

Table 1: Revision History

Date	Developer(s)	Change
10/05/2022	Namit Chopra, Brandon Duong Andrew Balmakund, Mohammad Harun Mihail Serafimovski	Finished First Version
01/04/2022	Namit Chopra, Brandon Duong Andrew Balmakund, Mohammad Harun Mihail Serafimovski	Second version
03/04/2022	Namit Chopra, Brandon Duong Andrew Balmakund, Mohammad Harun Mihail Serafimovski	Addressed GitHub issues

Software Requirements Specification

Farming Matters

Team #14, The Farmers

Brandon Duong

Andrew Balmakund

Mihail Serafimovski

Mohammad Harun

Namit Chopra

Contents

1	Project Drivers	1
1.1	The Purpose of the Project	1
1.2	The Stakeholders	1
1.2.1	The Client	1
1.2.2	The Players	1
1.2.3	Other Stakeholders	1
1.3	Mandated Constraints	2
1.4	Naming Conventions and Terminology	2
1.5	Relevant Facts and Assumptions about User Characteristics	3
2	Functional Requirements	3
2.1	The Scope of the Work and the Product	3
2.1.1	The Context of the Work	3
2.1.2	Work Partitioning	5
2.1.3	Individual Product Use Cases and Undesired Event Handling	5
2.2	Functional Requirements	10
2.3	Requirements That Are Likely/Unlikely to Change	12
3	Non-functional Requirements	13
3.1	Look and Feel Requirements	13
3.2	Usability and Humanity Requirements	13
3.3	Performance Requirements	13
3.4	Operational and Environmental Requirements	14
3.5	Maintainability and Support Requirements	14
3.6	Security Requirements	14
3.7	Cultural Requirements	15
3.8	Legal Requirements	15
3.9	Health and Safety Requirements	15
3.10	Requirements That Are Likely/Unlikely to Change	15
4	Traceability Matrix	16
5	Project Issues	16
5.1	Open Issues	16
5.2	Off-the-Shelf Solutions	16
5.3	New Problems	17
5.3.1	Effects on the Current Environment	17
5.3.2	Effects on the Installed Systems	17
5.3.3	Potential User Problems	17
5.3.4	Limitations in the Anticipated Implementation Environment That May Inhibit the New Product	17
5.3.5	Follow-Up Problems	17
5.4	Tasks	18

5.5	Requirements Phase-In Plan	18
5.6	Migration to the New Product	19
5.7	Risks	19
5.8	Costs	19
5.9	User Documentation and Training	20
5.10	Waiting Room	20
6	Appendix	23
6.1	Team technologies/skills - Reflection	23
6.2	Symbolic Parameters	24

List of Tables

1	Revision History	1
2	Work Partitioning Table	5
3	Likely/Unlikely to Change Table	12
4	Likely/Unlikely to Change Table	15
5	Traceability Matrix	16
6	Symbolic Parameter Table	25

List of Figures

1	Web Application Context Diagram. This diagram demonstrates the system's external boundary, that being the player interacting with the web client.	4
2	Use Case #6: Player completes a turn	9

This document describes the requirements for Farming Matters. The template for the Software Requirements Specification (SRS) is a subset of the Volere ([Robertson and Robertson, 2012](#)) template.

Some modifications made to the Volere subset template are:

- (Added) Section 3.10: Requirements that are likely/unlikely to change
- (Added) Section 4: Traceability matrix
- (Added) Section 5.5: Requirements Phase-In Plan
- (Added) Section 6.1: Team technologies/skills - Reflection
- (Removed) Section 5.11 Ideas for Solutions - Ideas for solutions are represented in the risks section, as mitigation is provided for each risk.

1 Project Drivers

1.1 The Purpose of the Project

The purpose of this project is to conduct survey research through an interactive and engaging activity. This will further help understand genuine decisions from the players to help with the research of understanding risk-making decisions.

1.2 The Stakeholders

1.2.1 The Client

The client of this project is Dr.Yiannakoulis who is the supervisor of this project. Dr.Yiannakoulis is part of the School of Earth, Environment, and Society at McMaster University.

1.2.2 The Players

Players are individuals that enjoy management and role-playing simulation games. As well as individuals willing to be a respondent to help conduct data for research.

1.2.3 Other Stakeholders

Other stakeholders would include the Ethics board. Some of the requirements regarding data collection are derived from this stakeholder. They also have to approve the final product to check if there are any ethical issues.

1.3 Mandated Constraints

- Description: The project shall have a budget of \$0.
Rationale: The developers of the project have no budget for this project. The professor might purchase game assets if he wishes to change the look of the game after the game has been created.
Fit Criterion: The developer team should use open-source and free software.
- Description: The project shall run properly on different operating systems like Linux, Windows, and Mac OS.
Rationale: The project shall be portable.
Fit Criterion: The player shall be able to run the game on different operating systems.
- Description: The client must present a consent form to participate in the study before starting the game.
Rationale: This is another requirement from both the Ethics Board and the client, and made this a point to emphasize.
- Description: The player must have the option of deleting their data.
Rationale: This is a requirement from the ethics board and players must be given this option.

1.4 Naming Conventions and Terminology

- Player: The user playing the game. The player or user is the participant and focus of the study.
- Land: An area where the player can interact with the farm. This includes planting crops, fertilizing crops, and adding buildings.
- Inventory: Where the player will be able to store items.
- Items: The player will be able to acquire these into their inventory, including seeds, crops, and fertilizer.
- Seed: Seeds are planted on the farm and evolve into crops once harvested.
- Crop: Once fully grown, seeds evolve into crops once harvested.
- Fertilizer: It is used to reduce the number of turns to harvest a crop.
- Focus groups: A set of players will be involved with discrete decision-making and another set of players will be involved with probabilistic decision-making.
- Turns: these are rounds that happen per season where a decision can be made.

- Seasons: Including Winter, Spring, Summer, and Fall. The current season changes depending on the turn number, and has an effect on which crops can be grown. Each season lasts SEASON_LENGTH turns.
- Key Questions: These are the compulsory questions that will be asked to the player. The first will be whether the player wants to pay the consultant for advice. The second question will ask the player if they want to purchase insurance for crops.

1.5 Relevant Facts and Assumptions about User Characteristics

The relevant facts and assumptions regarding player characteristics mainly pertain to a player. The player should have the physical and visual ability to operate a computer. This includes actions such as clicking a mouse and pressing buttons on the keyboard. The player should be familiar with the hardware they are using, in this case, a desktop or laptop. It would be preferable if the player had some knowledge of the basic idea of farming; however, there will be a tutorial to go over the basic rules of the game if they lack sufficient knowledge on how to play. Due to the research-oriented nature of the project and the ethical requirements, all players will be over the age of 18.

2 Functional Requirements

The following section outlines the scope of the product, use cases, and functional requirements. Functional requirements provide the core function(s) that the system should have. They were elicited and confirmed through discussions with the client. A rationale is also provided for each requirement.

2.1 The Scope of the Work and the Product

2.1.1 The Context of the Work

The Farming Matters game will create interactive and engaging management and role-playing game. The player will be provided with risk-making decisions involving the management of their farm. The game will involve in-game currency where players will have the ability to buy and sell items from an in-game shop. The system will also track and save each player's decisions which will be available for them to see at the end of each turn. The player will also have the ability to customize their farm layout and place items within their available land.

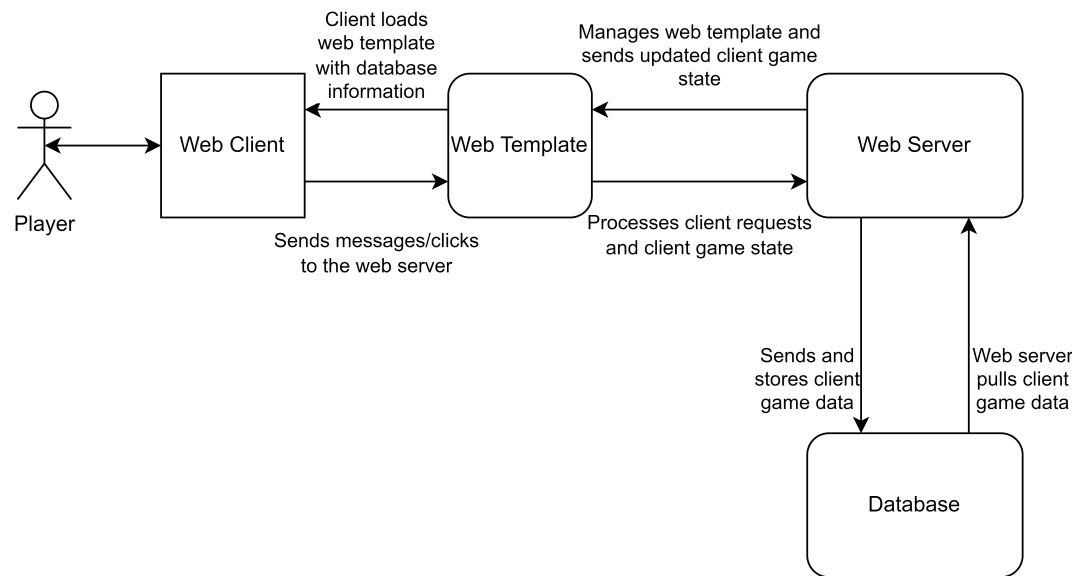


Figure 1: Web Application Context Diagram. This diagram demonstrates the system's external boundary, that being the player interacting with the web client.

2.1.2 Work Partitioning

Event	Input/Output	Summary
Player ends turn	Input: the player selects the end turn option	System responds and updates current game state
Collect player decisions	Input: all prior user decisions, player progression, and focus group Output: Group data based on focus group	System provides organized data
Player creates account	Input: Username and password Output: Associated account	The player inputs their desired username and password and the system creates an account and stores the credentials
Player requests to delete their data	Input: Username and password Output: Associated account and all its data is deleted	The player inputs their account's username and password and the system deletes the associated account and all its corresponding data

Table 2: Work Partitioning Table

2.1.3 Individual Product Use Cases and Undesired Event Handling

Use case #1: Player creates account

Primary Actor: Player

Supporting Actors: None

Precondition: The player has completed the consent form

Trigger: The player is directed to the create account page

Main Success Scenario

1. Player is on the home page and selects to create account
2. Player is on the create account page
3. Player provides the required information
4. System verifies all required information has been provided
5. Player completes human verification task
6. System validates human verification
7. System securely registers user information
8. System assigns a player to receive either probabilistic or deterministic information

9. Player is redirected back to the Home page

Secondary Scenarios:

1. Player fails to provide the required information: one or more fields have not been provided by the player. The player should be aware of which field is missing/required
2. Player fails to provide a valid email address. The player should be aware of the proper email formatting required to create an account
3. Player fails to meet the requirements of a password they provided. The player should be aware of the necessary details to create a valid password
4. Player fails to provide a matching password when asked to reenter the password they provided. The player should be aware if the initial password provided does not match the password in the password confirmation field
5. Player fails human verification task: the player has failed a human verification test which either means they entered in the wrong details or an automated script attack for account creation occurred

Success Postcondition: The player has successfully created an account and account information is stored and secured in a database.

Use case #2: Player signs into account

Primary Actor: Player

Supporting Actors: None

Precondition: The player has successfully created an account

Trigger: The player arrives at the login page

Main Success Scenario

1. Player is on the home page and selects to login
2. Player is on the login page
3. Player provides the required information
4. System verifies all required information has been provided
5. Player is redirected to the Home page as a logged-in player

Secondary Scenarios:

1. Login unsuccessful: Provided login details do not match any login details in the database. The player either does not have an account created or they have entered incorrect login details. In either case, the player should be aware if either issue is present.

Success Postcondition: The player has successfully logged into their created account and their game state is loaded from the database.

Use case #3: Player logs out of their account

Primary Actor: Player

Supporting Actors: None

Precondition: The player has successfully created an account and is currently logged in

Trigger: The player has selected an option to logout

Main Success Scenario

1. Player selects logout option
2. Player is redirected to the Home page

Secondary Scenarios:

1. Player is inactive: After INACTIVE_TIME period, the system shall log out the player automatically.

Success Postcondition: The player has successfully logged out of the system and all logged data has been stored successfully.

Use case #4: Player retrieves forgotten password

Primary Actor: Player

Supporting Actors: None

Precondition: The player has successfully created an account

Trigger: The player has selected an option to forget password

Main Success Scenario

1. Player accesses login page and selection forget password option
2. Player enters required information to gain back access to account
3. System sends the player follow-up details to confirm the legitimacy of their account
4. Player is able to provide a new password
5. System shall update the password of the current player in the database
6. Player is redirected to the Home page as a logged-in player

Secondary Scenarios:

1. System is unable to verify account details: The player may have entered the incorrect details needed to obtain back access to the account. The player should be aware if the details do not exist in the system or if the details do not match with the current records the system has

Success Postcondition: The player has successfully updated their account details and is logged in.

Use case #5: Player opts out to delete their data

Primary Actor: Player

Supporting Actors: None

Precondition: The player has successfully created an account and is currently signed in

Trigger: The player has selected an option to delete their data

Main Success Scenario

1. Player accesses selection to delete data
2. Player confirms the action of deleting their data
3. System logs the player out
4. System deletes all data belonging to the player
5. Player is redirected to the Home page

Secondary Scenarios:

1. Player decides not to delete their data upon confirmation screen: when presented with the option to confirm once again with deleting their data. If they decide not to and would like to cancel this option, the system will do nothing and simply redirect them back to the menu they accessed the delete their data

Success Postcondition: The player has successfully opted out to delete their data and is no longer stored in the database.

Use case# 6: Player completes a turn

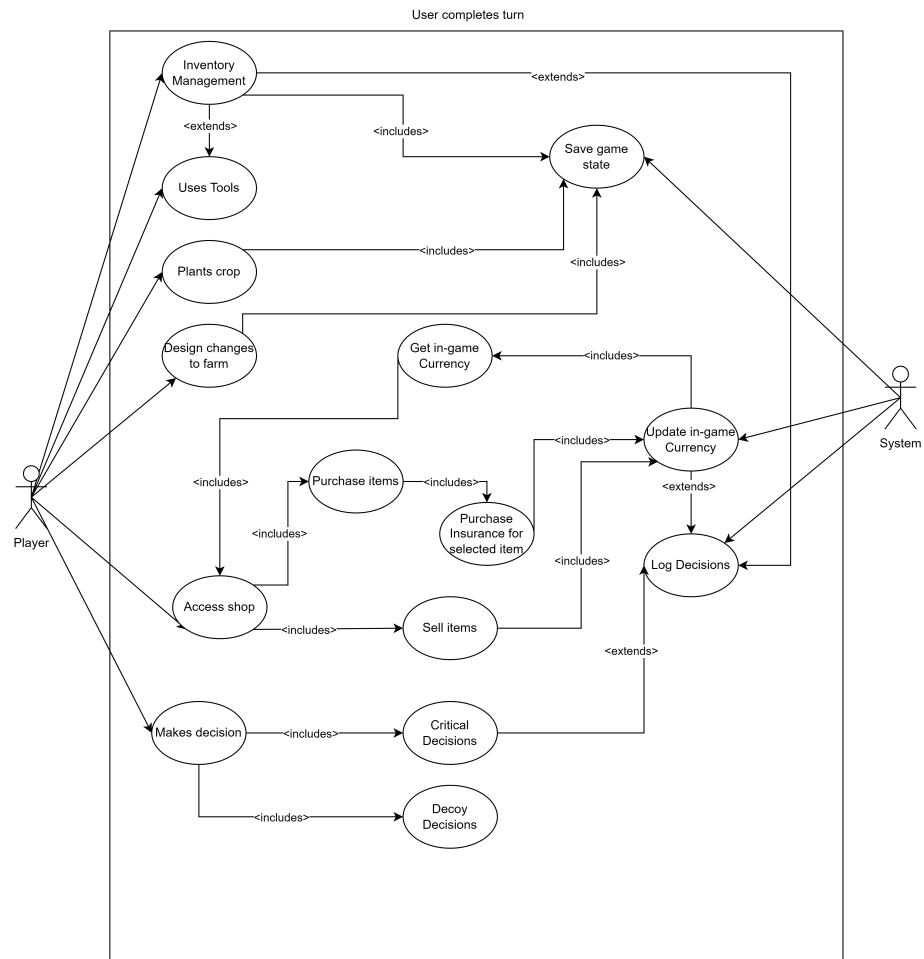


Figure 2: Use Case #6: Player completes a turn

2.2 Functional Requirements

- FR1. The system must allow players to create an account.
Rationale: The purpose of the game is to track player decisions and be able to see which participant did what decision. For this to happen, players must be associated with an account.
- FR2. The system must allow the player to reset their password.
Rationale: It is possible a player misplaces their password and loses all their account progress. This effectively removes the participant from the study and so this must be rectified.
- FR3. The system must allow players to accumulate in-game currency.
Rationale: In-game currency acts as a performance metric, and also helps engage the player in planning for their future decisions. Optimizing what one should spend their in-game currency on is both parts of the study and the engagement of the game itself.
- FR4. The system must allow the player to store their items in an inventory.
Rationale: The player must be able to view the items they currently own.
- FR5. The system must verify new players as humans.
Rationale: As anyone can make an account, the system is susceptible to related attacks. The system verifying a player as a human before creating their account may slow down these attempts.
- FR6. The system must allow players to purchase items from a shop.
Rationale: Helps engage the player with the feeling of being rewarded and a decision-making process of what to plan for the future.
- FR7. The system must allow players to place in-season seeds on their land.
Rationale: This will give the player more variability and control in terms of how they want to build their farm. It also restricts players from planting seeds in unsuitable seasons due to weather conditions.
- FR8. The system must allow players to grow seeds on owned land.
Rationale: The player will only be able to sell crops so the seeds must be fully grown and harvested before selling.
- FR9. The system must allow players to harvest the crops. Seeds will evolve into crops after the specified number of turns for each seed.
Rationale: This simulates the real-life growth process in farming. It also helps the players differentiate the different items that can be sold vs planted.
- FR10. The system must allow players to sell owned crops at a fluctuating price. The fluctuating prices for each crop will be generated using a normal distribution with each crop having a different base price (mean) and

fluctuation factor (standard deviation).

Rationale: Helps to somewhat simulate a real market where crop prices fluctuate. This was also a requirement that was emphasized by the client.

- FR11. The system must allow players to use fertilizer on planted crops.

Rationale: Helps the player to grow crops faster to anticipate any events that may destroy their crops.

- FR12. The system must allow players to buy unowned land.

Rationale: Players may want to increase their chances of crop yield and may want to plant a variety of them to further increase their total balance

- FR13. The system must prompt players for consulting advice every CONSULTING.INTERVAL turns.

Rationale: This decision is one of the two key questions the research is looking at. Whether people are willing to pay for information, and for what information (i.e deterministic or probabilistic).

- FR14. The system must allow players to buy insurance on purchased seeds.

Rationale: This decision is the second of the two key questions the research is looking at. Whether people are willing to pay for insurance, and in what circumstance (i.e from the start, or only after they've been affected by an insurable event).

- FR15. Harvested crops in the inventory can expire after a certain number of turns, given by the crop type. Once a crop expires, it is removed from the inventory without any compensation.

Rationale: This prevents players from holding onto the crop indefinitely until the highest price, which would take away from the strategizing to maximize the profit.

- FR16. The system must be able to log player decisions.

Rationale: This was a crucial requirement that was provided by Dr.Yiannakoulis who will be able to view these logs to analyze the data.

- FR17. The system must be able to save the player's game state.

Rationale: This will give players the option to continue where they left off if they want to continue playing at another time.

- FR18. The system must have a defined area of land for players to manage the farm.

Rationale: This will provide an environment in which players can visualize and interact with their farm.

- FR19. The system must be able to assign a player to a specific focus group.

Rationale: This is essential for the research study to demonstrate whether or not people prefer deterministic or probabilistic information.

- FR20. The system must include random events to occur EVENT_OCCURRENCE.
Rationale: Helps engage the player by needing them to plan for the possible unknown.
- FR21. The system must have the current season change over time.
Rationale: Helps engage the player as they must plan for what is possible during the different seasons in terms of what to plant, and what risks there are within each season.
- FR22. The system must assign users to one of two information types: probabilistic or deterministic date. All information provided will be given by the assigned method.
Rationale: The client would like to research whether the players prefer one type of method over the other.

2.3 Requirements That Are Likely/Unlikely to Change

Likely	Unlikely
FR10	FR1, FR2, FR3, FR4, FR5, FR6, FR7, FR8, FR9, FR11, FR12, FR13, FR14, FR15, FR16, FR17, FR18, FR19, FR20, FR21, FR22

Table 3: Likely/Unlikely to Change Table

- FR1: This is unlikely to change as accounts are needed to associate a specific player to their game state
- FR2: This is unlikely to change as accounts will need a way to be recovered should they happen to lose access to it
- FR3, FR4, FR6, FR7, FR8, FR9, FR11, FR12, FR18, FR20, FR21: These are unlikely to change as these are the basic core game mechanics that define the desired game loop
- FR5: This is unlikely to change as the system must ensure humans are playing the game for accurate data collection.
- FR10: This is likely to change as the stakeholder may want to implement a different pricing method for selling items
- FR13, FR14, FR15, FR16, FR19, FR22: These are unlikely to change as they implement the essentials behind the research study aspect
- FR17: This is unlikely to players must be able to play the game over multiple sessions

3 Non-functional Requirements

The following sections outline the non-functional requirements of the system. Non-Functional requirements are requirements set out to help outline how a system should be. These requirements consist of Look and Feel, Usability and Humanity, Performance, Operational and Environmental, Maintainability and Support, Security, Cultural, and Health and Safety.

3.1 Look and Feel Requirements

- LF1. The menu shall be minimalistic.
Fit Criteria: The menu should only require the necessary elements and not overwhelm the player.
- LF2. The menu and game interface shall follow a consistent theme.
Fit Criteria: All visual assets should include the same colors and style across all aspects of the game.
- LF3. The system must have engaging audio.
Fit Criteria: Survey a group of individuals and 90% of them should be satisfied with the audio assets.
- LF4. The system must have engaging graphics.
Fit Criteria: Survey a group of individuals and 90% of them should be satisfied with the graphical assets.
- LF5. The system must be responsive on monitors at least MIN_MONITOR.
Fit Criteria: All aspects of the system should be functioning accordingly and be responding at the resolution set to MIN_MONITOR.
- LF6. The system must have engaging gameplay.
Fit Criteria: Average number of turns played is MIN_TURNS.

3.2 Usability and Humanity Requirements

- UH1. The system must be easy to learn for people aged AGE_GROUP.
Fit Criteria: Survey a group of individuals and 95% of them should easily understand the in-game logic, understand the information provided to them and be able to interact with all the menus.

3.3 Performance Requirements

- PR1. The website must load within LOAD_TIME seconds.
Fit Criteria: The measured time from the player sending an initial request to access the website and the website loading is less than or equal to LOAD_TIME.

- PR2. The website must respond to any player interaction within RESPONSE_TIME seconds.
Fit Criteria: The measured time from the player interacting with the website and the website providing the player with some response is less than or equal to RESPONSE_TIME.
- PR3. The website must be available for UPTIME_THRESHOLD.
Fit Criteria: The measured time the website should be available to be running at least by.

3.4 Operational and Environmental Requirements

- OE1. The system must be functional on modern browsers.
Fit Criteria: The player shall be able to run the game on different modern browsers such as Google Chrome, Firefox, and Microsoft Edge.
- OE2. The system must be functional on any version of the browser it's used on released within the past SUPPORTED_VERSIONS.
Fit Criteria: The player shall be able to run the game on any supported version of their browser of choice (so long as their browser is supported).

3.5 Maintainability and Support Requirements

- MR1. The system must allow assets to be replaced.
Fit Criteria: The development team and client shall be able to change any visual or audio assets with ease and not disrupt any part of the system.

3.6 Security Requirements

- SR1. The system must verify that an account creation request do not come from an automated script.
Fit Criteria: There should be measures in place to prevent automated attacks.
- SR2. The logged and account data can only be accessed by the client and the development team.
Fit Criteria: Database permissions are granted to only approved personnel.
- SR3. Sensitive data will be stored using trusted third-party software.
Fit Criteria: It is beneficial to use a trusted third-party software that specializes in data protection and security. They will use up-to-date technology and algorithms to ensure the safety of the sensitive data that needs to be stored.

3.7 Cultural Requirements

N.A.

3.8 Legal Requirements

LR1. The system must not keep player data if a player retracts consent and wants to delete their data.

Fit Criteria: Upon request to delete data, there should be no instance of the player's data in the database. This includes any backed-up data that is kept on file of the player.

LR2. The system does not commit copyright infringement.

Fit Criteria: The system must only use assets that have an open license.

3.9 Health and Safety Requirements

N.A.

3.10 Requirements That Are Likely/Unlikely to Change

Likely	Unlikely
PR3	LF1, LF2, LF3, LF4, LF5, LF6, UH1, PR1, PR2, OE1, OE2, MR1, SR1, SR2, SR3, LR1, LR2

Table 4: Likely/Unlikely to Change Table

- LF1, LF2, LF3, LF4, LF6: These are unlikely to change as these are needed to fully immerse the player within the game and not think about the underlying research (to ensure genuine results)
- LF5, UH1, OE1, OE2: This is unlikely to change as these tackle the problem of having the application as accessible as possible
- PR1, PR2: These are unlikely to change to ensure user engagement is not sacrificed due to poor system response times
- PR3: This is likely to change as the uptime is dependent on how long the client chooses to conduct the study
- MR1: This is unlikely to change as it allows Dr.Yiannakoulis to tweak and further improve upon the finished product.
- LR1: This is unlikely to change as this is required for the ethics board to approve the finished product.

4 Traceability Matrix

The traceability matrix below shows the relationships between functional requirements and non-functional requirements/risks. Any cells with a 'Y' trace to each other.

	LF1	LF2	LF3	LF4	LF5	LF6	UH1	PR1	PR2	PR3	OE1	OE2	MR1	SR1	SR2	SR3	LR1	LR2	R1	R2	R3
FR1	Y	Y		Y	Y		Y	Y	Y	Y	Y	Y		Y	Y	Y	Y			Y	
FR2	Y	Y		Y	Y		Y	Y	Y	Y	Y	Y		Y	Y	Y	Y			Y	
FR3	Y	Y		Y		Y	Y	Y	Y	Y	Y	Y			Y		Y				Y
FR4	Y	Y		Y		Y	Y	Y	Y	Y	Y	Y	Y				Y		Y		Y
FR5											Y	Y		Y	Y	Y					
FR6	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y		Y	Y			Y
FR7	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y		Y	Y	Y		Y
FR8	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y		Y	Y	Y		Y
FR9	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y		Y	Y	Y		Y
FR10	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y		Y	Y			Y
FR11	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y		Y	Y	Y		Y
FR12	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y		Y	Y			Y
FR13	Y	Y		Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y		Y				Y
FR14	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y		Y	Y			Y
FR15			Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y		Y		Y		Y
FR16								Y	Y	Y					Y		Y			Y	
FR17								Y	Y	Y					Y		Y		Y	Y	
FR18	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y		Y	Y	Y		Y
FR19								Y	Y	Y					Y		Y				
FR20	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y			Y	Y		Y
FR21	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y			Y	Y		Y
FR22								Y	Y	Y					Y		Y				

Table 5: Traceability Matrix

5 Project Issues

The following section outlines the all relevant issues related to the project. The following category of issues considered are Open Issues, Off-the-Shelf Solutions, New Problems, Tasks, Requirements Phase-In Plan, Migration to the New Product, Risks, User Documentation, and Training and Waiting Room.

5.1 Open Issues

N/A

5.2 Off-the-Shelf Solutions

Although there are games that are themed around farming, they do not provide all the aspects required for this project. Games like FarmVille and Stardew Valley provide an entertaining and engaging way to manage a farm. However, they do not provide information about player decisions nor are they targeted toward research.

There are also lab experiments and surveys used for collecting data for the specified research area. The results from these methods often do not accurate information about how humans make decisions in real life. The subjects tend to complete these methods for the incentive which results in an inaccurate collection of data.

There aren't any off-the-shelf solutions available that address all aspects of this problem. The game must be engaging and collect information about risk vs reward decisions in managing a farm. The information collected must also match the format which can be used for research. The combination of information collection, an engaging game, and the visibility of logged information is unavailable off-the-shelf.

5.3 New Problems

5.3.1 Effects on the Current Environment

The game will be graphics intensive which may contribute to a lag on certain devices. This will be exacerbated if the machine is running numerous other applications. The main game decisions also need to be logged into a database which may contribute to a lag as well.

5.3.2 Effects on the Installed Systems

N/A

5.3.3 Potential User Problems

The learning curve play is an important factor that determines how many players play the game. As the game should depict real-life decisions needed to manage a farm, the game may be complex and contain many different aspects. Failure to provide an informative guide may result in a lack of understanding of how to play the game. Furthermore, it may also result in many players quitting before they have made any progress in the game.

5.3.4 Limitations in the Anticipated Implementation Environment That May Inhibit the New Product

A browser and an internet connection are required to play the game. It is not uncommon for one to lose an internet connection while using the browser. Players may experience a loss of connection as they are playing the game which will limit the product. The player may be able to interact with the interface of the game; however, none of the decisions will be logged nor will the game state be saved.

5.3.5 Follow-Up Problems

There will be a risk if the libraries get updated during the development. This may result in in-game features being unable to run or load.

The research may need to be updated that better targets the information being searched for in the game. The game may not be providing the information that is useful to the research. The game may need to be re-designed to better obtain the information needed for the research area.

5.4 Tasks

The following is the link to our project schedule and all task deadlines: [Gantt chart](#).

5.5 Requirements Phase-In Plan

Requirements	Priority	Deadline
FR1	High	Dec. 1, 2022
FR2	Low	Jan. 31, 2023
FR3	High	Dec. 1, 2022
FR4	High	Jan. 9, 2023
FR5	Low	Jan. 9, 2023
FR6	High	Dec. 31, 2022
FR7	High	Nov. 14, 2022
FR8	High	Jan. 9, 2023
FR9	High	Jan, 31, 2023
FR10	High	Jan. 9, 2023
FR11	Medium	Nov. 14, 2022
FR12	Medium	Nov. 14, 2022
FR13	High	Nov. 14, 2022
FR14	High	Dec. 31, 2022
FR15	High	Jan. 9, 2023
FR16	High	Nov. 14, 2022
FR17	Medium	Jan. 9, 2023
FR18	High	Jan. 9, 2023
FR19	High	Dec. 31, 2022
FR20	Medium	Nov. 14, 2022
FR21	Medium	Jan. 9, 2023
FR22	High	Dec. 31, 2022

We specified the following requirements: FR1,FR3,FR4, FR6, FR7, FR8, FR9,FR18 above that are considered to be a High priority because these requirements are essential to creating a minimum viable product. The remaing requirements are also High priority, FR10,FR13, FR14, FR15, FR16, FR19, FR22, because these are core elements of the research study.

The requirements that are Medium priority are related to the game mechanics and will allow a more enriching experience for the player.

The requirements that are Low priority are related to the additional add-ons to the game that could further enhance the player experience and make

the game more interesting

5.6 Migration to the New Product

N/A

5.7 Risks

- R1. May experience lag when rendering 3D-models

Description: When rendering 3D models for the crops there may be some lag if there are too many models and if it is run on a low-spec machine. As well as any other 3D models use to render the visual environment of the game.

Mitigation: The assets will be preloaded upon logging in to avoid fetching and loading the 3D models on different re-renders. The player can also enable settings on their browser to enable hardware acceleration to make gameplay more smooth and reduce lag.

- R2. Ethics board does not approve the project

Description: In order for the game to be used as a research tool, it must be approved by a regulatory ethics board. There is an extremely minor risk that the ethics board could not approve the project after the game is made.

Mitigation: Our supervisor is very knowledgeable about the ethics board requirements, so we worked closely with him to capture requirements that would ensure approval. [FR15](#) is the requirement that could help to mitigate this risk along with the constraint that allows players to delete their data.

- R3. Game is not immersive

Description: The entire purpose of the game is to gather research data in a way where the player does not feel as if they are in a research environment. To do this, we need to ensure that the player has fun while playing and feels immersed in the game. There is a risk they may not, which would defeat the purpose of the game.

Mitigation: We are mitigating this risk by adding many gameplay mechanics and features that will ensure the player feels immersed. We are additionally looking at existing popular farming games to see what makes people feel immersed, and incorporating that into our design.

5.8 Costs

There are two costs for this project. The first is the graphical assets used in the game. We will have to pay an artist to create digital assets in the form of images, icons, etc. for us to use in the game. Based on discussions with the supervisor, the graphics for the game will be 2d. Taking this into account, we researched the costs of 2d digital art and there are several different price models. If we want custom assets, we will either purchase a set of assets, or we will pay

an artist hourly to create them. Overall, graphical assets will cost no more than \$300 for a polished game.

With this being said, we have decided to run the project with zero budget initially, and for that reason, we will use all free graphical assets. If the supervisor decides to pay for assets in the future, the system shall be designed in a way where the assets will be easily replaceable (as captured in [MR1](#)).

The second primary cost will be the operating cost of the servers needed to run the backend code. Our supervisor (Dr. Yiannakoulis) said that he would handle the hosting of the code on his end, so we should consider that an external cost (ie. it's not added to the project budget). Regardless, since this project does not need to be scaled to high levels of traffic, the operating costs should be low (less than \$50/year)

All other functionality can be achieved at no cost, by using open-source solutions. Therefore, the total cost to the group will be \$0.

5.9 User Documentation and Training

Players will be presented with the following documentation and training throughout the game:

- **Menu With Gameplay Help:** The player will be able to access a menu that contains instructions on the overall gameplay, including the goals of the game, the actions that can be done in a turn, how to make money, etc.
- **Menu with controls:** The player will be able to access a menu that contains instructions on the controls of the game.
- **Tooltips:** Throughout the player's gameplay, any components which do not have a clear purpose will have tooltips when the player hovers over them.
- **Ethics Documentation:** The player will be given documentation about the ethics of the research and their consent, as required by the ethics board.

5.10 Waiting Room

This section serves as a place to keep track of potential future requirements or functionalities that we may not have time to implement.

- **Asynchronous multiplayer:** This will be an exciting feature as players will be able to compete with other players in terms of how efficient they were with their turns, and what decisions the other players made. Multiplayer gameplay is usually a feature found in most games today; however,

it will be asynchronous so players don't have to wait on others before they can perform actions.

- **Mini-games:** This will be a fun in-game feature that aims to keep players enticed. Mini-games could be played for various reasons. This could be an enjoyable option to earn money for buying items required for their farm. It is also an additional feature to keep the players engaged and motivated to continue playing the game. Combined with asynchronous multiplayer, leaderboards could also be introduced to track minigame performance.
- **Visualizing Data:** The data logged and sent to databases could be graphed and visualized in many other ways. This would also be useful for analyzing the results of the experiment.

References

James Robertson and Suzanne Robertson. *Volere Requirements Specification Template*. Atlantic Systems Guild Limited, 16 edition, 2012.

6 Appendix

The following section outlines miscellaneous information that may be required to understand the material discussed throughout this document. Such information includes Team technologies/skills and Symbolic Parameters.

6.1 Team technologies/skills - Reflection

1. **ReactJS:** To master this technology, the following members will go through an in-depth tutorial on ReactJS [Tutorial: Intro to React](#) and complete a basic project that can be found on YouTube (i.e Create a basic sign-up/login-page) to exercise different components taught in the ReactJS tutorial. This will help in both the development of this project and future projects as ReactJS is one of the most popular frameworks used in web development. ReactJS is also very similar to other popular web development frameworks such as VueJS, so learning ReactJS will help with learning similar frameworks in the future.
 - Students: Mihail, Brandon, Namit, Andrew, Mohammad
2. **Team Management:** To master this skill, the following members will go through an online LinkedIn course about team management and effective team leadership [Leadership and Management Online Training Courses](#). Another approach is to take on a leadership role when completing the mini projects above that will be done in small groups of 3 or 4 for each group. This skill was chosen because team management is a large factor in a project's success that is often overlooked and gaining experience now will help for any future projects.
 - Students: Mihail, Namit, Mohammad
3. **Web Development:** To master this technology, the following member will go through a series of basic web development fundamentals and concepts such as [HTML Tutorial](#), [CSS Tutorial](#) and [JavaScript Tutorial](#) provided by W3Schools. As well, a mini project suggested above will be used to demonstrate the understanding and skills learned through the tutorials. This skill was chosen because this member had very little experience with this tech stack and learning this could be helpful for further projects and work.
 - Student: Mohammad
4. **ExpressJS:** To master this technology, the following member will go through an in-depth tutorial and documentation [Express - Node.js Web Application Framework](#). As well, a mini project suggested above will be used to demonstrate the understanding and skills learned through the in-depth tutorial and documentation. This technology was chosen because the following members had a small experience with Express.js and would

like to further get a better understanding of middleware in web development, and routing, and ultimately better at server-side development. This will prove useful in the future as most web development projects require knowledge and the development of a backend. Express.js is similar to many popular backend frameworks, therefore if a future project's backend uses some other framework similar to Express.js, the student can still leverage their previous experience with Express.js.

- Student: Andrew

5. **SQL Database:** To master this technology, the following members will go through an in-depth tutorial for [SQL](#). They will also get hands-on experience by working on a mini-project that will leverage SQL to demonstrate the understanding and skills learned through the tutorial. If needed, other team members with prior SQL experience can assist. This technology was chosen as it is relevant to the industry and will be used in the project. A database is another large component of many web development endeavors, and so exercising this skill can help design effective schema for future projects.

- Students: Namit, Brandon

6.2 Symbolic Parameters

The definition of the requirements will likely call for `SYMBOLIC_CONSTANTS`. Their values are defined in this section for easy maintenance.

Table 6: **Symbolic Parameter Table**

Symbolic Parameter	Description	Value
AGE_GROUP	The age of the players playing the game	18 and above
MIN_MONITOR	The minimum supported monitor resolution for responsiveness	1280 by 720
INACTIVE_TIME	The time used to establish an inactive player. No data has been logged or any control inputs (mouse or keyboard) by the player	15 minutes
LOAD_TIME	The maximum time allowed for the application to successfully load	5 seconds
RESPONSE_TIME	The maximum time allowed for the application to respond to player input	5 seconds
CONSULTING_INTERVAL	The number of turns between each consultant visit	3 turns
EVENT_OCCURRENCE	The number of times an event will occur throughout a season	2/season
SEASON_LENGTH	The number of turns per season	3 turns
SUPPORTED_VERSIONS	The oldest supported version of browsers	1 year
MIN_TURNS	The minimum amount of turns played needed for a study participant to be a significant data point	12 turns
UPTIME_THRESHOLD	The minimum amount of time the website should be up and running	4 weeks