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**
Date1	Name(s)	Description of changes
Date2	Name(s)	Description of changes
...	...	...

Contents

[You are free to modify this template. —SS]

1 Introduction

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.

[You can include your definition of what a hazard is here. —SS]

2 Scope and Purpose of Hazard Analysis

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)

4 Critical Assumptions

[These assumptions that are made about the software or system. You should minimize the number of assumptions that remove potential hazards. For instance, you could assume a part will never fail, but it is generally better to include this potential failure mode. —SS]

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

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.

6.2 Access Requirements

ACR1. test

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**

6.4 Privacy Requirements

PR1. test

6.5 Audit Requirements

N/A

6.6 Immunity Requirements

N/A

7 Roadmap

[Which safety requirements will be implemented as part of the capstone timeline? Which requirements will be implemented in the future? —SS]