hazard Analysis **1**

3 System Boundaries and Components **1**

4 Critical Assumptions **2**

5 Failure Mode and Effect Analysis **2**

6 Safety and Security Requirements **3**

6.1 Security Requirements . . . 3
6.2 Access Requirements . . . 4
6.3 Integrity Requirements . . . 4
6.4 Privacy Requirements . . . 5
6.5 Audit Requirements . . . 5
6.6 Immunity Requirements . . . 5

7 Roadmap **6**

8 Appendix **7**

8.1 Symbolic Parameter Table . . . 7

List of Tables

1 Revision History . . . i
2 **FMEA Table** . . . 2
3 **Roadmap Table** . . . 6
4 **Symbolic Parameter Table** . . . 7

1 Introduction

Based on Nancy Leveson's work, a hazard is any property or condition in The Farming Matters game that fails or alters its intended function when coupled with the environment. This document outlines the Hazard Analysis for the Farming Matters game. The Farming Matters game is an engaging way to collect authentic data to support the research study that focuses on whether or not people prefer probabilistic or deterministic information.

2 Scope and Purpose of Hazard Analysis

The scope of this document is to provide an analysis regarding hazards of the different system boundaries and components, how to mitigate each hazard, and provide safety and security requirements. It is important to note, the hardware of choice in which our system is run (i.e. the user's device), is beyond our control as players will play on their unique combination of physical hardware through a web browser. Accounting for all combinations is not possible, which ideally is not necessary as the system should be developed in such a way that it generally works as long as the player's setup can run a web browser. Similarly, it is assumed that all game mechanics and functionality work as intended, as it is impractical to account for all different combinations of input. Ideally, the normal operation would result in no, or at least minor, bugs that do not affect the logged data. The VnV report will ensure this. The team has focused to mitigate failures in the following components: Authentication System, Backend Server, Database System, and User Interface

3 System Boundaries and Components

The system will be divided into the following components:

1. The application including both the frontend and backend consists of the:
    (a) Authentication System
    (b) Backend Server
    (c) Database System
    (d) User Interface
2. The physical setup (computer, keyboard, mouse, laptop)

The authentication system component is responsible for allowing users to create an account and log in as well as allowing existing users to log in. The backend server component is responsible for handling all requests regarding the login system and database system as well as responding to these requests. The database system component is responsible for the handling of user data.

4 Critical Assumptions

There are no critical assumptions.

5 Failure Mode and Effect Analysis

Table 2: **FMEA Table**

Design Functions	**Failure Modes**	**Effects of Failure**	**Causes of Failure**	**Recommended Action**	**SR**	**Ref**	**Severity**
Database	Server can not fulfill all API requests in API_RESPONSE_TIME	Can not store all user decisions, losing data necessary for the underlying research study	Too many people playing and making API requests at the same time, or server unexpectedly goes down	Ensure a queue-login system is enforced, only allow a max amount of users to play at the same time. If the server is down, do not allow players to continue playing as to not lose any data	IR1	H1-1	High
	Database can not handle all database requests in DATABASE _RESPONSE_TIME	Can not store all user decisions, losing data necessary for the underlying research study	Storage of the database is full or too many people playing and making API requests at the same time	Admin can download all data (user decisions) from the database and delete the data on the database afterward, hence creating additional storage. Admins could also increase database storage or request capacity, or only allow a max amount of users to play at the same time	IR2	H1-2	High
Authentication	Unauthorized user is able to log into the game	Logged player decisions cannot be traced to an account/user	Database authentication issue	Ensure only authorized user decisions are logged	ACR2, ACR3	H2-1	Medium
	Bots are able to play the game	Logged decisions are inauthentic and detrimental to the underlying research	Attacker develops script to automate account creation and play game	Ensure account creation includes captcha	SR1	H2-2	Medium
	Account sharing	Logged decisions do not reflect the decision-making of one person and are detrimental to the underlying research	The user shares their account login information with their peers	The user must accept the guidelines and rules before playing the game	IR4, IR5	H2-3	High
	User opening multiple sessions	Logged decisions from current and previous sessions may be overwritten and thus lose data	The user logs in multiple times on the same device or multiple devices	The user must log out before creating a new session or the system will automatically log them out of the old sessions in order to create a new session	ACR4	H2-4	High

Design Functions	**Failure Modes**	**Effects of Failure**	**Causes of Failure**	**Recommended Action**	**SR**	**Ref**	**Severity**
Internet Connectivity	Loses internet connection during gameplay	User loses all progress made since the last save before losing internet connection	Hardware is having connectivity problems	To save current progress, wait till internet access has been retrieved in order for the system to perform an automatic save. Otherwise, the game will resume at the most recent saved progress	IR3	H3-2	Low
General	Web browser or the tab unexpectedly closes	User loses all progress made since the last save	Not enough computer resources available, significant host operating system crash, accidental close of web browser or tab	Close unused applications and other web browser tabs that are unused on a host computer. Have the saving be frequent so as to not lose too much progress when this failure mode happens	IR1	H4-1	Low
	Game is slow to respond to user input	User is effectively unable to play the game.	User's hardware is insufficient to run the game	Provide a specifications guide in the to inform users what minimum specifications are required to run the game	IR1	H4-2	Low

6 Safety and Security Requirements

The following requirements include requirements in the Software Specification Document. It also lists new requirements which will be added to the Software Specification Document and have been written in **bold**.

6.1 Security Requirements

SR1. The system must not allow automation of creating accounts.
Rationale: It is important for the data collection of the study to not include robotic responses for the data to be as authentic as possible

SR2. **The system will encrypt all user passwords with a sufficient encryption algorithm.**
Rationale: It is important for the data collection of the study to not have multiple people play on the same account for the data to be as authentic as possible

6.2 Access Requirements

ACR1. **The frontend system shall allow access to any user.**
Rationale: Unauthenticated users will need to authenticate themselves

ACR2. **The backend system shall only allow unauthenticated access to login-related functionality.**
Rationale: Unauthenticated users will need to authenticate themselves

ACR3. **The backend system shall only allow access to authenticated users for all other (non-login) functionality.**
Rationale: It is important for the data collection of the study for the data to be traceable to specific user accounts to be able to inspect their tendencies

ACR4. **The backend system shall allow only up to one user to have one user logged-in session at any point and time.**
Rationale: It is important for the data collection of the study to not allow multiple concurrent sessions on the same account so as to not overwrite any data

6.3 Integrity Requirements

IR1. **The system will be able to handle all API requests in API_RESPONSE_TIME**
Rationale: From Human-Computer Interfaces, a maximum of 1 second wait time is needed for a user's flow of thought to stay uninterrupted

IR2. **The system will be able to handle all database requests in DATABASE_RESPONSE_TIME**
Rationale: From Human-Computer Interfaces, a maximum of 1 second wait time is needed for a user's flow of thought to stay uninterrupted

IR3. **The system will be able to handle the unexpected loss of connection to the server**
Rationale: It is important for the data collection of the study to not corrupt, lose, or overwrite any data

IR4. **The user shall agree to the terms and conditions before using the application**
Rationale: Users must agree to the terms and conditions of the study for the system to be approved by the ethics board

IR5. **The system shall warn users regarding account sharing and how it will skew the data collection for research**
Rationale: It is important for the data collection of the study to not have multiple people play on the same account for the data to be as authentic as possible

6.4 Privacy Requirements

PR1. **The system shall delete all user data if the user decides to opt out of data collection**
Rationale: It is important that users can opt out of the study at any time for the system to be approved by the ethics board

PR2. **The application only requires an email provided by the user**
Rationale: User must account only require an email to be approved by the ethics board

6.5 Audit Requirements

N/A

6.6 Immunity Requirements

N/A

7 Roadmap

Table 3: **Roadmap Table**

Timeline	**Requirements**	**Rationale**
POC	ACR1	In order to demonstrate the POC, the frontend must be accessible to an unauthenticated user on a device running the POC code locally
	IR1	Backend functionality will be needed for the POC, so all API requests needed for the POC should be handled properly
End of Capstone	SR1 IR5	These are needed to prevent skewing of the research data obtained in the project, as discussed with the project supervisor.
	SR2 ACR2 ACR3	A login system is needed as part of the final project in order to save user data, among other things. This includes proper encryption for passwords and backend authentication-based access.
	IR2	Database functionality will be expected in the final project, therefore all database requests should be handled properly
	IR4 PR1	These requirements must be fulfilled in order to gain approval from the Ethics board. Users must accept some terms and must be able to opt out of data collection at any time.
	PR2	In order to get approval from the Ethics board as fast as possible, the final project should collect minimal data required to make an account.
Future	IR3	To handle the loss of user connection, some type of autosave will have to be implemented. This is not part of the scope of the final project, but it is a valid concern, so it will be considered in the future.
	AUR1	Storing gameplay statistics further than user decisions would be useful, but is not part of the data needed for the core project and may complicate Ethics board approval. Therefore it will be considered in the future.

8 Appendix

8.1 Symbolic Parameter Table

Table 4: **Symbolic Parameter Table**

Symbolic Parameter	**Description**	**Value**
API_RESPONSE_TIME	The maximum amount of time allowed for the system to respond to the API request	0.5 seconds
DATABASE_RESPONSE_TIME	The maximum amount of time allowed for the system to respond to the database request	0.25 seconds