

# CSC 431

## Above The Grave Game

### Software Requirements Specification (SRS)

#### Team 05

Drew Barnes	Requirements Engineer
Daniel Barry	Scrum Master
Parth Sharma	System Architect

# Version History

Version	Date	Author(s)	Change Comments
1.0	02/23/22	Drew Barnes Daniel Barry Parth Sharma	First Draft
2.0	04/30/23	Drew Barnes Daniel Barry Parth Sharma	Updates based on Feedback

# Table of Contents

<b>Version History</b>	<b>2</b>
<b>Table of Contents</b>	<b>3</b>
<b>Table of Tables</b>	<b>4</b>
<b>Table of Figures</b>	<b>5</b>
1. System Requirements	
1.1. Functional Requirements	6
1.1.1 Game Menu	6
1.1.2 Play Game	6
1.1.3 View Card	7
1.1.4 View Card Collection	7
1.1.5 Options	8
1.1.6 Select Card	8
1.1.7 Victory/Lose Condition	9
1.1.8 Battle UI	9
1.1.9 Useful Notifications	9
1.2. Non-Functional Requirements	10
1.2.1 Efficient Loading	10
1.2.2 Usability	10
1.2.3 Network Security	10
1.2.4 User Data/Card Collection Privacy	10
2. System Constraints	11
2.1. Tool Constraints	11
2.1.1 Mobile Application Framework Constraint	11
2.1.1 Source Code Editor	11
2.2. Language Constraints	11
2.2.1 Backend and FrontEnd Language	12
2.3. Platform Constraints	12
2.3.1 Mobile Application Platform	
2.4. Hardware Constraints	12
2.4.1 Mobile Device	
2.5. Network Constraints	13
2.5.2 Internet Connection	
2.6. Deployment Constraints	13
2.6.1 Unity Development	
2.7. Transition & Support Constraints	13

2.7.1 Augmented Reality Maintenance	
2.7.2 End of Life	
2.8. Budget & Schedule Constraints	13
2.8.1 Project Timeline	
2.8.2 Budget Contract	
3. Requirements Modeling	
3.1 User and System	
15	
4. Evolutionary Requirements	16
4.1 Functional Requirements	16
4.1.2 Sound Effects within our Battle UI	
4.2. Non-Functional Requirements	17
4.2.1 Battle UI Sound Effect Speed and Duration	

# Table of Tables

## System Requirements

Functional Requirements	1.1
Game Menu	1.1.1
Play Game	1.1.2
View Card	1.1.3
View Card Collection	1.1.4
Options	1.1.5
Select Card	1.1.6
Victory/Lose Condition	1.1.7
Battle UI	1.1.8
Useful Notifications	1.1.9
Non-Functional Requirements	
Efficient Loading	1.2.1
Usability	1.2.2
Network Security	1.2.3
Frame Rate	1.2.4
Sound Quality	1.2.5
User Data/Card Collection Privacy	1.2.6

## System Constraints

Tool Constraints	2.1
Mobile Application Framework Constraint	2.1.1
Source Code Editor	2.1.2
Language Constraints	2.2
BackEnd and FrontEnd Language	2.2.1
Platform Constraints	2.3
Mobile Application Platform	2.3.1
Hardware Constraints	2.4
Mobile Device	2.4.1
Network Constraints	2.5
Internet Connection	2.5.1
Deployment	2.6
Unity Deployment	2.6.1
Transition and Support	2.7
Augmented Reality Maintenance	2.7.1
End of Life	2.7.2

Budget and Schedule	2.8
Project Timeline	2.8.1
Budget Constraint	2.8.2

## **Evolutionary Requirements**

Functional Requirements	4.1
Sound Effects within our Battle UI	4.1.1
Non-Functional Requirements	4.2
Battle UI Sound Effect Speed and Duration	4.2.1

# Table of Figures

## Requirements Modeling

User and System	3.1
-----------------	-----

# 1. System Requirements

## 1.1. Functional Requirements

### 1.1.1. Game Menu

Title	Navigating game features + Signing In
Description	This will be a form allowing users to navigate to which aspect of the game they want to interact with upon initial startup.
Priority	0
Precondition(s)	The user must have installed the application and opened it to the menu booting.
Basic Flow	<ul style="list-style-type: none"><li>• User opens application</li><li>• User is presented with game menu/options</li><li>• Returning user/Logged In user may have different options</li><li>• User is redirected to whichever function they touch as a UI button on their smartphone device</li></ul>
Postconditions(s)	User is prompted to enter preferences information and the user has a working account.
Use Case Diagram	3.1

### 1.1.2. Play Game

Title	Loading into Above The Grave Game
Description	The user taps the "Play Game" button which turns on the camera of the user's device to start the game.
Priority	0
Precondition(s)	The user must have the game installed and given the app permission to use the device's camera.



Basic Flow	<ul style="list-style-type: none"> <li>• User opens application</li> <li>• User chooses the "Play Game" option of the game menu</li> <li>• Permission is asked to use camera if it has not been so earlier, then camera turns on to look for image targets</li> </ul>
Postconditions(s)	The user is directed back to the game menu
Use Case Diagram	3.1

### 1.1.3. View Card

Title	View Your Trading Card's 3D model
Description	The user is able to view their card's 3D model without being in game mode
Priority	1
Precondition(s)	The user must have the game installed and given the app permission to use the device's camera.
Basic Flow	<ul style="list-style-type: none"> <li>• User opens the application</li> <li>• User chooses the "View Card" option of the game menu</li> <li>• Permission is asked to use camera if it has not been so earlier, then camera turns on to look for image targets</li> <li>• 3D model is displayed on the Trading Card aka the ImageTarget</li> </ul>
Postconditions(s)	The user is directed back to the game menu
Use Case Diagram	3.1

### 1.1.4. View Card Collection

Title	View Card Collection
Description	The user will view all cards that have been scanned by the app. These will be documented and the user will be able to look at the models and descriptions/moves without the need for the card.
Priority	0

Precondition(s)	The user needs to open the app and choose the option "View Card Collection"
Basic Flow	<ul style="list-style-type: none"> <li>• The user opens the app</li> <li>• The user views the game menu</li> <li>• The user chooses "View Card Collection" which takes the user to the cards they have scanned within the app.</li> </ul>
Postconditions(s)	The user is directed back to the game menu
Use Case Diagram	3.1

### 1.1.5. Options

Title	View options and change options of game
Description	The user browses a list that contains various options and the ability to change them
Priority	1
Precondition(s)	The user opens the app and views the game menu with UI Button showing options
Basic Flow	<ul style="list-style-type: none"> <li>• The user opens the app and sees the game menu</li> <li>• The user clicks the "Options Button"</li> <li>• The user enters the options list</li> <li>• The user toggles options and returns to the game menu.</li> </ul>
Postconditions(s)	The user is provided with options to toggle. Then they can exit and return to the main menu.
Use Case Diagram	3.1

### 1.1.6. Select Card

Title	User selects card to use in battle
Description	The user has pressed "Play Game" and once the image target has been recognized the user then confirms that this is the card they will use in battle.
Priority	0
Precondition(s)	The user chooses to "play game" on the game menu and then puts an image target in front of the camera.

Basic Flow	<ul style="list-style-type: none"> <li>• The user opens the app and sees the game menu</li> <li>• The user clicks "Play Game"</li> <li>• The user puts their trading card in front of the camera.</li> <li>• The user then confirms this card as their selection</li> </ul>
Postconditions(s)	The user is prompted to allow player 2 to confirm their card and image target. After which the card game will start.
Use Case Diagram	3.1

### 1.1.7. Battle UI

Title	Viewing Moves of Card
Description	The user browses a list of moves of their respective card to utilize in the game.
Priority	1
Precondition(s)	The user needs to have a card already selected to load in the move set
Basic Flow	<ul style="list-style-type: none"> <li>• The user opens the app, plays the game, and selects a card</li> <li>• The user clicks the move they want to use and it is used in the game</li> </ul>
Postconditions(s)	The user then sees the stats such as health change based on the move they chose against their opponent.
Use Case Diagram	3.1

### 1.1.8. Victory/Lose Condition

Title	Show Win/Lose Condition once game has played its course
Description	The user is shown if they have won or loss their match and are able to see their stats in the game
Priority	0
Precondition(s)	The user needs to open the app and have played the game through at least once.

Basic Flow	<ul style="list-style-type: none"> <li>• The user opens the app and is placed on the game menu screen</li> <li>• The user clicks play game and plays a game through</li> <li>• The user either loses or wins any given game</li> <li>• The win/lose menu is displayed with stats for the user to learn from.</li> </ul>
Postconditions(s)	The user is able to view the results and exit back to the main menu
Use Case Diagram	3.1

### 1.1.9. Useful Notifications

Title	Useful Notifications
Description	Users will be able to see when new cards are added to the game
Priority	4
Applicable FR(s)	3.1

## 1.2. Non-Functional Requirements

### 1.2.1. Efficient Loading

Title	Efficient Loading and Camera opening
Description	All game menu UI Button functions and the camera for the AR Image Targeting should boot in less than 500 ms
Priority	3
Applicable FR(s)	1.1.2, 1.1.3, 1.1.4, 1.1.5

### 1.2.2. Usability

Title	Usability
Description	The app must include clear instructions to play and a friendly user interface.
Priority	4
Applicable FR(s)	1.1.4

### 1.2.3. Network Security

Title	Network Security
Description	The application will be protected from misuse with any potentially online aspect implemented
Priority	1
Applicable FR(s)	1.1.4, 1.1.2

### 1.2.4. Frame Rate

Title	Frame Rate
Description	The application should maintain a consistent frame rate of at least 70 frames per second (FPS)
Priority	3
Applicable FR(s)	1.1.4, 1.1.2

### 1.2.5. Sound Quality

Title	Sound Quality
Description	The application should provide high-quality sound effects with a bit rate of 185 kbps or higher
Priority	3
Applicable FR(s)	1.1.4, 1.1.2

### 1.2.6. User Data/Card Collection Privacy

Title	User Data/Card Collection Privacy
Description	All user data will be encrypted to protect them from potential bad actors
Priority	1
Applicable FR(s)	1.1.4

## 2. System Constraints

### 2.1. Tool Constraints

#### 2.1.1. Mobile Application Framework Constraint

Title	Mobile Application Framework Constraint
Description	We will be using Unity for our application, because it is an extensive cross-platform game engine and it supports both iOS and Android.
Priority	2

#### 2.1.2. Source Code Editor

Title	Source Code Editor
Description	The IDE where we write our scripts will be on Visual Studio Code.
Priority	2

### 2.2. Language Constraints

#### 2.2.1. BackEnd and FrontEnd language

Title	BackEnd and FrontEnd language
Description	Unity is a native C++ game engine. We are going to write our code in C#.
Priority	2

### 2.3. Platform Constraints

#### 2.3.1. Mobile Application Platform

Title	Mobile Application Platform
Description	Our supported platforms will be iOS and Android.
Priority	3

### 2.4. Hardware Constraints

### 2.4.1. Mobile Device

Title	Mobile Devices: iPhone and Galaxy
Description	Certain hardware features on mobile devices are required for our application. Having a working camera on the user's device is obviously required.
Priority	2

## 2.5. Network Constraints

### 2.5.1. Internet Connection

Title	Internet Connection
Description	Internet Connection is only required if the user chooses to play with another user on a different device. Internet connection is not required for the option menu and viewing cards.
Priority	2

## 2.6. Deployment Constraints

### 2.6.1. Unity Deployment

Title	Unity deployment
Description	We will deploy our application with Unity to bring our game available to app stores.
Priority	3

## 2.7. Transition & Support Constraints

### 2.7.1. Augmented Reality Maintenance

Title	Augmented Reality Maintenance
Description	Effective maintenance is essential and will be iterative with our Augmented Reality feature to prevent unplanned downtime and maximize performance.
Priority	2

### 2.7.2. End of Life

Title	End of life
Description	By the end of our semester, the members of our team will no longer work on Above the Grave.
Priority	1

## 2.8. Budget & Schedule Constraints

### 2.8.1. Project Timeline

Title	Project Timeline
Description	Our application would be deployed before the semester ends.
Priority	3

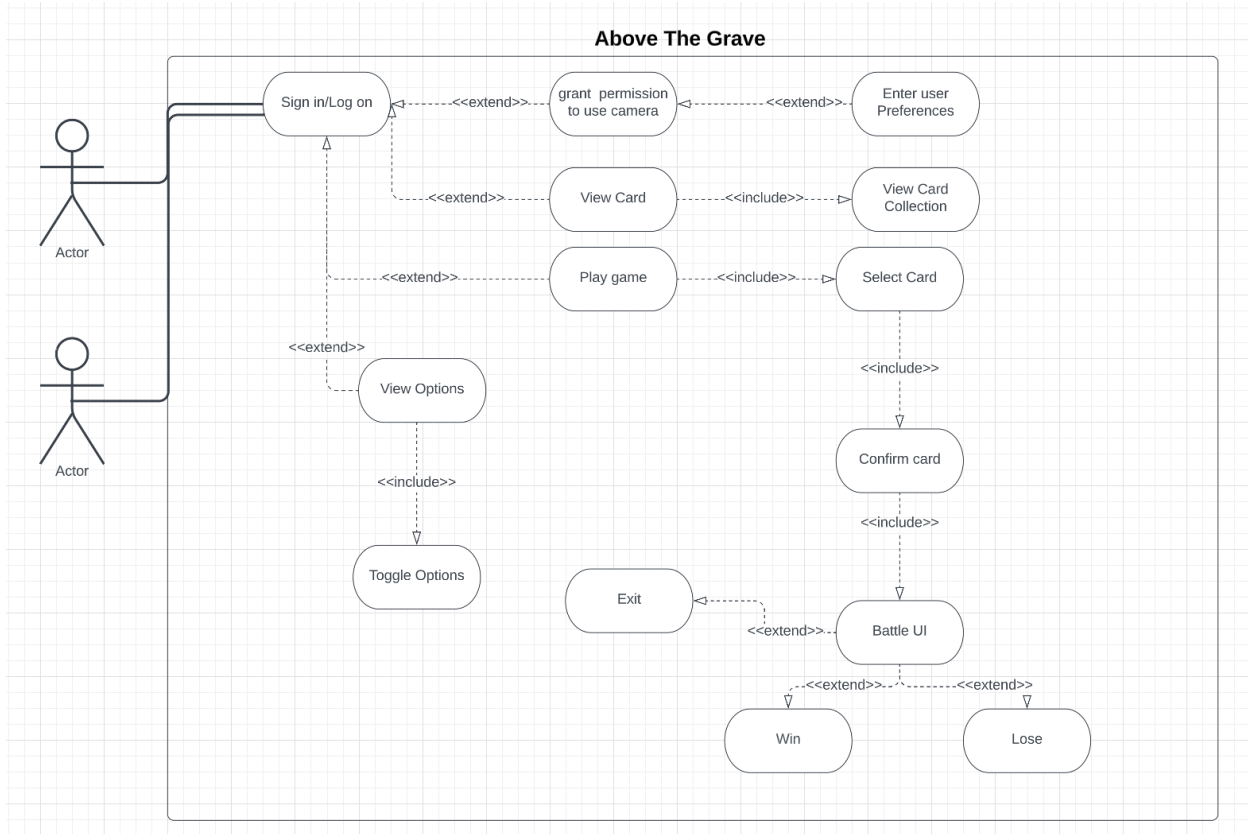
### 2.8.2. Budget Constraint

Title	Budget Constraint
Description	There is no budget for Above the Grave.
Priority	2

## 3. Requirements Modeling

### 3.1. User and System





## 4. Evolutionary Requirements

### 4.1. Functional Requirements

#### 4.1.1. Sound Effects within our Battle UI

Title	Sound effects on Battle UI
Description	When a user selects a card and plays the game, each move or attack will come with a certain sound effect.
Priority	2
Precondition(s)	The user needs to have a card already selected, and join the Battle UI. User then must make a move/attack.
Postconditions(s)	User makes another attack, or successfully defeats the opponent.
Use Case Diagram	4.1.1

## 4.2. Non-Functional Requirements

### 4.2.1. Battle UI Sound Effect Speed and Duration

Title	Sound Effect Speed and Duration.
Description	Sound Effect will occur almost immediately after the user makes a move/attack, and must end before the player makes their next move.
Priority	2
Applicable FR(s)	4.1.1