Hazard Analysis
Farming Matters

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

Table 1: Revision History

Date	**Developer(s)**	**Change**
10/19/2022	Namit Chopra, Brandon Duong Andrew Balmakund, Mohammad Harun Mihail Serafimovski	Finished first version

Contents

List of Tables

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.

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:
   (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**
Database	Server can not fulfill all user requests	Can't store all user decisions, losing data necessary for the underlying research study	Too many people playing and making API requests at the same time	Ensure a queue-login system is enforced, only allow a max amount of users to play the game	IR1	H1-1
	Database can not store any more data	Can not store any user decisions, losing data necessary for the underlying research study	Storage of database is full	Admin's can download all data (user decisions) from the database and delete the data on the database afterwards, hence creating additional storage. Admins could also increase database storage capacity	IR2	H1-2
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
	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
	Account sharing	Logged decisions do not reflect 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
	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 on 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

Design Functions	Failure Modes	Effects of Failure	Causes of Failure	Recommended Action	SR	Ref
Internet Connectivity	Loses internet connection during gameplay	User loses all progress made during the current session prior to 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 a automatic save. Otherwise, the game will resume at the most recent saved progress	IR3	H3-2
General	Host computer, Web browser or the tab crashes	User loses all progress made during the current session prior to closing application	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	IR1	H4-1
	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

6 Safety and Security Requirements

The following requirements includes 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.

SR2. **The system will encrypt all user passwords with a sufficient encryption algorithm.**

6.2 Access Requirements

ACR1. **The frontend system shall allow access to any public user.**

ACR2. **The backend system shall only allow unauthenticated access to login related functionality.**

ACR3. **The backend system shall only allow access to authenticated users for all other (non-login) functionality.**

ACR4. **The backend system shall allow only to up to one user to have one user logged-in session at any point and time.**

6.3 Integrity Requirements

IR1. **The system will be able to handle all API requests in API_RESPONSE_TIME**

IR2. **The system will be able to handle all database requests in DATABASE_RESPONSE_TIME**

IR3. **The system will be able to handle unexpected loss of connection to the server**

IR4. **The user shall agree to the terms and condition before using the application**

IR5. **The system shall warn users regarding account sharing and how it will skew the data collection for research**

6.4 Privacy Requirements

PR1. **The system shall delete all user data if user decides to opt out of data collection**

PR2. **The application only requires an email provided by the user**

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	These are needed to prevent skewing of the research data obtained in the project, as discussed with the project supervisor.
	IR5	
	SR2	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.
	ACR2	
	ACR3	
	IR2	Database functionality will be expected in the final project, therefore all database requests should be handled properly
	IR4	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.
	PR1	
	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 back to the API request	0.5 seconds
DATABASE_RESPONSE_TIME	The maximum amount of time allowed for the system to respond back to the database request	0.25 seconds