

# Software Requirements Specification for Software Eng: Mac AR

Team #11, Mac-AR  
Student 1 Matthew Collard  
Student 2 Sam Gorman  
Student 3 Ethan Kannampuzha  
Student 4 Kieran Gara

April 4, 2024

# Contents

<b>1</b>	<b>Project Drivers</b>	<b>1</b>
1.1	The Purpose of the Project . . . . .	1
1.2	The Stakeholders . . . . .	1
1.2.1	The Client . . . . .	1
1.2.2	Hands-on Users of the Application . . . . .	2
1.3	Mandated Constraints . . . . .	2
1.3.1	Solution Constraints . . . . .	2
1.3.2	Implementation Environment of the Current System . . . . .	2
1.3.3	Partner or Collaborative Applications . . . . .	2
1.3.4	Off-the-Shelf Software . . . . .	3
1.3.5	Anticipated Workplace Environment . . . . .	3
1.3.6	Schedule Constraints . . . . .	3
1.3.7	Budget Constraints . . . . .	3
1.3.8	Enterprise Constraints . . . . .	3
1.4	Naming Conventions and Terminology . . . . .	4
1.5	Relevant Facts and Assumptions . . . . .	4
1.5.1	User Assumptions . . . . .	5
1.5.2	Device Assumptions . . . . .	5
<b>2</b>	<b>Functional Requirements</b>	<b>5</b>
2.1	The Scope of the Work and the Product . . . . .	5
2.1.1	Context of the Work . . . . .	5
2.1.2	Individual Product Use Cases . . . . .	7
2.2	Functional Requirements . . . . .	9
2.2.1	Create Game Room . . . . .	9
2.2.2	Join Game Room . . . . .	10
2.2.3	Edit Room Settings . . . . .	10
2.2.4	Exit Room . . . . .	11
2.2.5	Start Game . . . . .	11
2.2.6	Puzzle Interaction . . . . .	12
2.2.7	Hint Request . . . . .	12
2.2.8	Skip Puzzle . . . . .	12
2.2.9	Send Text Message . . . . .	13
2.2.10	Read Message . . . . .	13
2.2.11	Voice Communication . . . . .	13
2.2.12	Simon Says Puzzle . . . . .	14
2.2.13	Maze Puzzle . . . . .	14
2.2.14	Isometric Puzzle . . . . .	15
2.2.15	Combination Puzzle . . . . .	15
2.2.16	Wires Puzzle . . . . .	16
2.3	Formal Specification . . . . .	17

2.4	Functional Requirements That Are Likely/Unlikely to Change . . . . .	18
<b>3</b>	<b>Non-functional Requirements</b>	<b>19</b>
3.1	Look and Feel Requirements . . . . .	19
3.2	Usability and Humanity Requirements . . . . .	19
3.3	Performance Requirements . . . . .	20
3.4	Operational and Environmental Requirements . . . . .	20
3.5	Maintainability and Support Requirements . . . . .	20
3.6	Security Requirements . . . . .	21
3.7	Cultural Requirements . . . . .	21
3.8	Legal Requirements . . . . .	21
3.9	Health and Safety Requirements . . . . .	21
3.10	Non-Functional Requirements That Are Likely/Unlikely to Change . . . . .	21
<b>4</b>	<b>Traceability Matrix</b>	<b>23</b>
<b>5</b>	<b>Project Issues</b>	<b>24</b>
5.1	Open Issues . . . . .	25
5.2	Off-the-Shelf Solutions . . . . .	25
5.2.1	Ready-Made Products . . . . .	25
5.2.2	Reusable Components . . . . .	25
5.2.3	Products That Can Be Copied . . . . .	25
5.3	New Problems . . . . .	26
5.3.1	Effects on the Current Environment . . . . .	26
5.3.2	Effects on the Installed Systems . . . . .	26
5.3.3	Potential User Problems . . . . .	26
5.3.4	Limitations in the Anticipated Implementation Environment That May Inhibit the New Product . . . . .	26
5.3.5	Follow-Up Problems . . . . .	26
5.4	Tasks . . . . .	26
5.4.1	Project Planning . . . . .	27
5.4.2	Planning of the Development Phases . . . . .	27
5.5	Migration to the New Product . . . . .	27
5.5.1	Requirements for Migration to the New Product . . . . .	28
5.5.2	Data That Has to Be Modified or Translated for the New System . . . . .	28
5.6	Risks . . . . .	28
5.7	Costs . . . . .	28
5.8	User Documentation and Training . . . . .	28
5.8.1	User Documentation Requirements . . . . .	28
5.8.2	Training Requirements . . . . .	28
5.9	Waiting Room . . . . .	29
5.10	Ideas for Solutions . . . . .	29

<b>6</b>	<b>Appendix</b>	<b>29</b>
6.1	Reflection . . . . .	29
6.2	Symbolic Parameters . . . . .	30

Table 1: Revision History

Date	Developer(s)	Change
September 26, 2023	Matthew and Sam	Initial changes to the project
September 26, 2023	Matthew	Initial NFR changes
September 28, 2023	All	General format editing and addition of the Use Case Diagram
September 29, 2023	Matthew	Addition of Use cases 1-5 and related functional requirements
September 29, 2023	Kieran	Addition of Use cases 5-10 and related functional requirements
September 29, 2023	Ethan	Added Read-Made Products section to Project Issues
September 30, 2023	Ethan	Completed rest of sections in Project Issues
October 1, 2023	Sam	Filled out sections 1, 6
October 6, 2023	All	Group worked on traceability matrix and finishing rest of SRS
October 27, 2023	Ethan	Updated SRS based on feedback from Team 2 and Team 18
November 3, 2023	Ethan	Updated SRS to include Hazard Analysis requirements
January 4, 2024	Ethan	Updated SRS to include requirement for notifying user to update to latest version of application
March 1, 2024	Ethan	Updated SRS to remove Save/Load/Delete Requirements
March 6, 2024	Sam	Updated PI7 to be more explicit about progress bar
April 3, 2024	Kieran	Revision 1 updates complete

This document describes the requirements for Mac-AR. The template for the Software Requirements Specification (SRS) is a subset of the Volere template ([Robertson and Robertson, 2012](#)) with the following modifications:

- (Added) Section 2.3: Formal Specification
- (Added) Section 2.4: Functional Requirements That Are Likely/Unlikely to Change
- (Added) Section 3.9: Health and Safety Requirements
- (Added) Section 3.10: Non-Functional Requirements That Are Likely/Unlikely to Change
- (Added) Section 4: Traceability matrix
- (Added) Section 5.5: Cutover section replaced with “Migration to the New Product”
- (Added) Section 6: Appendix
- (Merged) Section 1.2: “Stakeholders” and “Users of the Product” sections merged under “Stakeholders” section
- (Modified) Modified overall section nesting and numbering for more logical grouping

# 1 Project Drivers

This section will give a general overview of the motivation behind creating Mac-AR, along with what constraints it needs to be created within.

## 1.1 The Purpose of the Project

In the modern world, it has become increasingly harder to meet new people and form connections with others. This has become especially true in recent years, with the pandemic isolating people even further. Mac-AR aims to solve this by providing a fun, interactive means to cooperate with others and form new connections. As a team, users will be able to work alongside people either co-located or remotely at places they prefer, to solve interesting puzzles and bond with each other.

## 1.2 The Stakeholders

The following groups and individuals are the reason for the existence of Mac-AR.

### 1.2.1 The Client

Dr Irene Yuan is the sole client for the project, requesting that the tool be made. She will act as the point of contact during development and ensure that the final result is in line with what was envisioned.

### 1.2.2 Hands-on Users of the Application

The users are the ones who will be using the tool and building connections with others. The main user base is intended to be young adults, roughly ages 18-30, who enjoy games and are moderately comfortable using technology.

## 1.3 Mandated Constraints

The project will be developed and released within the following constraints.

### 1.3.1 Solution Constraints

*Description:* The product shall be built as a mobile application.

*Rationale:* The supervisor wants the application to be a mobile application, as to ensure the largest freedom to the users for where and how the product will be used.

*Fit Criterion:* The product shall be written using Unity engine and be accessible on both Android and iOS application stores.

*Description:* The product shall be able to function on a variety of mobile devices, using both iOS and Android.

*Rationale:* To provide service to the widest array of users, the most popular phone brands, and by extension their operating systems, need to be supported.

*Fit Criterion:* The product shall be tested to function on popular mobile devices using iOS and Android, specifically

- Samsung S20 FE 5G
- iPhone SE
- Motorola G7 Power
- Google Pixel 7
- iPhone Special Edition 2020

### 1.3.2 Implementation Environment of the Current System

As outlined in section [1.3.1](#), the tool will need to be developed for execution on modern mobile devices. There are currently no constraints on the server side of the project, although that may need to be revised as the tool grows in scope.

### 1.3.3 Partner or Collaborative Applications

In the current design of the product, there are no partner or collaborative applications that will work along with the product. Therefore, there are no partner or collaborative constraints.

### 1.3.4 Off-the-Shelf Software

The following off-the-shelf software will be utilized:

- Unity, along with its available libraries for AR app development and communication
- Visual Studio, for development

### 1.3.5 Anticipated Workplace Environment

The anticipated workplace environment will be intentionally broad. The product can be used from anywhere the user has access to a mobile device and internet to run the application, with no limits on being indoors.

### 1.3.6 Schedule Constraints

The general document and development milestone constraints are as follows:

Table 2: Schedule Constraints

Date	Deliverable
Sept 25, 2023	Problem Statement
Sept 25, 2023	Development Plan
Oct 5, 2023	Requirements Document
Oct 20, 2023	Hazard Analysis
Nov 3, 2023	Verification and Validation Plan
Nov 13-24, 2023	Proof of Concept Demo
Jan 17, 2024	Design Document
Feb 5-16, 2024	Revision 0 Demo
Mar 6, 2024	Verification and Validation Report
Mar 18-29, 2024	Final Demo (Rev 1)
Apr 4, 2024	Final Documentation

As a general constraint applied on top of all of these, all work should be done a few days in advance to allow for time to review.

### 1.3.7 Budget Constraints

The project has no monetary budget. If there are any necessary purchases for development, the cost shall be paid by the project members and reimbursed by the supervisor. Furthermore, these purchases may not exceed \$750.

### 1.3.8 Enterprise Constraints

The project currently has no enterprise constraints.



## 1.4 Naming Conventions and Terminology

Symbol	description
Asset	Any game object that appears in the game
A	Assumption
AR	Augmented Reality
DD	Data Definition
GD	General Definition
GPS	Global Positioning System
GS	Goal Statement
HA	Hazard Analysis
IM	Instance Model
LC	Likely Change
MG	Module Guide
MIS	Module Interface Specification
PoC	Proof of Concept
PS	Physical System Description
Puzzle State	The current variable values and UI appearance of the puzzle
R	Requirement
Rev	Revision
RS	Required Skill
Scene	The area of the game visible on the device
SRS	Software Requirements Specification
Software Eng	Software Engineering
TA	Teaching Assistant
TM	Theoretical Model
UI	User Interface
VR	Virtual Reality
VnV	Verification and Validation
WCAG	Web Content Accessibility Guidelines
XR	Extended Reality

## 1.5 Relevant Facts and Assumptions

The product will be developed under the following assumptions.

### **1.5.1 User Assumptions**

- A1 The user will provide consent for the app to access the camera on the device which is running the app.
- A2 The user will provide the app to access locations services and track where the user is while actively using the app.

### **1.5.2 Device Assumptions**

- A3 Devices will only have a maximum of one instance of the app running at any given time.
- A4 The device that the app is running on will have a functional camera.
- A5 The device that the app is running on will have a functional gyroscope, or equivalent technology to measure device tilt.
- A6 The device that the app is running on will have a functional GPS, or equivalent technology to measure device location.

## **2 Functional Requirements**

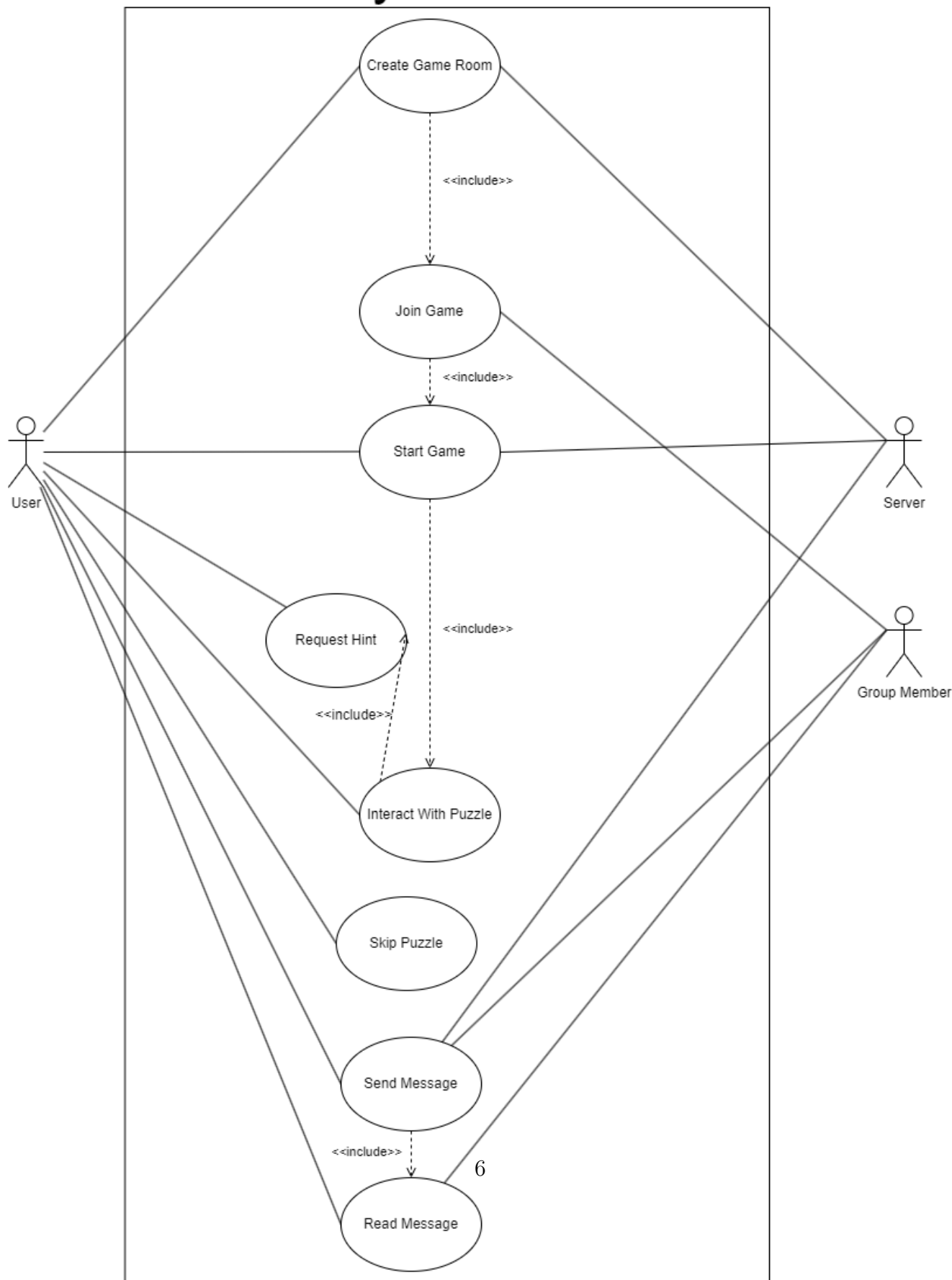
The following section will go over the necessary functional requirement information for the project, including use cases.

### **2.1 The Scope of the Work and the Product**

#### **2.1.1 Context of the Work**

The context diagram depicted below illustrates the interactions of the system with adjacent external systems and services.

# System



### 2.1.2 Individual Product Use Cases

**Use Case #1:** User creates a game room

**Trigger:** The user selects the create game room button

**Pre-condition:** N/A

**Outcome:**

1. The system creates a room for users to join, and automatically makes the user join the room
2. The user can choose the room settings
3. The server sends the data of the game room to all the users, to show that the room is available to join, and the capacity of the room, and the current users that have joined the room

**Use Case #2:** User Joins Game

**Trigger:** The user selects a room to join that has available capacity

**Pre-condition:** Another user has created a game room

**Outcome:**

1. The system links the user to the game room, and decreases the available capacity in the room
2. The server retrieves the data from the game room and updates the game room for all the users with the newly joined user's name, and decreases the visible room capacity

**Use Case #3:** User edits the game room settings

**Trigger:** The user of the game room clicks the edit game settings button

**Pre-condition:** The user created the game room

**Outcome:**

1. The system opens the game room settings page
2. The settings are displayed for the user
3. The user chooses the settings they want to change
4. The system updates the game room settings
5. The server informs the other group members of the settings changes
6. The user closes the settings page

**Use Case #4:** User exits the game room

**Trigger:** The user clicks the exit game room button

**Pre-condition:** The user is currently in a game room

**Outcome:**

1. The user is removed from the game room
2. The system updates the information of the game room
3. The server sends the updated information to each group member in the game room
4. The user is brought back to the main menu

**Use Case #5:** User starts the game

**Trigger:** The users in a game room click the start game button

**Pre-condition:** The game room has all group members present

**Outcome:**

1. The server launches the game and synchronizes the user and group member data.
2. The user performs a synchronization step to map the room and place the puzzles
3. The system shows the user and group members the first puzzles
4. The user and group members are each given a unique view on the puzzles they have available to them

**Use Case #6:** User interacts with a puzzle

**Trigger:** The user selects a puzzle to attempt

**Pre-condition:** The user has started a puzzle

**Outcome:**

1. The system displays an enlarged and interactable version of the puzzle
2. The user performs an action on the puzzle
3. The system registers the action, performs any necessary puzzle state changes and displays the new puzzle state to the user

**Use Case #7:** User requests a hint

**Trigger:** The user presses the request hint button

**Pre-condition:** The user has started a puzzle

**Outcome:**

1. The system displays the predetermined hint for the associated puzzle
2. The user reads the hint and presses the button to close the hint
3. The system closes the hint display

**Use Case #8:** User skips a puzzle

**Trigger:** The user hits the skip button

**Pre-condition:** The user is interacting with a puzzle

**Outcome:**

1. The system saves the state of the puzzle and closes the puzzle display, returning to the main scene

**Use Case #9:** User sends a message

**Trigger:** The user presses a communication button

**Pre-condition:** More than one user are in a game room

**Outcome:**

1. The system queries the server to determine the group members in the room
2. The system displays the other group members in the room to the user
3. The system displays the interface for the user to create the message
4. The user creates the message and hits the send button
5. The system uses the server to send the message group members devices and to create a message notification on those devices. It then notifies the sender that the message has been sent

**Use Case #10:** User reads a message

**Trigger:** The user clicks on a received message notification

**Pre-condition:** The user has received a message

**Outcome:**

1. The system removes the message notification and displays the message to the user
2. The user reads the message, then closes the message display window
3. The system returns the user to the main game scene

## 2.2 Functional Requirements

### 2.2.1 Create Game Room

- CG1. The system shall allow the user to create a game room

**Rationale:** The user must be able to group together with other users, creating the game room will allow the users to organize the group

- CG2. The system shall allow the user to enter a name for the game room

**Rationale:** The users must be able to uniquely identify the game rooms to allow the other users to find the room they want to join in case they have a group they wanted to solve the puzzles with

- CG3. The system shall allow the user to set the size of the game room

**Rationale:** The user should be allowed to limit the number of group members in the game room in case they want a small amount of group members or large amount of group members

- CG4. The system shall allow the user to put a password on the game room  
**Rationale:** The user should be able to allow or disallow any other users from joining except the ones they want in the case they are playing with friends
- CG5. The system shall redirect the user to a new screen when in a game room  
**Rationale:** The users in the room should have a display of the game room and the amount of group members in the room etc.

### 2.2.2 Join Game Room

- JG1. The system shall allow the user to join an existing game room with available capacity  
**Rationale:** The user should be able to join a room to group up with other users
- JG2. The system shall allow the user to enter a password for the room if another user set a password for the room  
**Rationale:** In the case of a game room having a password, the user should be able to have a space to enter the password
- JG3. When a user joins the game room the system shall update the game room information to add another member, and decrease the available capacity  
**Rationale:** When a user joins a game room, that information should be updated to let the users see the status of the room to inform them on which game room to join
- JG4. The system shall redirect the user to a new screen upon successfully connecting to a room  
**Rationale:** When a user joins the room, they should get redirected to a new page where they can see the information on the current game room and wait for the host to start the game
- JG5. The system shall not allow a user to join a game room that is at max capacity  
**Rationale:** If the game room is full, the room should not be able to take in more users because it is at max capacity

### 2.2.3 Edit Room Settings

- RS1. The system shall allow the user that created the game room to edit the settings of the game room  
**Rationale:** If a user input the incorrect settings when making a room, the user will have the ability to edit the settings
- RS2. The system shall populate all the changeable settings on the user's screen  
**Rationale:** The user should be able to see all the settings that are able to be changed

- RS3. The system shall allow the user to change the password of the room  
**Rationale:** The user should be able to change the password of the room to ensure security
- RS4. The system shall display the updated settings to the game members in the game room  
**Rationale:** When a setting changes the other users in the game room should be able to see the changes in case they will affect them

#### 2.2.4 Exit Room

- ER1. The system shall allow the user to leave the existing game room  
**Rationale:** If the user does not wish to stay in a game room, the user should be able to leave
- ER2. When a group member leaves the room the system shall update the game room and display the one less user  
**Rationale:** The system should update based on the current number of users
- ER3. When the a group member leaves the room the server shall update the screens of all group member's remaining in the room to show the new amount of users and the updated capacity  
**Rationale:** The server should update all the other users on the current capacity and number of users in the game room to inform them of any possible changes

#### 2.2.5 Start Game

- ST1. The system shall allow the user to start the game  
**Rationale:** The users should be able to start the game to play the puzzles
- ST2. The system shall load all the assets used to play the game  
**Rationale:** Loading all the assets at the start of the game will stop pauses during gameplay
- ST3. The system shall determine an order of the puzzles, and which group members can see which parts of the puzzles  
**Rationale:** In games with more than one puzzle, there needs to be an order to solve the questions. The system will determine a logical order, and assign which group members can see which parts of the puzzles
- ST4. The system shall reveal the available puzzles to the group members and commence the game  
**Rationale:** The group members should be able to view the puzzles they have access to



### 2.2.6 Puzzle Interaction

- PI1. The system shall allow users to perform all necessary actions on the puzzle to complete the puzzle  
**Rationale:** The user must be able to complete the puzzles in order to progress in the game
- PI2. The system shall display the new puzzle state after a user performs an action on a puzzle  
**Rationale:** The user must be able to see the results of their actions in order to decide how to proceed with the puzzle
- PI3. The system shall update the puzzle state of group members cooperating on an updated puzzle  
**Rationale:** When multiple group members are cooperating on a puzzle they must be able to see updates caused by the actions of one another
- PI4. The system shall notify the user when they have completed a puzzle  
**Rationale:** Users should know when they've completed a puzzle so they know they can move on. A notification at the end of a puzzle can also make completing it more satisfying for the user

### 2.2.7 Hint Request

- HR1. The system shall have two hints for each puzzle  
**Rationale:** If a user gets stuck on a puzzle they should have an alternative to giving up on the puzzle
- HR2. The system shall have a means of requesting a hint for the puzzle to be displayed  
**Rationale:** The user must be able to request the hint for the puzzle
- HR3. The system shall allow the user to close the hint display  
**Rationale:** The user should not have to keep the hint on the screen, as it may obstruct other functions

### 2.2.8 Skip Puzzle

- SP1. The system shall allow the user to skip a puzzle and return to the main scene  
**Rationale:** The user should not have to complete a puzzle to attempt a different one. They should be able to leave a puzzle and return to it later
- SP2. The system shall save the users progress on a skipped puzzle  
**Rationale:** If a user skips a puzzle they should be able to return to it without having to start over

SP3. The system shall notify any group members cooperating on the skipped puzzle that it has been skipped

**Rationale:** A group members work on a puzzle will typically depend on another group members work on the puzzle. They must be notified the puzzle has been skipped so they are not waiting on an action from the group member

### 2.2.9 Send Text Message

SM1. The system shall contain an interface that allows the user to compose text messages

**Rationale:** The user must be able to create their own custom messages to allow for optimal communication in puzzles and to create the most realistic social experience

SM2. The system shall allow the user to send a message to the users in the game room

**Rationale:** The primary goal of the game is to allow users to have social interaction from a remote setting. All users in the game must be able to interact with one another to complete this goal

SM3. The system shall allow the user to cancel a message before sending it

**Rationale:** If a user decides they do not want to send a message they must be able to return to the game without having to send something

SM4. The system shall notify the user once the message has been sent

**Rationale:** It is important that the user receive feedback that the message has been sent so there is no uncertainty or confusion for the user

### 2.2.10 Read Message

RM1. The system shall notify the user when a new message is received

**Rationale:** Communication is core to the game. If a user doesn't know when they receive a message good communication will not be maintained

RM2. The system shall allow the user to view messages

**Rationale:** Users must be able to see what other users are trying to communicate to them for social purposes and to complete puzzles

RM3. The system shall allow users to close a message

**Rationale:** Users must be able to return to the mains scene after viewing a message to continue game play

### 2.2.11 Voice Communication

VM1. The system shall allow users to communicate with one another through a voice chat

**Rationale:** To allow more fluid and more complex communication users should be able to communicate through a voice chat in addition to the text messaging.

- VM2. The system shall allow users to mute themselves in the voice chat at any time while they are in the game room  
**Rationale:** Users should not be forced to utilise the voice communication system as they may be in a location with unwanted background noise or where they cannot speak freely

### 2.2.12 Simon Says Puzzle

- SS1. The system shall display the coloured buttons to the host user at the start of the game  
**Rationale:** Users must be able to see puzzle assets that they need to interact with
- SS2. The system shall display the colour cube to all users other than the host at the start of the game  
**Rationale:** Users must be able to see puzzle assets that they need to interact with
- SS3. The system shall display the current Simon Says level to all users while the puzzle is active  
**Rationale:** The level indicates the current number of correct sequences the group has inputted, and shows progression through the puzzle
- SS4. The system shall display a sequence of colours on the colour cube when the puzzle is active  
**Rationale:** The sequence of colours must be shown in order so that the users can communicate and the individual with the coloured buttons can enter the correct sequence
- SS5. The system shall increase the Simon Says level once the user inputs colour sequence using the coloured buttons  
**Rationale:** Once the user enters the correct sequence, a new, longer sequence is generated which must be completed in order to reach the end of the puzzle
- SS6. The system shall display the puzzle as completed once level 3 is completed  
**Rationale:** The puzzle should be able to be completed in order for users to be able to progress through the escape room. Additionally, level 3 is the highest level since the puzzle is supposed to take several minutes to solve and the time it takes to reach this level is in accordance to that

### 2.2.13 Maze Puzzle

- MP1. The system shall randomly create a maze with a starting location, an end goal, and a ball that falls with gravity.  
**Rationale:** The puzzle should be highly re-playable, providing a different experience for users each time, and the maze needs an end goal to reach

- MP2. The system shall choose a user to be in control of rotating the maze  
**Rationale:** The goal of the puzzle is to have one person rotate the maze, and the rest of the users to guide the person in charge of rotating
- MP3. The system shall make the inside of the maze only visible to users that can't rotate the maze  
**Rationale:** If the user in charge of rotating could see inside the maze, it would defeat the whole purpose
- MP4. When the user in charge of rotating rotates their phone, the maze should rotate proportionally  
**Rationale:** The user should be able to rotate the maze easily and have a reasonable amount of positive feedback
- MP5. The system shall complete the puzzle when the ball reaches the end goal  
**Rationale:** The users should have a way to finish the maze

#### 2.2.14 Isometric Puzzle

- IP1. The system shall randomly assign letter and number combinations to every user in the game, based on a word the system chooses  
**Rationale:** Each user needs to have their own respective puzzle elements they can interact with, and there needs to be an end goal word.
- IP2. The system shall generate arrangements of blocks for each user that resemble letters and numbers based on the angle the user views them, and based on the letter and number combos each user was assigned  
**Rationale:** The goal of the puzzle is to complete a word based on letters in positions, the position given by the number they see, and the letter given by the letter they see
- IP3. The system shall have left and right buttons to switch between letter and number combinations  
**Rationale:** For groups of users with a size less than the amount of letters in the word, they need a way to switch between the available letters and numbers they have
- IP4. The system shall have an input field for users to enter their final guess into  
**Rationale:** The users need to way to guess what the word is to complete the puzzle
- IP5. When the word entered into the input field is the same as the word the system drew letter and number combinations from, the puzzle completes  
**Rationale:** There needs to be a puzzle completion condition

#### 2.2.15 Combination Puzzle

- CP1. The system shall randomly select a select a set of instructions and associated four digit combination from the available set of instruction-combination pairings

**Rationale:** The puzzle should be highly re-playable, providing a different experience for users each time

- CP2. The combination associated with each instruction set shall be the only possible four digit combination given the associated instructions

**Rationale:** There must be exactly one solution for each version of the puzzle in order for users to be able to solve it using the instructions.

- CP3. The system shall update the current combination display field for all users in the room with the correct digit when a user presses a button corresponding to the correct digit of the combination

**Rationale:** When a user enters a correct digit of the combination, all users should see this reflected on the keypad interface so that they know that number was correct and can progress in the puzzle

- CP4. The system shall indicate to all users when a user has pressed a keypad button associated with an incorrect digit of the combination

**Rationale:** Users must be able to tell when they've entered an incorrect digit so that they do not continue trying this digit

- CP5. The system shall reset the combination display field for all users when a user has pressed a keypad button associated with an incorrect digit of the combination

**Rationale:** Resetting the combination entry on each incorrect input encourages more communication as there is a greater penalty for incorrect inputs. It also helps prevent spam guessing all possible combinations

- CP6. The system shall complete the puzzle once all four digits of the combination have been entered correctly, displaying a "Correct" message on the keypad interface

**Rationale:** When the puzzle has been successfully completed users must be able to move on to the next puzzle. The puzzle should have a visual indicator of this so users know to start looking for the next task.

### 2.2.16 Wires Puzzle

- WP1. The puzzle shall display a box of four coloured wires to the user designated at the active user

**Rationale:** A single user must be able to see the puzzle asset that they can interact with

- WP2. A set of four pairs of coloured dots shall be displayed for all users not designated as the active user

**Rationale:** The remaining users must be able to see the puzzle asset containing the solution to the puzzle

- WP3. The four coloured wires must be able to be manipulated by the user such that their tip follows the movement of the users finger while they are dragged across the screen  
**Rationale:** To complete the puzzle, the wires must be able to be dragged into their designated positions
- WP4. When a wire tip is dragged by a user, the position of the wire must remain in place when the wire is released  
**Rationale:** When the user stops dragging, the wires must not reset to their original positions to allow the puzzle to be completed
- WP5. The four pairs of coloured dots shall match up with the proper solution to the wires puzzle. Each pair of dots shall indicate a correct connection, with the colour of the node as the dot on top and the colour of the wire as the dot on the bottom  
**Rationale:** The active part of the puzzle must align with the hint given to the other users for the puzzle to be able to be completed
- WP6. When the wires are connected to the nodes in an improper sequence, no change occurs  
**Rationale:** The puzzle must only be able to be completed with a single correct sequence
- WP7. When the wires are connected to the nodes in the correct sequence, the puzzle completes  
**Rationale:** The correct sequence must complete the puzzle
- WP8. When the puzzle is spawned and inactive, its lights will be off  
**Rationale:** The initial state of the puzzle will have its lights off to indicate to the user not to interact with it yet
- WP9. When the puzzle is set to become active, its lights will turn red  
**Rationale:** The lights will turn red to indicate that the puzzle is active but not yet completed
- WP10. When the puzzle is completed, its lights will turn green  
**Rationale:** The lights will turn green to indicate to the user that the puzzle has been completed

## 2.3 Formal Specification

$$(\forall r \in rooms | (\sum_{user \in r} 1) \leq r.roomCapacity)$$

The users that exist in a room shall be within the capacity of the room

$$(\forall a, b \in puzzles | (a \subset b) \implies (a \implies b))$$

For all the puzzles in the room, if puzzle b depends on a, b will come after a

Likely	Unlikely
CG4, JG2, RS3, HR1, HR2, HR3, SP1, SP2, SP3, SS1, SS2, SS3, SS4, SS5, SS6, MP1, MP2, MP3, MP4, MP5, IP1, IP2, IP3, IP4, IP5, CP1, CP2, CP3, CP4, CP5, CP6, WP1, WP2, WP3, WP4, WP5, WP6, WP7, WP8, WP9, WP10	CG1, CG2, CG3, CG5, JG1, JG3, JG4, JG5, RS1, RS2, RS4, ST1, ST2, ST3, ST4, PI1, PI2, PI3, PI4, SM1, SM2, SM3, SM4, RM1, RM2, RM3, VM1, VM2, Formal Specification

Table 3: Non-Functional Likely/Unlikely to Change Table

## 2.4 Functional Requirements That Are Likely/Unlikely to Change

- CG1, CG2, CG3, CG5 are all unlikely to change as they are basic properties/needs of a game room that users are able to join
- JG1, JG3, JG4, JG5 are all unlikely to change as capacity updates and scene changes when joining a room will always be necessary features
- RS1, RS2, RS3 are all unlikely to change as the capability to edit room settings after creating a room adds convenience to the game that will be useful when performing the study and for casual players of the game
- CG4, JG2, RS3 are fairly likely to change as it is not necessary that a game room have a password. It may be decided that this requirement is not needed
- ST1, ST2, ST3, ST4 are all unlikely to change as the game start procedure is core to the identity of the game as a multiplayer augmented reality communication based game
- PI1, PI2, PI3, PI4 are all unlikely to change as it is unlikely the game will be modified such that users do not need to be able to interact with their puzzles and see the result of these interaction
- HR1, HR2, HR3, SP1, SP2, SP3 are all likely to change as hints and skips have been primarily implemented to facilitate the study. If the game changes to become more time-based and competitive these features may change or be removed
- SM1, SM2, SM3, SM4, RM1, RM2, RM3, VM1, VM2 are all unlikely to change as they are the core properties of text and voice communication which are fundamental to the game which is based on communication. Requirements may however be added to these sections if these features are expanded upon in the future
- All individual puzzle requirements (SS, MP, IP, CP, WP) are likely to changes as the puzzles in the game may be swapped out or heavily modified in the future. The

individual puzzles are the most variable part of this game and could easily be changed without altering the identity and purpose of the game

- The formal specification is unlikely to change as a room should never exceed its capacity

## 3 Non-functional Requirements

The following section will describe the non-functional requirements of the application.

### 3.1 Look and Feel Requirements

LF1. The system shall adjust and scale to fit the physical screen size

*Fit criterion: The display should cover the entire screen and none of it should be cutoff*

LF2. The system shall have fonts and colours that will allow users to easily read the text

*Fit criterion: The system shall abide by the Web Content Accessibility Guidelines (WCAG) 2.1 AA standard (Cooper, 2023) for fonts and colours*

### 3.2 Usability and Humanity Requirements

UH1. The system shall be accessible by any supported mobile devices using up to date software

*Fit criterion: The system should allow the application to be launched displaying the initial menu screen on the devices listed in solution constraint 1.3.1 with an operating system satisfying OE1*

UH2. The system shall notify the user if there is no network, or they get disconnected

*Fit criterion: The system should produce a notification when network connection is lost*

UH3. The system shall reasonably be able to be navigated

*Fit criterion: The system shall abide by the WCAG 2.1 AA standard (Cooper, 2023) for accessibility*

UH4. The system shall prompt the user to re-enter an area with internet connection when it detects there is no network

*Fit criterion: The system should produce a notification when network connection is lost*

UH5. The system shall prompt the user if their current environment is unsuitable to use the application

*Fit criterion: The system will produce a pop up notification during startup to let user know of issues with their environment, as well as if after startup the user environment becomes unsuitable. This may include lighting or distance to spawned puzzles*



UH6. The system shall allow users to reconnect to their game session if they become disconnected

*Fit criterion: The system will prompt the user to reconnect to their game session through a reconnect button, which upon pressing will reconnect to the user's previous session*

UH7. The system shall inform users who are using an old version of the application to check for updates and use the latest version

*Fit criterion: The system will produce a notification to let the user know to update their application to the latest version, in order to play online*

### 3.3 Performance Requirements

PR1. The system shall respond to user interaction within 5 seconds

*Fit criterion: The system shall respond within a reasonable amount of time to a user's request*

PR2. The system shall be available for users at any time or display the reasoning for the system outage

*Fit criterion: The system will be accessible 24 hours a day unless the server is undergoing maintenance or experiencing an outage. If there is maintenance or an outage an error message is displayed stating the respective issue*

### 3.4 Operational and Environmental Requirements

OE1. The system shall be available on Android 11 or above and iOS 16 or above

*Fit criterion: The system shall be able to be downloaded from the Google Play Store and Apple Store on devices that have Android 11 or iOS 16 (or later) operating systems*

### 3.5 Maintainability and Support Requirements

MS1. The system shall be well documented and kept up to date, and have updated design documents

*Fit criterion: All changes in the system design are to be reflected in the documents in the GitHub*

MS2. The source code shall be well commented for future developers

*Fit criterion: The system will have author names and dates present in developer code, as well as descriptions of what every system module and function do. An external developer must be able to understand the function of any part of the code from the comments.*

### 3.6 Security Requirements

SR1. The system shall keep user data private

*Fit criterion: Game room passwords or IP addresses should not be displayed and should only be accessible to the internal system functions. No external data queries to the database or the system should be able to retrieve this information.*

SR2. The users will only be allowed to see limited data. Unnecessary data will not be displayed to the user as providing the user with data related to implementation or other users could leave other users or the system vulnerable to malicious attacks

*Fit criterion: The system shall only show users any data required in order to play the game*

### 3.7 Cultural Requirements

CR1 The system shall not use any offensive images, text, or sound that could offend any religious or culture groups *Fit criterion: The game shall comply with the Entertainment Software Rating Board (ESRB) “E for Everyone” rating standards ([Entertainment Software Rating Board, 2024](#))*

CR2 The system shall use Canadian English

### 3.8 Legal Requirements

LR1 The system shall not infringe on the rights of any person(s), and shall not use any assets that infringe on copyright claims. When using open source resources, the developers shall give appropriate credit *Fit criterion: All assets used are open source with credit provided to authors or free access. No copyrighted assets are used*

### 3.9 Health and Safety Requirements

HS1 The system shall give a warning to the user to be aware of their surroundings while using the system, and to not bump into any objects or obstacles in their path

### 3.10 Non-Functional Requirements That Are Likely/Unlikely to Change

- LF1 and LF2 are unlikely to change as they are basic visual requirements to make the app usable
- UH1, UH2, UH3, UH4, UH5, UH6, UH7 are unlikely to change as they are navigation, environment and network based requirements that are required for reasonable usability of the game

Likely	Unlikely
PR1, OE1, CR2	LF1, LF2, UH1, UH2, UH3, UH4, UH5, UH6, UH7, PR2, MS1, MS2, SR1, SR2, CR1, LR1, HS1

Table 4: Non-Functional Likely/Unlikely to Change Table

- PR1 is likely to change. As game systems improve the maximum response time should be able to be reduced significantly
- PR2 is unlikely to change as users should always be notified when the game is not available
- OE1 is likely to change as newer devices come out and the system evolves to meet those specifications
- MS1 and MS2 are unlikely to change as documentation should always remain up to date to facilitate maintainability
- SR1 and SR2 are unlikely to change as user data confidentiality and protection should always remain in place
- CR1 is unlikely to change as the game should never contain offensive content
- CR2 is likely to change as the game may be expanded to include more languages
- LR1 is unlikely to change as the game must remain legally viable
- HS1 is unlikely to change as this warning is needed to protect users

## 4 Traceability Matrix

	LF1	LF2	UH1	UH2	UH3	PR1	PR2	OE1	MR1	MR2	SR1	SR2	CR1	CR2	LR1	HS1
CG1	X		X		X	X	X	X			X	X		X		
CG2	X	X	X		X	X	X	X				X	X	X		
CG3	X		X	X	X	X	X	X				X		X		
CG4	X	X	X		X		X	X			X	X		X		
CG5	X		X	X	X			X			X	X				
JG1				X		X	X	X				X				
JG2	X	X	X		X		X	X			X	X		X		
JG3	X	X	X		X			X								
JG4	X		X	X	X			X			X	X				
RS1	X	X			x			X						X		
RS2	X	X	X		X			X				X				
RS3	X	X	X		X			X			X	X				
RS4	X	X	X		X			X								
RS5	X	X	X	X	X	X		X			X		X	X		
RS6	X	X	X		X		X	X								
ER1	X	X	X	X	X		X	X			X			X		
ER2	X	X	X		X			X			X			X		
ER3	X	X	X		X			X			X			X		
ST1	X	X	X		X	X	X	X								X
ST2		X	X					X	X	X			X	X	X	
ST3	X	X	X					X								
ST4								X	X	X						
ST5	X		X				X	X								
ST6	X		X					X								

Table 5: Traceability Matrix

	LF1	LF2	UH1	UH2	UH3	PR1	PR2	OE1	MR1	MR2	SR1	SR2	CR1	CR2	LR1	HS1
PI1	X	X	X	X	X	X	X	X				X	X	X		X
PI2	X	X	X		X	X	X	X			X	X	X	X		X
PI3	X	X	X		X	X	X	X				X	X	X		
PI4	X	X	X			X	X	X				X	X	X		
PI5	X	X	X	X		X	X	X			X	X	X	X		
PI6	X	X	X	X		X	X	X					X	X		
PI7	X	X	X	X		X	X	X					X	X		
HR1							X	X	X	X	X	X			X	
HR2	X	X	X	X	X	X	X	X					X	X		
HR3	X	X		X	X	X	X	X					X	X		
SP1	X	X		X	X	X	X	X					X	X		
SP2						X	X	X	X	X	X	X			X	
SP3	X	X		X	X	X	X	X					X	X		
SM1	X	X	X	X	X		X	X				X		X		
SM2	X	X	X	X	X		X	X				X		X		
SM3	X	X	X	X	X	X	X	X			X	X		X		
SM4	X	X	X	X	X	X	X	X			X	X		X		
SM5	X	X	X	X		X	X	X			X	X	X	X		
RM1	X	X	X	X	X	X	X	X			X	X	X	X		
RM2	X	X	X	X	X	X	X	X			X	X	X	X		
RM3	X	X	X	X	X	X	X	X			X	X	X	X		
SS1																
SS2																
SS3																
SS4																
SS5																
SS6																
MP1																
MP2																
MP3																
MP4																
MP5																
IP1																
IP2																
IP3																
IP4																
IP5																
CP1																
CP2																
CP3																

Table 6: Traceability Matrix

## 5 Project Issues

The following section will go over the current issues for the project. It will describe open issues, current off-the-shelf solutions, new problems, and tasks that need to be worked on.

## 5.1 Open Issues

There are no open issues for this project.

## 5.2 Off-the-Shelf Solutions

The following section will go over existing off-the-shelf solutions and reusable components from these solutions.

### 5.2.1 Ready-Made Products

There are several already made products that have similar aspects to the project however they are missing some key aspects. Examples include ARctic Escape and We Were Here. Similarly to this project, ARctic Escape is an AR game that encourages social interaction. Users must work together to solve puzzles and complete the game (Knoll et al., 2023). The main difference between this project and ARctic Escape is that in ARctic Escape, users must be in the same area/room to play the game, however, Mac-AR is based around users being able to interact and play the game remotely. Continuing on, We Were Here is a two player game in which users must work together to solve difficult puzzles. The game can be played remotely between the two players and there is an in-game communication system that allows the users to talk to each other. We Were Here, however, does not have any AR component to it, which is one of the main aspects of this project.

### 5.2.2 Reusable Components

Plenty of libraries and other frameworks that already exist can be used in the project. AR Foundation is one such framework that can be used through Unity. It supports features such as device tracking and 2D image tracking which can be used for tracking the movement of users phones while playing the game. Additionally, there are the Zed-Rig plugins that Unity supports. These can help provide functionality for AR object placement as well as spatial mapping. Continuing on, ARctic Escape which was mentioned in the Ready-Made Products, was able to have spatial mapping through a Snapchat Lens. This allowed for users to record their surroundings and have the room they were in mapped and able to be used in the AR environment.

### 5.2.3 Products That Can Be Copied

Currently, the only known product that is similar to this project and can be copied without copyright issues is ARctic Escape, however, this game is implemented much differently than the plan for this project. ARctic Escape was implemented as a Snapchat Lens, however, the goal for this project is to create a standalone application through the Unity framework.

## **5.3 New Problems**

The following section will describe potential problems of the application's creation. It will go over the effects on the current environment and installed system, as well as user problems and other follow-up problems.

### **5.3.1 Effects on the Current Environment**

The application will affect the way people who are isolated/remote are able to connect with others.

### **5.3.2 Effects on the Installed Systems**

The application will not have any effect on the mobile devices it is installed on. It will act as a standalone application on the device and should not affect any other parts of the device.

### **5.3.3 Potential User Problems**

A potential user problem may be the difficulty in solving in-game puzzles/tasks. If tasks are too difficult to complete, it may result in users wanting to quit the game, and therefore resulting in a decreased level of social interaction. Additionally, if the tasks are too easy to solve, this may also reduce the level of social interaction as users will not need to communicate as much. This may also result in users not enjoying the play.

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

It is assumed that users will be able to use the application in an area with internet connection, however, it is not unusual that people may have Internet issues, which may cause them to lose connection to the Internet. This results in a problem as without Internet connection users will disconnect from the host server, causing them to not be able to play the game or interact with their partner.

### **5.3.5 Follow-Up Problems**

Other problems include if libraries used in the creation of the application get updated while the project is ongoing. This may result in certain in-game functionality not behaving as intended. Another problem that may arise is if users are able to find ways to solve tasks by themselves, without the need for interaction with others. The main purpose of the application is to have social interaction between users, and this would result in users not needing to communicate with each other.

## **5.4 Tasks**

The following section will go over the planning of the project.

### 5.4.1 Project Planning

The project schedule will follow the deadline for the deliverables outlined in the SFWRENG 4G06 course outline.

Revision #	Deliverable	Due Date
Revision 0	Hazard Analysis	October 20, 2023
	V&V Plan	November 3, 2023
	PoC Demo	November 13-24, 2023
	Design Document	January 17, 2024
	Revision 0 Demo	February 5-16, 2024
	V&V Demo	March 6, 2024
Revision 1	Final Demo	March 18-29, 2024
	EXPO Demo	April, 2024
	Final Documentation	April 4, 2024

### 5.4.2 Planning of the Development Phases

The development of the project will be split into three phases:

- Creation of a proof of concept and documentation
- Initial development of the application with consideration to PoC
- Refinement of application and documentation after user feedback

The creation of the proof of concept demonstration will aid in overcoming the main challenges of the system. It will demonstrate the ability for multiple users to interact in AR, and provide the documentation for the future application to be developed.

The initial development of the application will take the design of the PoC and improve on the concepts. Four puzzles will be implemented, and the multiplayer functionalities will also be implemented. The Save/Load game functional requirements and the hint and skip system will not be required for revision 0 as they are not crucial parts of the overall system. The stakeholders will be able to see the implementation and provide feedback.

The refinement will take all the user feedback and improve on the initial design. There is no expectation for major features to be developed at this stage, just improvements on the existing system and the addition of potential stretch goals.

## 5.5 Migration to the New Product

The following section will go everything required for migration to the application.



### **5.5.1 Requirements for Migration to the New Product**

There are no requirements for migration for the application.

### **5.5.2 Data That Has to Be Modified or Translated for the New System**

No data needs to be modified or translated for the application.

## **5.6 Risks**

The main risks of the project are:

- Users losing awareness of their surroundings and potentially getting injured
- Server overload could cause lag in gameplay, resulting in an unpleasant user experience
- Potential data loss if the server storing saved game data malfunctions/breaks
- Failure to meet deadlines will cause setbacks for the project. In the event of this, some requirements might not be met, and documents will have to be modified to reflect this.

## **5.7 Costs**

There is no financial costs associated with the development of the application, and there is no cost for users downloading and using the application. Furthermore, the time to development time is estimated to be six months, with work being divided amongst four developers.

## **5.8 User Documentation and Training**

The following section will go over any training and documentation that is needed for users, as well as waiting room ideas and ideas for implementation

### **5.8.1 User Documentation Requirements**

The users will have a user help document available to them within the application in the form of a gameplay menu. It will contain information on how to use the application, as well as any instructions regarding interacting with puzzles/tasks.

### **5.8.2 Training Requirements**

The users will have a "tutorial" when they first start the game that will teach them how to use the application. It will show the users how to use AR technology, and give them a simple puzzle to solve.

## 5.9 Waiting Room

There are currently no requirements or functionalities that are not being included in the initial release. Future ones may be added later on.

## 5.10 Ideas for Solutions

During the requirements and idea generation phases, additional ideas were generated on how to implement the solution.

- Form
  - Since the application will be developed using Unity framework, the idea of using Unity based libraries and plugins for AR framework such as AR Foundation and Zed-Rig.

# 6 Appendix

This section will be used to capture miscellaneous information that couldn't easily be fit into the rest of the document.

## 6.1 Reflection

A new and unique set of skills will be required for the completion of this project. It is hoped that the entire team will have an exposure to everything listed below, but the team members in charge of fostering the skill and development of the sections that require it are also listed.

**RS1 Mobile App Development:** Developing a mobile application presents unique challenges that aren't present when developing a standard computer application. Dealing with limited memory and processing power, navigating the licensing and system requirements for different devices, and fully utilizing the tools embedded in phones are all skills that will need to be developed over the course of the project.

- **Main Team Members:** Sam, Matthew

**RS2 Augmented Reality:** Integrating AR into an application requires an understanding of how to get a piece of software to integrate with the real world. Team members will need to learn how to make the tool process the readings from various video and location instruments, calculate how to place objects into a 3D space, and give the user the tools to manipulate the world through their augmented view.

- **Main Team Members:** Kieran, Ethan, Sam, Matthew

RS3 **Communication Services:** Allowing different devices, running on different types of software to communicate with each other over long distances is a vital part of capturing the collaborative aspect of the project. The team will need to investigate how to accomplish this networking.

– **Main Team Members:** Kieran, Matthew

RS4 **Location Services:** Tracking the location of the user and integrating nearby landmarks in real time adds additional depth to the tool. A vital skill will be getting the tool to interface with existing location services.

– **Main Team Members:** Ethan, Sam

RS5 **Game Design:** Getting a large and diverse user base is essential for the collaborative aspect of the tool. Researching what makes a game fun to play and striking a balance between the difficulty and complexity of the puzzles is what will drive this engagement.

– **Main Team Members:** Kieran, Ethan

## 6.2 Symbolic Parameters

There are currently no symbolic parameters that need to be recorded for this document. Any future symbols will be recorded in the following table:

Symbol	Value	Description

## References

Michael Cooper. Wcag 2.1. <https://www.w3.org/TR/WCAG21/>, 2023.

Entertainment Software Rating Board, 2024. *ESRB Ratings Guide*, 2024.

Theodore Knoll, Amna Liaqat, and Andres Hernandez. Arctic escape: Promoting social connection, teamwork, and collaboration using a co-located augmented reality escape room. <https://dl.acm.org/doi/10.1145/3544549.3585841/>, 2023.

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