

## HW2 Report

<b>Project title</b>	HW1 - Fullerton Commencement		
<b>Semester:</b>	Spring 2022	<b>Report Due date:</b>	May 11, 2023
<b>Team name:</b>	Agile Developers		
<b>Member name:</b>	Daniel VanDenEykel	<b>Email:</b>	<a href="mailto:d.vandeneykel@csu.fullerton.edu">d.vandeneykel@csu.fullerton.edu</a>
<b>Member name:</b>	Kris Melgar Morales	<b>Email:</b>	<a href="mailto:cmelgarmorales@csu.fullerton.edu">cめlgarmorales@csu.fullerton.edu</a>
<b>Member name:</b>	Nicholas Jones	<b>Email:</b>	<a href="mailto:nicholasj898@csu.fullerton.edu">nicholasj898@csu.fullerton.edu</a>
<b>Member name:</b>	Mario Linares	<b>Email:</b>	<a href="mailto:mariolinares@csu.fullerton.edu">mariolinares@csu.fullerton.edu</a>
<b>Professor:</b>	Kirti Chaudhary	<b>Reviewer (optional)</b>	

## Revision History

Date	Revision	Description	Author
2/20/2023	1	Created document layout, added Table of Contents, added details to Introduction about Development case and iterations, added details about inception iteration and moved requirements to iteration of elaboration. First draft of Vision written.	Kris
2/20/2023	2	Adding remaining functional and non-functional requirements. Added Technology Preparation to Elaboration 1 based on Research	All Members in Team
2/26/2023	3	Added Use case, diagram and sequence diagram for 2 functional Requirements. Added AIFS to elaboration 2	Kris
2/26/2023	4	Added use-case diagram and sequence diagram for "Navigating the Event" functional requirement	Nick
2/26/2023	5	Added UseCase diagram for gift shop	Mario
2/26/2023	6	Added Sequence diagram for gift shop	Mario
3/1/2023	7	Add Use Case, Diagram, and Sequence diagram for "Graduation Movie Sequence" requirement	Daniel
3/8/2023	8	Revised and updated the UseCase and Sequence diagram for the gift shop	Mario
3/8/2023	9	Add a few lines to the vision and add short description on the Fullerton commencement	Mario
3/10/2023	10	Added Construction Phase 1 Material for Functional Requirements assigned to Kris. Includes Test cases and file structure for requirements	Kris
3/10/2023	11	Add construction phase 1 and 2 for the Gift shop	Mario
3/13/2023	12	Added Traceability Matrix, with details for Requirements ROBLOX-1 and ROBLOX-2. Modified Development Case. Added System and Activity/Interaction diagrams	Kris
3/13/2023	13	Revised own diagrams and added component basics for the diagrams	Nick
3/15/2023	14	Added construction Phase 1 material for functional requirements ROBLOX-4	Nick
3/16/2023	15	Added construction phase 2 documentation page on using ROBLOX-1 feature. Moved Traceability diagram from <a href="#">draw.io</a> to confluence. Added Transition phase documentation for building and publishing game. Appended Team Charter page from inception to end of document	Kris
3/16/2023	16	Table of Contents filled in, formatted for word.	Kris
5/9/2023	17	Updated Requirements(Roblox-6,roblox-7) Requirements, Analysis and Traceability(Use Case + Activity Diagram +Sequence Diagram), Implementation(Code Structure) and	Kris

Date	Revision	Description	Author
		Deployment and Testing(Test Cases, Tracability Matrix/Diagram) as part of construction 3	
5/9/2023	18	Added “Scrum instead of Agile-UP” Written Requirement to HW2 Report	Nick
5/9/2023	20	Updated Requirements(Roblox-7), Analysis(Use case, and sequence diagram), Implementation, Deployment, Testing(Test Cases), and updated traceability matrix/diagram	Mario
5/10/2023	21	Reviewed Report and Scrum Vs Agile UP and proofread and edited document	Daniel
5/10/2023	22	Reformatted Document and Confluence to include HW1 Report in HW2 Report. Modified Team Charter Evaluation goals to include HW2	Kris
5/11/2023	23	Added Updated system sequence diagram and activity diagram to Construction 3 for Analysis and Design	Kris

**Table of Contents**

Revision History	2
Table of Contents	4
Project Plan	5
Introduction	7
Inception	9
Technology Preparation	13
Elaboration 1	10
Elaboration 2	15
Construction 1	32
Construction 2	44
Construction 3	55
Scrum Vs Agile UP	78
Transition	80
Team Charter	88

**Project Plan**

What	Who	Expected Date Done	Estimation	Resources	Status
Inception Meeting	Everyone	2/6	1-2hrs	Discord	Complete
Initial Project Plan Created	Kris	2/13	1-2hrs	Confluence	Complete
5 Functional and 2 Nonfunctional Requirements	Everyone	2/13	1 hr	Confluence	Complete
Risk List	Kris	2/13	20-30 minutes	Confluence	Complete
Team Charter	Everyone	2/13	20-30 minutes	Confluence	Complete
Elaboration 1 Meeting	Everyone	2/13	1-2hrs	Discord	Complete
Use cases for Requirements	Everyone	2/20	20-30 minutes	Confluence	Complete
Elaboration 2 Meeting	Everyone	2/20	1-2hrs	Discord	Complete
Diagrams for Use cases	Everyone	2/27	1-2 hrs	Confluence, Draw.io	Complete
Construction 1 Meeting	Everyone	2/27	1-2hrs	Discord	Complete
Implementation for Requirements	Everyone	3/6	3-4 hrs	Bitbucket, Roblox Studio	Complete
Test Cases for Requirements	Everyone	3/6	1-2hrs	Confluence, Roblox Studio	Complete
Construct 2 Meeting	Everyone	3/6	1-2hrs	Discord	Complete
Documentation for Requirements as needed	Everyone	3/13	1-2hrs	Confluence	Complete
High Level Class Diagram	Mario	3/13	1-2hrs	Confluence, Drw.io	Complete
System and Interaction Diagrams	Kris	3/13	1-2hrs	Confluence, Draw.io	Complete
Traceability Matrix	Everyone	3/13	20-30 minutes	Confluence	Complete
Traceability Matrix	Kris	3/13	20-30 minutes	Confluence, Draw.io	Complete
Transition Meeting	Everyone	3/13	1-2hrs	Discord	Complete
Final Build Published to Github	Kris	3/16	30 minutes	Github, Bitbucket	Complete

What	Who	Expected Date Done	Estimation	Resources	Status
Documentation made for Building and Publishing roblox Game	Kris	3/16	1-2hrs	Github, Confluence, Bitbucket, Roblox Studio	Complete
Final Report PDF	Kris	3/16	30 minutes	Confluence, Google Docs, Adobe Acrobat	Complete
Construction 3 Meeting	Everyone	4/3	30 minutes	Discord	Complete
Additional Requirements	Kris,Mario	5/11	1-2hrs	Confluence	Complete
Additional Analysis and Design	Kris,Mario	5/11	2-3hrs	Confluence, Drawio	Complete
Additional Implementation	Kris,Mario	5/11	4-8hrs	Confluence, Bitbucket, Roblox Studio	Complete
Additional Deployment and Testing	Kris,Mario	5/11	2-3hrs	Confluence, Bitbucket, Roblox Studio	Complete
HW2 Part B Report	Daniel,Nicholas	5/11	3-4hrs	Confluence, Class Book	Complete
Final Build Published to Github	Kris	5/11	30 minutes	Github, Bitbucket	Complete
Final Report PDF	Kris	5/11	30 minutes	Confluence, Google Docs, Adobe Acrobat	Complete

## Introduction

### **Business Model:**

Fullerton Commencement is a Roblox game that allows students at California State University, Fullerton to celebrate, with friends and family, a graduation commencement virtually. This game was developed using Roblox Studio and following the Unified Process (UP) which gave a structure that allowed for agile principles to be used in the planning, designing, and development of our game.

### **Vision**

A wonderful commencement event that strives to match the energy and spirit of graduation at CSUF. It is our goal to allow as many people as possible experience their loved ones and friends walk up on that stage and start the next chapter in their life. The personality of our student bodies will be on display as they celebrate their achievement over their various year of studying. No matter how far student are from home or campus, they will be able to experience their slice of CSUF in a festive online form. Roblox will truly bring our student body together in times where on campus commencement is unavailable. Our Roblox application will allow for students, parents, and faculty to experience the CSUF commencement ceremony in a manner that keeps everyone safe and reflects the traditional in person commencement ceremony. We believe that students should be able to celebrate such an important milestone in their lives despite the current health restrictions and our Roblox commencement event will make sure that celebration can continue to take place.

### **Development Case**

Disciplines	Work Products	Notes
Business Model	Vision	5-10 lines of text, should encapsulate what is needed by CSUF Faculty to host event
Business Model	Development Case	This table. Contains all expected work products for project
Requirements	Functional Requirements	Focus on features that user can interact with
Requirements	Non-Functional requirements	Focus on scalability, security, etc
Analysis and Design	Use Cases and Sequence Diagrams	Detailed Flow of Requirements
Analysis and Design	Architecturally Influential Factors	Factors like scalability performance and security and ways to improve features with respect to them.
Analysis and Design	Architectural Design Documents	Research more into how to write design document
Analysis and Design	High Level Design of Software Architecture	Class Diagrams that encapsulates structure of project.
Analysis and Design	Technology Preparation	Necessary research to program in Roblox, with Lua and Rojo
Implementation	Roblox Game	Github Repository, or uploaded game world of Roblox Map on Studio
Project Management	Iteration Plan	Plan for work products to be started throughout the iterations
Project Management	Risk List	Expected risk for project from developer perspective to user experience.

**Project Management:****Iteration Plan**

Discipline	Techniques	Artifact	Inception	Elaboration 1	Elaboration 2	Construction 1	Construction 2	Construction 3	Transition
Business modeling	Pretend to be Dean of University, and think what will be needed for commencement.	Vision, Development Case	Start	Refine	Refine	Refine	Refine	Refine	Refine
Requirements	Thinking about user actions and requests.	Functional and Non Functional Requirements	Start	Refine	Refine	Refine	Refine	Refine	Refine
Analysis and Design	Unify format for diagrams.	Use cases and sequence diagrams. Technology Preparation		Start	Refine	Refine	Refine	Refine	Refine
Implementation	Separate task into issues. Features separated into different branches	AIFS Risk Factors Class and Activity Diagrams Feature Implementation			Start	Refine	Refine	Refine	Refine
Deployment and Test	Branch and Merge strategies.	Traceability Matrix Test Cases Documentation				Start	Refine	Refine	

## Inception

### **Business Model:**

For Fullerton Commencement, we need to clearly state our goals and necessary work products to follow the methods of the unified process. Our overall goal is to have an event where commencement users can participate in a online commencement event similar to the commencement event on campus at CSUF.

### **Objectives**

An online Roblox commencement environment that mimics the CSUF commencement event.

An event that cannot be ruined by bad actors

An event that is fun for the graduates and visitors

A well documented software package that can be understood and setup by faculty and staff at CSUF.

A conflict free development environment between commits and merges

A roblox event that can run smoothly on at least one platform.

### **Requirements:**

Our main requirement for this event is for commencement participants will have the ability to see their characters walk onto the graduation stage and receive a diploma from an official/npcc. It is not critical to mimic the CSUF campus, but effort should still be put into having the graduation area be similar to actual event on campus. Users should also be unable to join the event if they were not allowed.

### **Project Management:**

#### **Risk List**

Risk	Severity	Impact	Response
Unauthorized user join the Commencement event and begin to grief	Mid	Event is disturbed and student and faculty experience of the event is negative	Create scripts that disable users from joining based on a predefined whitelist
Game Server cannot load / crashes during execution	High	Event is disturbed and student and faculty experience of the event is negative	Ensure game is able to execute between implementation of different features
Developers Fall Behind on Documentation and Diagrams	Low	More time will be needed between iterations to fill in the gaps from proceeding to fast.	Work on documentation before proceeding forward in iteration

## Elaboration

### **Iteration 1**

#### **Requirements:**

Requirement	User Story	Importance	Diagram ID	Taken By
Unauthorized users are not able to enter the commencement event.	Alex has shared a link/invite with his friend to the commencement event. Alex is not authorized to have a link to the event. When Alex attempt to join the event, he is blocked from entering the server.	HIGH	ID-ROBLOX-1	 <b>KM</b> <a href="#">Kris</a> <a href="#">Melgar</a> <a href="#">Morales</a>
Allow attendees to purchase gifts for graduates	Bob is attending the commencement as a guest to watch his friend David receive his diploma. At the commencement he will be able to buy flowers to give to his friend after he has received his diploma.	MED	ID-ROBLOX-3	 <b>ML</b> <a href="#">Mario</a> <a href="#">Linares</a>
A scripted movie will play when the commencement begins and graduate will be able to walk up to the stage and receive a diploma	Once all of the players are ready, the host will begin the commencement by playing a scripted movie. Rick is a graduate and he will walk up to the stage and receive his diploma when it is his turn.	HIGH	ID-ROBLOX-5	 <b>DV</b> <a href="#">Daniel</a> <a href="#">VanDenEykel</a>
Users can explore event and interact with other people before and after the commencement event	Jim is an attendee at the commencement event, viewing David graduate. While waiting for the event to begin, Jim can walk around the buildings surrounding the campus and interact with other students and visitors.	MED-HIGH	ID-ROBLOX-4	 <b>NJ</b> <a href="#">Nicholas</a> <a href="#">Jones</a>
Tuffy NPC exists and gives Tuffy related merchandise to user.	Alan walks up to Tuffy in the Roblox world, and interacts with it. Alan then see text from turfy and receives a free gift.	LOW		<b>X</b> <b>X</b>
Graduation area and stage are represented in the Roblox world for users to sit in to wait to and get their diplomas.	Alan enter the graduation event area, and sees the seating area for graduates, accompanied by a stage.	MID-HIGH	ID-ROBLOX-2	 <b>KM</b> <a href="#">Kris</a> <a href="#">Melgar</a> <a href="#">Morales</a>
When users get their diploma, they can choose a preset choice of emotes to do.	Cindy is about to go up to receive her diploma. She shes a window on her screen that show different emotes that she can prepare when she goes up on stage. She clicks an emote, later goes on stage to receive	LOW-MID		<b>X</b> <b>X</b>

Requirement	User Story	Importance	Diagram ID	Taken By
	her diploma, and successfully preforms the selected emote.			
Graduates are able to select cap and gown to wear before the graduation event.	When Ricardo enters Fullerton Commencement from the Roblox Launcher, he is greeted to a screen with colors to choose for their cap and gown. When he enters the game world, he sees that his roblox avatar is wearing the selected outfit.	LOW-MID	X	X
User are automatically recognized as a graduate or a visitor to the event.	Daniel is a graduate. He enters the graduation event. Daniels sees the screen to choose his cap and gown colors when they join.  Ariel enters the graduation and is a visitor. She does not see the cap and gown selection screen and is moved straight to the lobby.	MID	X	X
Visitors are visible in the visiting seating when the graduation event is in progress.	Daniel is a graduate, and his roblox character is able to see the people he invited as visitors are in the visitor seating as the scripted commencement events plays out	MID	X	X

Non-Functional Requirement	User Story	Importance	To Implement
The commencement scripts will feature a user-friendly interface for the Host to control all aspects of the event.	Julia is hosting a commencement event and needs to make some modifications. The interface will provide any functionality to allow Julia to change aspects of the event.	LOW	X
The event must complete on all users computers without random disconnects and crashes	John joins the event for the commencement 20 minutes before it starts. He remains in the game world all the way until the event is over. John did not experience any issues with his connection to the game world and did not crash	HIGH	✓
The commencement event can support up to 100 players concurrently		MID	X
General layout of roblox maps matches the CSUF campus	Irene is in the roblox world for the CSUF commencement event.		X
All assets and objects used in the Roblox world must be shader compatible	Derrick is in the game world and notices that the objects in the world all have shaders. There is not a single object that fails to be properly shaded, and it looks pleasing to their eye.	HIGH	✓

## Technology Preparation

While some people on our team may have played a Roblox game before, we were inexperienced as developers using Roblox Studio. As a result, we would need to learn what languages and documentation were required to be able to produce a Roblox game.

In Roblox Studio, the game is defined as the interaction between different objects and models, such as the actual game environment, the players themselves, the lighting engine, material service, the server side scripts, the starting equipment and UI, the team system, the sound service, chat engine and localization service. These systems have objects and scripts underneath, and when they are loaded into Roblox when a game is initialized, the scripts that belong to the loaded objects are executed. Thus anything we add to our game will exist at the top level of file system as objects with scripts as its children.

Originally our Roblox game was shared among our developer using the tools provided by Roblox themselves. While this may have been helpful for beginning development, we decided to see if there was a way to integrate source control so that we could not only develop our environments in Roblox Studio, but also track the commits in a cloud based git software like Bitbucket, which worked natively with our document repository tool Confluence. This necessity led us to the software called Rojo, located at <https://rojo.space/>, which allows us to use our personal IDEs to code and maintain version control separate from Roblox Studio.

Our source repository now consists of files that could be taken by the Rojo software, to generate a rbxlx file, which is the file that contains all the information about a game in Roblox studio, and can be imported into it for testing and development. This was done by taking our first workspace on Roblox Studio, and following the documentation given by rojo when converting a Roblox Studio project into its Rojo equivalent.

Rojo has a feature that allows the source control repository to be synched and live tested with Roblox Studio. While we did attempt to run the synchronization software, it did not return the intended results, leaving us to find a way to synchronize the object system of Roblox Studio with the file system of Rojo. For example, if we were to want to move an object called EventModel to the Rojo repository, we would have to go into roblox studio and right click the top level of the object, and save it as rbmx file, which contains the binary information of the object. Then it would be necessary to edit default.project.json in the rojo repository, as it controls the object system in Rojo.

Between our developers, it was our responsibility to create issues in our workflow management software, Jira, create new branches for said issues, and then finish the work on them. Developers would also need to rebase from the master branch and resolve any conflicts. Eventually developers need to pull and merge their issue branches into the main branch when it is complete. Following this workflow for our source control will reduce the amount of conflicts between different commits by the team.

To ensure our main risk of unintended user joining the commencement was able to be solved with Roblox Studio, we searched the documentation for any event listeners that would initialize when users would attempt to join. In the Roblox Studio Documentation

When Roblox attempts to load an object into a world, the respective scripts are called and ran. In Roblox Studio, these scripts are written with Lua, located at <https://www.lua.org/>, which is a lightweight scripting language that supports various programming constructs like object-oriented programming, functional program, etc. Since the concepts of other languages are integrated into lua, learning lua for our developers will not be a problem. This issue can also be addressed by attempting to use AI models like ChatGPT to learn how to program certain concepts in lua. This has been tested with this snippet generated by ChatGPT

```
if stringValue then
```

```
-- Access the value of the StringValue
local substrings = string.split(stringValue.Value, "\n")
-- Convert each substring to a number and add it to an array
```

```
for i, substring in ipairs(substrings) do
    txt[i] = tonumber(substring)
end

-- Print the array of numbers
print(table.concat(txt, ", "))

else
    print("Could not find the StringValue instance!")
end
```

when asking how to tokenize a txt file converted to a string, and then compare said string to another value. It was not necessary to know the methods that exist in the string or table classes, or how loops work. This, however, is just a supplement to our programming, and we will be necessary to look at documentation for Lua while we implement our functional requirements.

Overall our team will continue to work with Rojo and Roblox Studio, to make our commencement environment, while continuing to scan the documentation for both Lua and Roblox Studio

**Iteration 2****Analysis and Design:****Use Case and Sequence Diagrams**

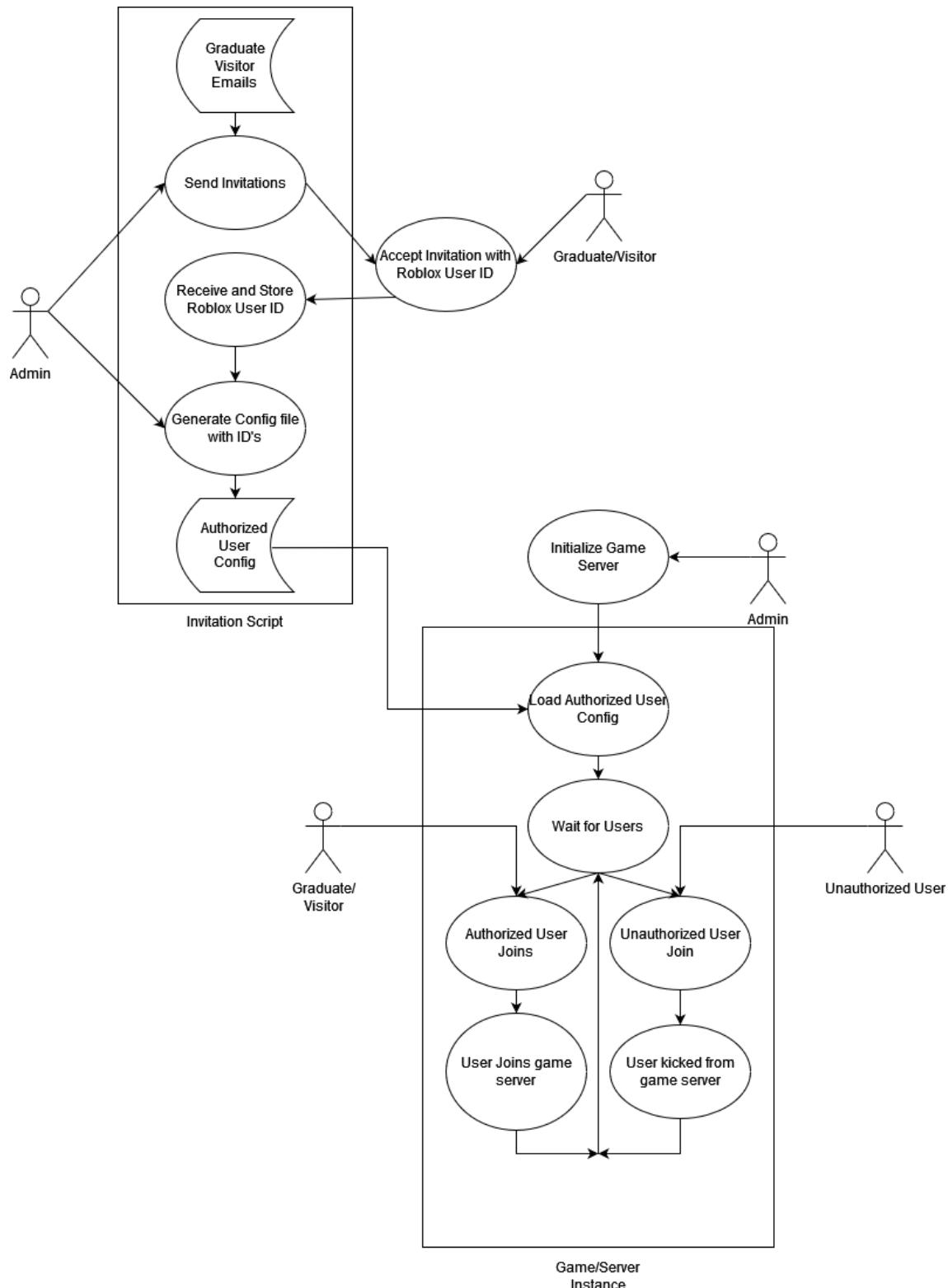
Version	Date Created	Originator	Changes Made
0.1	2/26/2023	Kris Melgar Morales	Created use case for Unauthorized access, with diagrams

Requirement	User Story	Importance	Taken By
Unauthorized users are not able to enter the commencement event.	Alex has shared a link/invite with his friend to the commencement event. Alex is not authorized to have a link to the event. When Alex attempt to join the event, he is blocked from entering the server.	HIGH	 <u>Kris</u> <u>Melgar</u> <u>Morales</u>

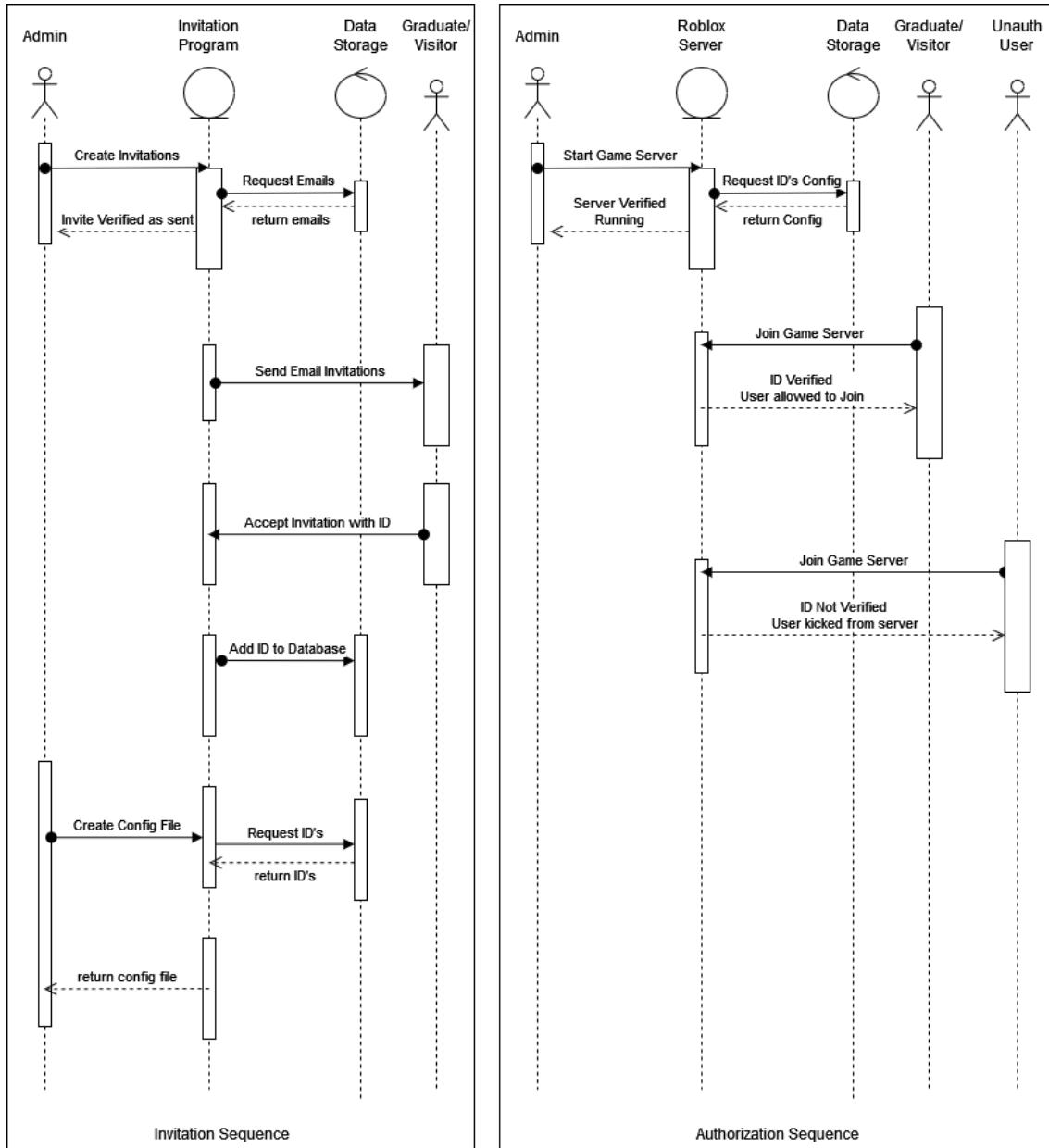
**Component basics**

Diagram ID	ID-Roblox-1
Description	For the Roblox commencement event, the only allow guest to the server should be those that have been given an invitation, for either being a graduate, or a visitor celebrating a graduates commencement. Unauthorized user will not be allowed entry into the event whatsoever.
Primary Roles	Graduate, Visitor, Unauthorized User(Troll)
Pre-Conditions	User is logged into Roblox. User has been authorized to join the roblox event.
Trigger	User joins the server.
Basic Flow	1. User logs into Roblox. 2. User launches the event for the Roblox Commencement. 3. User has been authorized to join, so they are able to proceed into event after loading the game. 4. User is able to experience event as a Graduate or a visitor.
Alternate Flow	1. Unauthorized user logs into Roblox. 2. User launches the event for the Roblox Commencement. 3. User has not been authorized to join, so they are kicked out and/or unable to connect to the server. 4. Unauthorized user remains unable to join no matter what.
Post-Conditions	Users who are verified for the event are able to join, unauthorized users remain unable to join the event.

## Use Case Diagram



# Sequence Diagram



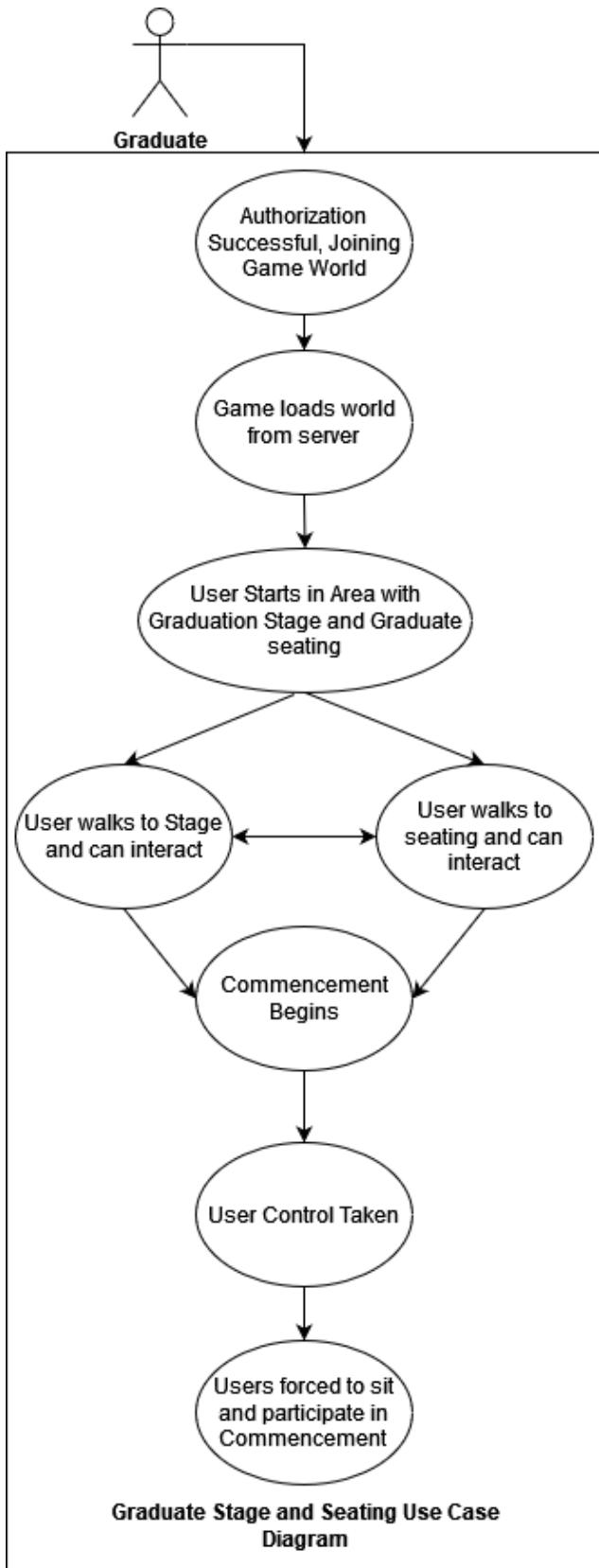
Version	Date Created	Originator	Changes Made
1.0	2/26/2023	Kris Melgar Morales	Created use case, use case diagram and sequence diagram

Requirement	User Story	Importance	Taken By
Graduation area and stage are represented in the Roblox world for users to sit in to wait to and get their diplomas.	Alan enter the graduation event area, and sees the seating area for graduates, accompanied by a stage.	MID-HIGH	 Kris Melgar Morales

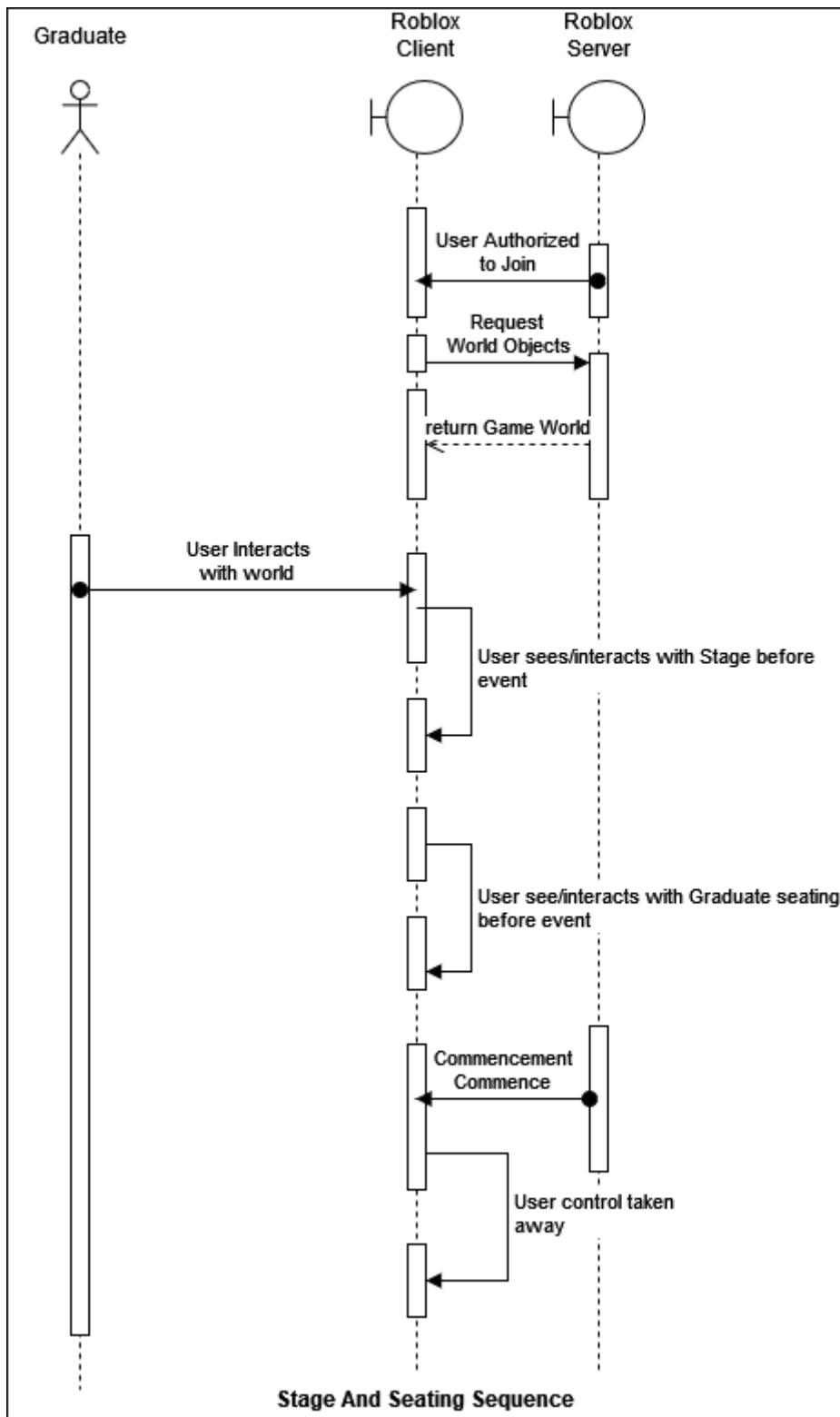
## Component basics

Diagram ID	ID-Roblox-2
Description	Users participating in the event can view a representation of a stage and seating area, as one would experience during a commencement event.
Primary Roles	Graduates
Pre-Conditions	Graduates are logged in and verified to the game event
Trigger	Upon game world entry
Basic Flow	<ol style="list-style-type: none"> <li>Graduate loads the game world and enters the game instance.</li> <li>Graduate is able to see a seating area with chairs for all possible graduate.</li> <li>Graduate is able to see a stage that will be used when diplomas are handed out.</li> </ol>
Alternate Flow	N/A
Post-Conditions	N/A

## Use Case Diagram



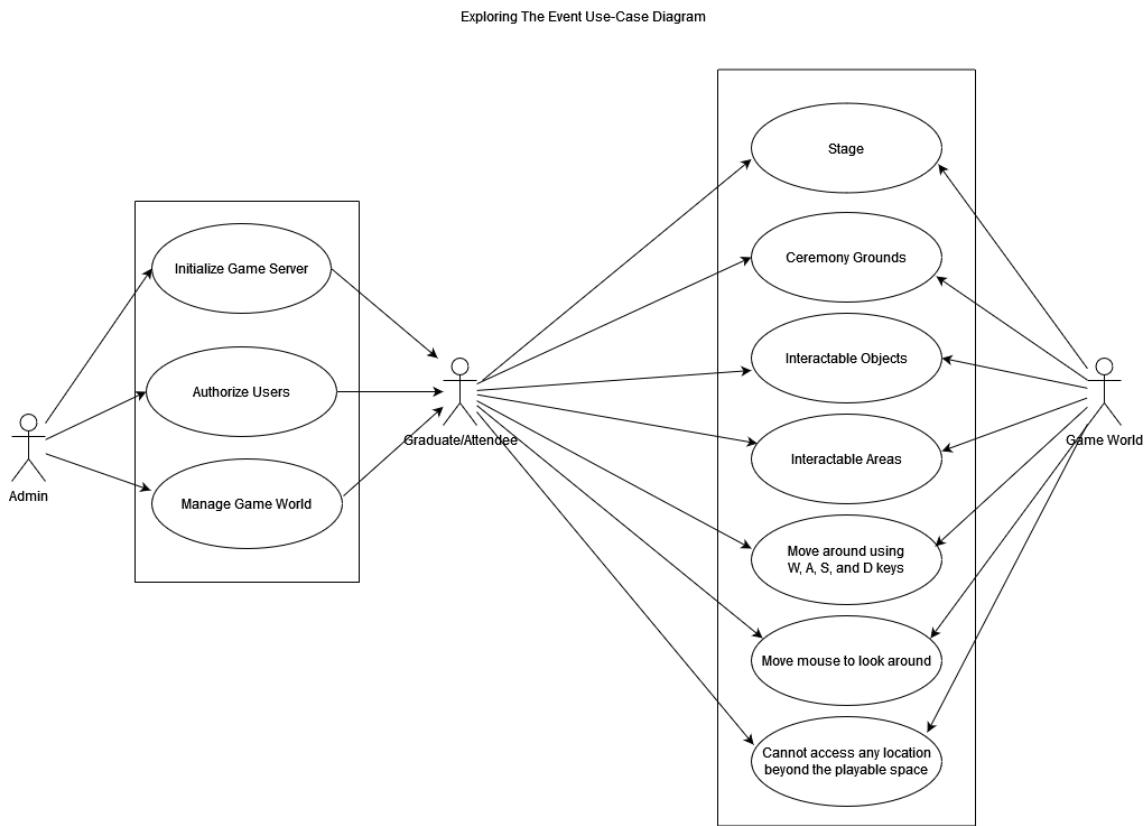
## Sequence Diagram



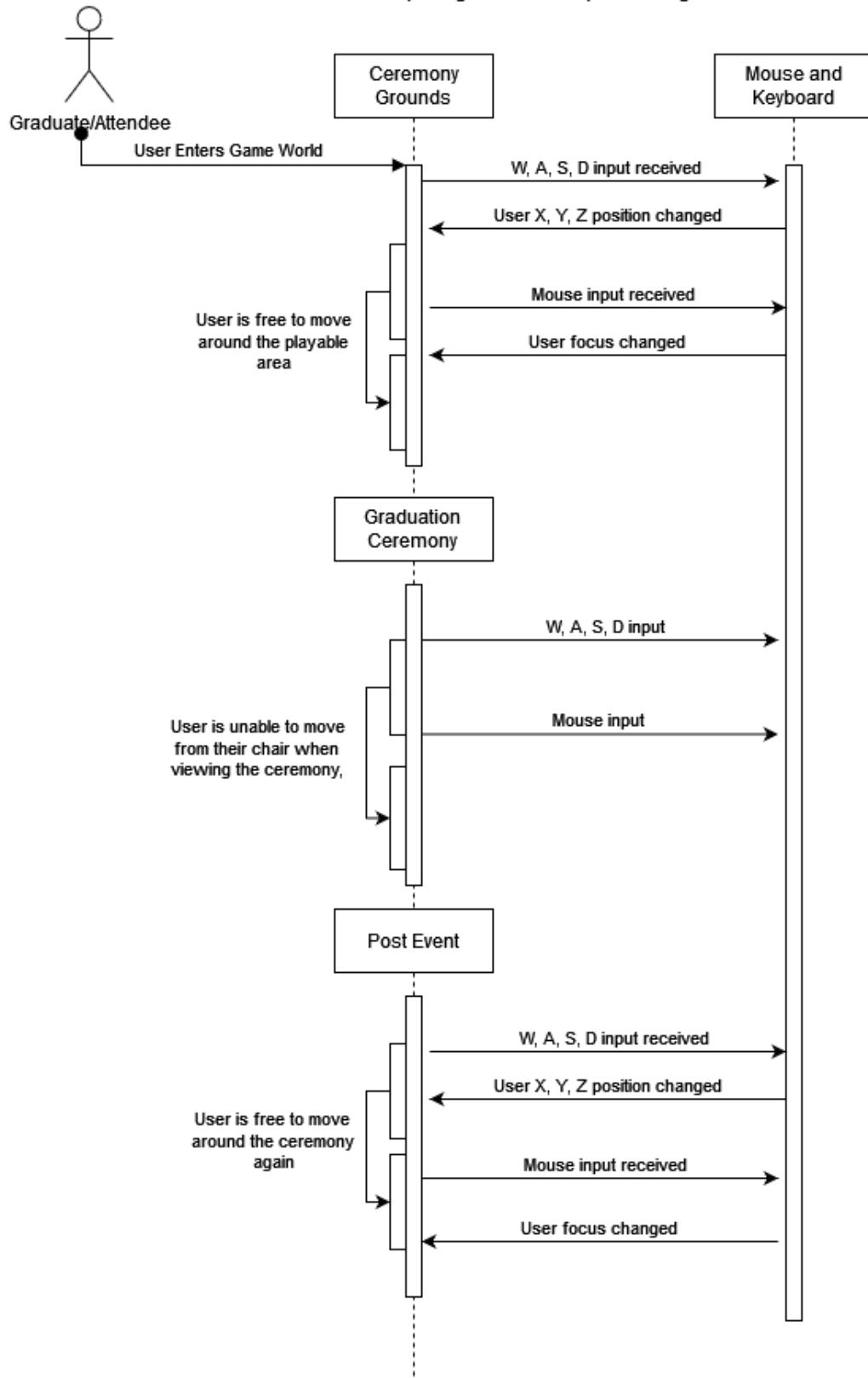
Version	Date Created	Originator	Changes Made
1.0	2/26/2023	Nicholas Jones	Created use case and sequence diagram for Event Exploration

Requirement	User Story	Importance	Taken By
Users can explore event and interact with other people before and after the commencement event	Jim is an attendee at the commencement event, viewing David graduate. While waiting for the event to begin, Jim can walk around the buildings surrounding the campus and interact with other students and visitors.	MED	 <a href="#">Nicholas Jones</a>

Diagram ID	ID-Roblox-4
Description	Users participating in the event can interact with the stage and seating area, meet other graduates or attendees, and access the shop before the event starts.
Primary Roles	Graduates, Attendees
Pre-Conditions	Game world must be initialized Graduates and attendees are authorized to participate in the event
Trigger	Upon game world entry Upon event ceremony
Basic Flow	<ol style="list-style-type: none"> <li>Graduate or attendee loads into the game world and enters the game instance.</li> <li>Graduate or attendee can move using W, A, S, D and look around using the mouse.</li> <li>Graduate or attendee is locked to their chair to view the event.</li> <li>Graduate or attendee can return to interacting with the game world using W, A, S, D and their mouse once the event is over.</li> </ol>
Alternate Flow	N/A
Post-Conditions	N/A



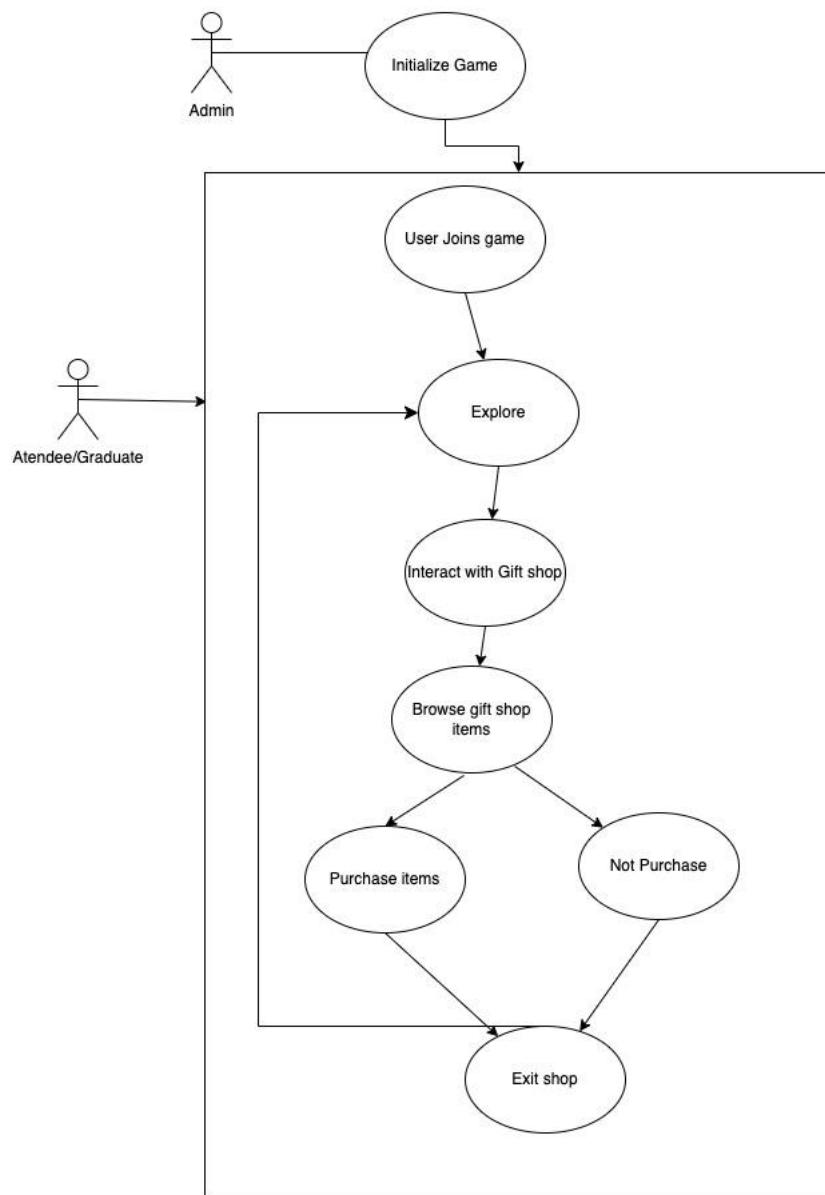
## Exploring the Event Sequence Diagram



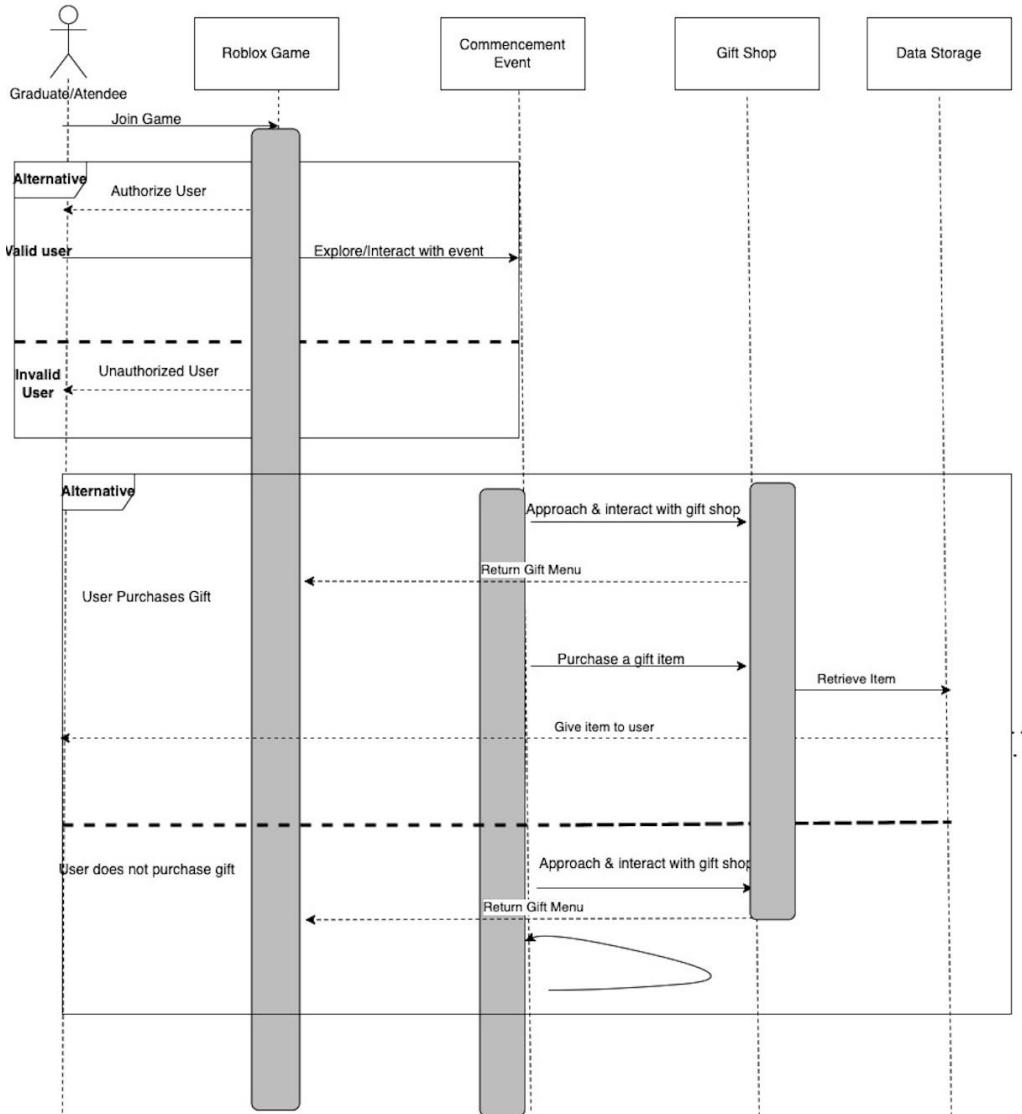
Version	Date Created	Originator	Changes Made
0.1	2/27/2023	Mario Linares	Created use case for Gift shop

Requirement	User Story	Importance	Taken By
Allow attendees to purchase gifts for graduates	Bob is attending the commencement as a guest to watch his friend David receive his diploma. At the commencement he will be able to buy flowers to give to his friend after he has received his diploma.	MED	 <a href="#">Mario Linares</a>

Diagram ID	ID-Roblox-3
Description	For the Roblox commencement event, graduates and guests should be able to purchase gifts from a gift shop for the graduates.
Primary Roles	Graduate and Visitor
Pre-Conditions	User is authorized to be present at the event.
Trigger	User walks up to the shop and engages with it.
Basic Flow	<ol style="list-style-type: none"> <li>1. User logs into Roblox and is authorized to be present at the commencement</li> <li>2. User walks up to a shop and can engage with it.</li> <li>3. User can view the gift items</li> <li>4. User purchases an item</li> <li>5. Exits the shop</li> </ol>
Alternate Flow	<ol style="list-style-type: none"> <li>1. User logs into Roblox and is authorized to be present at the commencement</li> <li>2. User walks up to a shop and can engage with it.</li> <li>3. User can view the gift items</li> <li>4. Exits the shop without making any purchase</li> </ol>
Post-Conditions	N/A



**Figure 1 Use Case Diagram: Gift Shop**

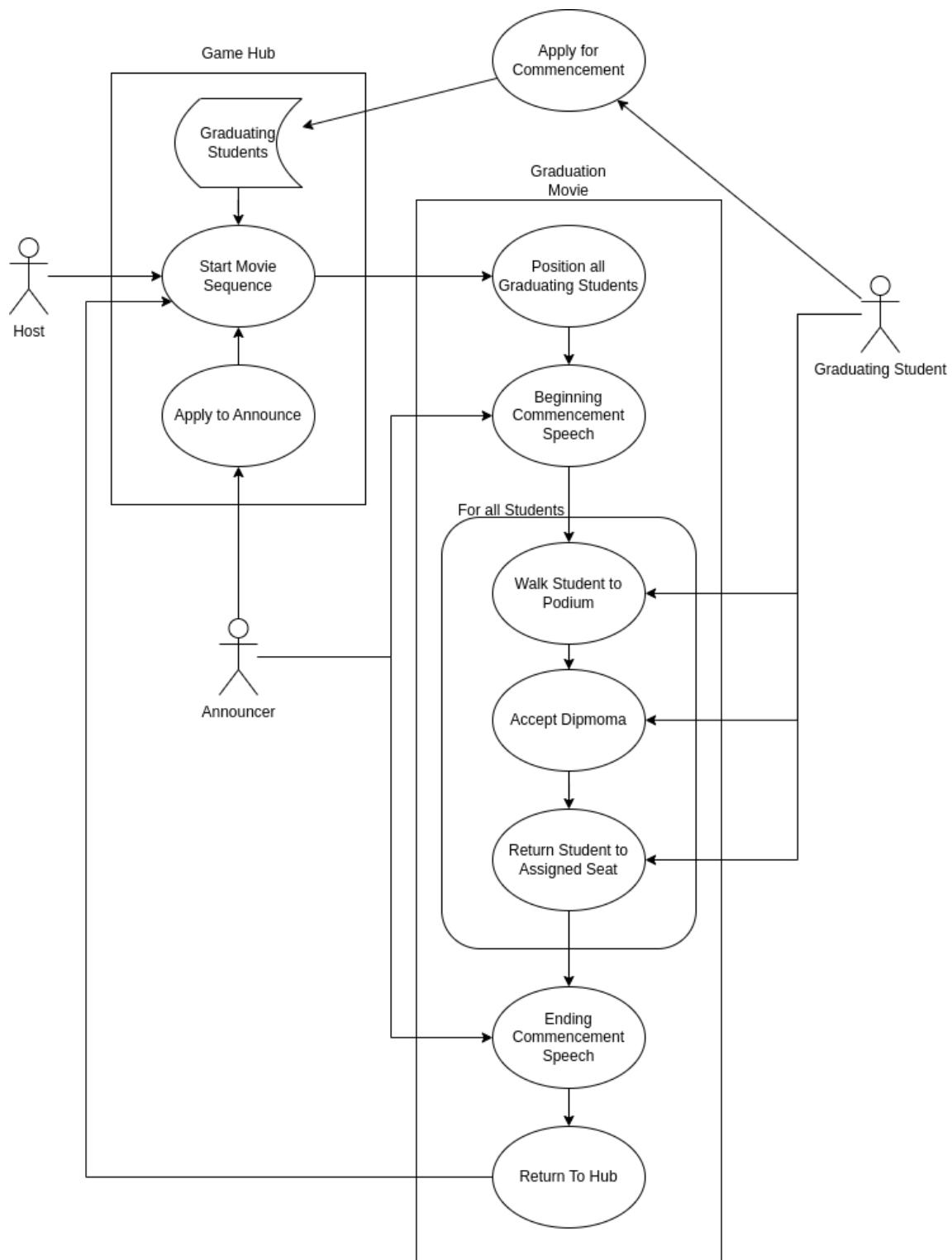
**Figure 2 Sequence Diagram: Gift shop**

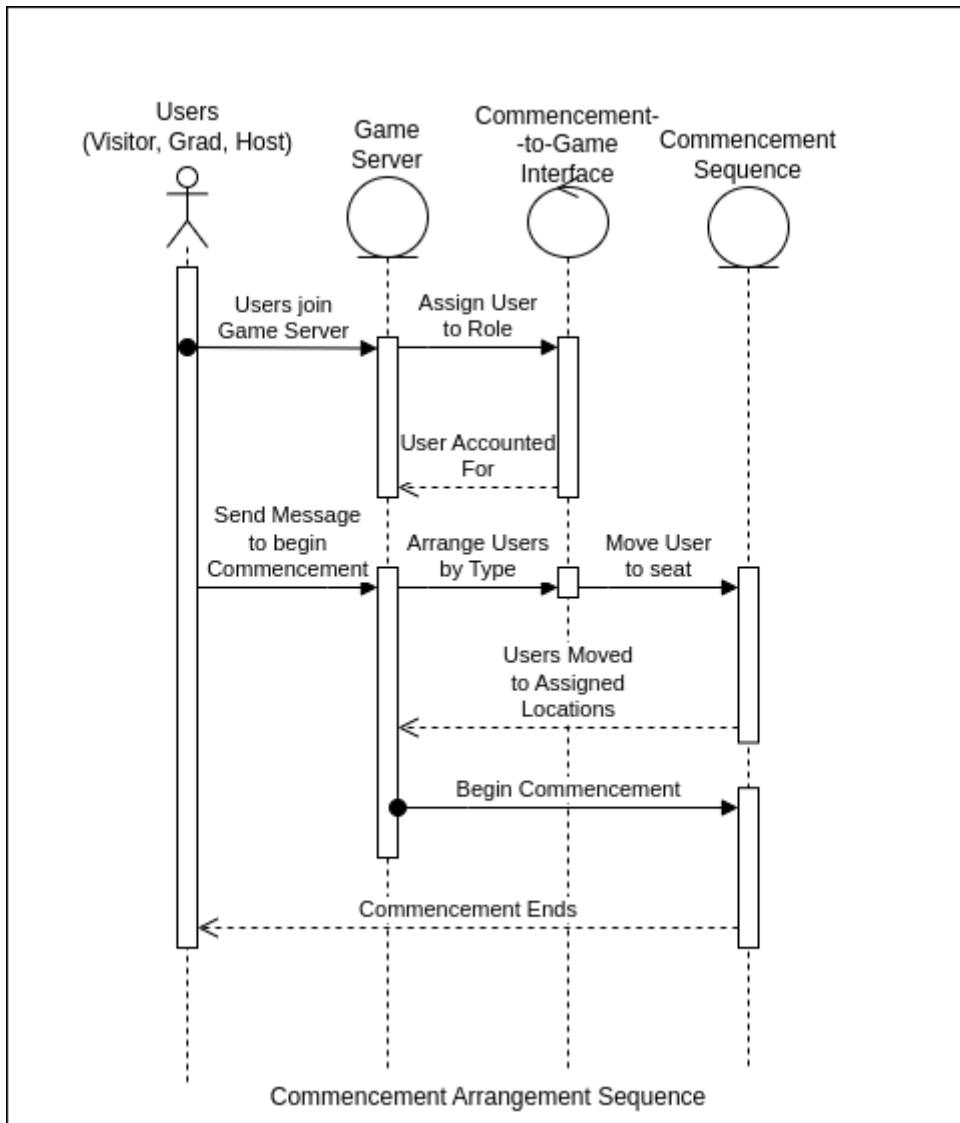
Version	Date Created	Originator	Changes Made
0.1	2/28/23	Daniel VanDenEykel	

Requirement	User Story	Importance	Taken By
A scripted movie will play when the commencement begins and graduate will be able to walk up to the stage and receive a diploma	Once all of the players are ready, the host will begin the commencement by playing a scripted movie. Rick is a graduate and he will walk up to the stage and receive his diploma when it is his turn.	HIGH	 <a href="#">Daniel VanDenEykel</a>

## Component Basics

Diagram ID	ID-Roblox-5
Description	When the host starts the commencement, it will place the graduates in the seated positions in front of the podium and the visitors on the bleachers along the sides of the stadium. As the commencement goes, each graduate will automatically stand up and walk to the podium to receive their diploma, perform an action or speech message, and then return to their seat. The commencement sequence ends once all the graduates obtain their diploma.
Primary Roles	Host, Graduates, Visitors
Pre-Conditions	The attendees and host are all in the game and ready to begin the commencement.
Trigger	The host starts the commencement movie.
Basic Flow	<ol style="list-style-type: none"> <li>1. The host starts the movie.</li> <li>2. Every member in the game is placed in their respective positions at the stadium.</li> <li>3. Each graduate goes through their part of the ceremony: walking up, taking the diploma, and walking back.</li> <li>4. The ceremony automatically ends once all graduates have been awarded.</li> </ol>
Post-Conditions	All graduates who have completed the ceremony will now have a diploma item and won't need to repeat their part of the ceremony.





## Architecturally Influential Factors

### **Performance**

The commencement event is celebrated on Roblox, with access through PCs, mobile devices and game consoles. As such the expected hardware that Roblox will run on is unknown, and will lead to different performance issues across architectures

### **Critical Features**

**Frame Lag reduced:** When the event occurs, users should not notice frames being skipped due to slow rendering performance on a device.

**Server Response time:** For certain scripts, information about a user will need to be updated both client and server side. Any change a user would make on a client should be reflected as soon as possible on server, and then those changes should propagate as fast as possible.

### **Architecture Design:**

**Asynchronous Programming:** Use asynchronous programming techniques to minimize the response time between the client and the server. This will ensure that the changes made by a user on a client are reflected as soon as possible on the server, and then those changes should propagate as fast as possible.

**Minimized Scripts:** Ensure the amount of scripts and models loaded at a time is reduced to the smallest subset that produce the expected events needed for the event. Less scripts to load mean faster rendering of each frame.

### **Scalability**

**Description:** The commencement event is celebrated by all the graduates at CSUF including the visitors. As such, there will be a large amount of people attempting to enter the event at once

### **Critical Features:**

**User Authorization:** Any user attempting to join the event will force the server to search the necessary credentials needed to verify a Roblox account. As the scale of the people invited increases, the load times of the user account data will increase for each sign in linearly for each extra person tracked in the system.

**Seating for Graduates:** When a scripted event starts for all the users in the Roblox space, they will need to be assigned seating.

**Architecture Design:** To address these scalability concerns, we propose the following architecture design decisions:

**100 Player Support:** Ensure all systems like the scripted ceremony, and event space is able to support the max player count defined by Roblox at 100. Features must keep in mind that the event should be at the max player count, based on the amount of graduates that exist at CSUF.

### **Security**

**Description:** The commencement event is expected to be distributed across a large number of graduates and invitees including children, and the security of the platform is of paramount importance. The event will involve social event between the graduates and visitors of CSUF.

### **Critical Features:**

**User Authentication:** The system must deny users who identification does not match the reservation.

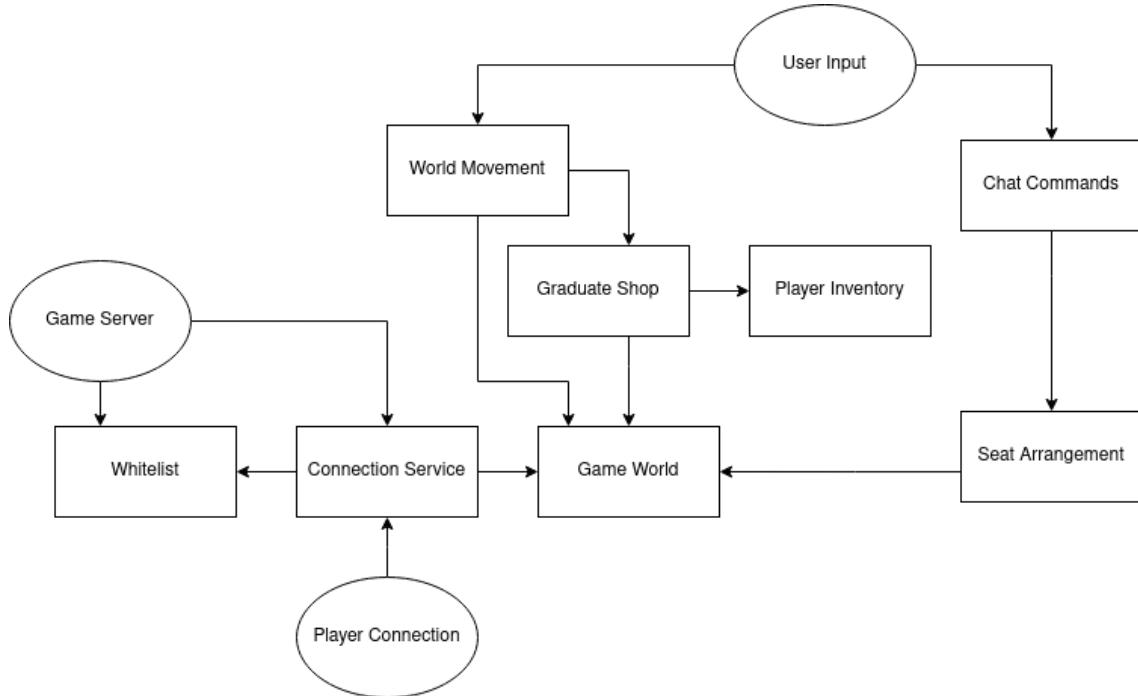
**Architecture Design:** To address these security concerns, we propose the following architecture design decisions:

**Whitelist Mechanism:** Implement a whitelist system, where users whose Roblox ID is not included in a whitelist file are denied entry into server upon intimidating a connection

**Authentication Mechanism:** Implement a user account system based on limited invite keys. Users can sign up with a key, and ensures only a limited amount of invites can be attributed to a limited number of user accounts.

These architecture design decisions will ensure that the Roblox event platform is secure and provides a safe environment for all participants.

### Architectural Design



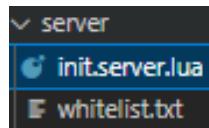
## Construction

### **Iteration 1**

#### **Implementation:**

##### **ROBLOX 1: Unauthorized Users**

Below is the details for the feature programmed in Lua and the File structure in our GitHub repository



**Figure 3 Folder Directory**

```
src > server > whitelist.txt
1 1643749915
2 1643749915
3 1643749915
4 1643749915
```

**Figure 4 whitelist.txt**

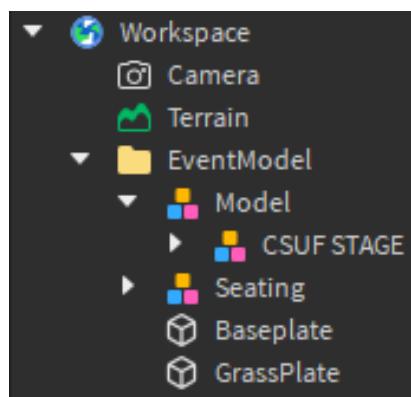
```
initserver.lua 2 ×
src > server > initserver.lua > ...
1 print("Hello world, from server!")
2 -- Get a reference to the server script
3 local serverScript = game:GetService("ServerScriptService"):WaitForChild("Server")
4
5 -- Get a reference to the StringValue instance
6 local stringValue = serverScript:FindFirstChild("whitelist")
7
8 -- Check if the StringValue was found
9 local whitelistUserIDs = {}
10 if stringValue then
11     -- Access the value of the StringValue
12     local substrings = string.split(stringValue.Value, "\n")
13     -- Convert each substring to a number and add it to an array
14     for i, substring in ipairs(substrings) do
15         whitelistUserIDs[i] = tonumber(substring)
16     end
17     -- Print the array of numbers
18     print(table.concat(whitelistUserIDs, ", "))
19 else
20     print("Could not find the StringValue instance!")
21 end
22
23 local Players = game:GetService("Players")
24
25 Players.PlayerAdded:Connect(function(player)
26     if not table.find(whitelistUserIDs, player.UserId) then --If the UserID value is not in the table this returns nil. In Lua nil equals false.
27         player:Kick("You are not whitelisted on this server.")
28     end
29 end)
```

**Figure 5 init.server.lua**

When a user connects to the Roblox server, a listener event PlayerAdded.Connect() is called. Upon receipt of the event, the player id of the user who connected to server is checked against an array of IDs that should be allowed on the server. If the ID's match, the event listener ends its routine and normal game execution occurs. If the ID's do not match, the player is faced with a message "You are not whitelisted on this server." This script was written with assistance from ChatGPT, with respect to taking a string of newline delimited ID tokens and converting into a searchable array.

**ROBLOX 2: Stage and Seating****Figure 6 Folder Directory**

```
"EventModel": {
    "$path": "src/MainMap/EventMap.rbxm"
}
```

**Figure 7 default.project.json object path for EventModel****Figure 8 Object Directory**

The stage consists of a stage, spotlight, ramp, rail guards, curtains and girder beam models pulled from the Roblox Model Search space.

The backdrop and cover around the bottom of the stage was created with scaled rectangular prism parts.

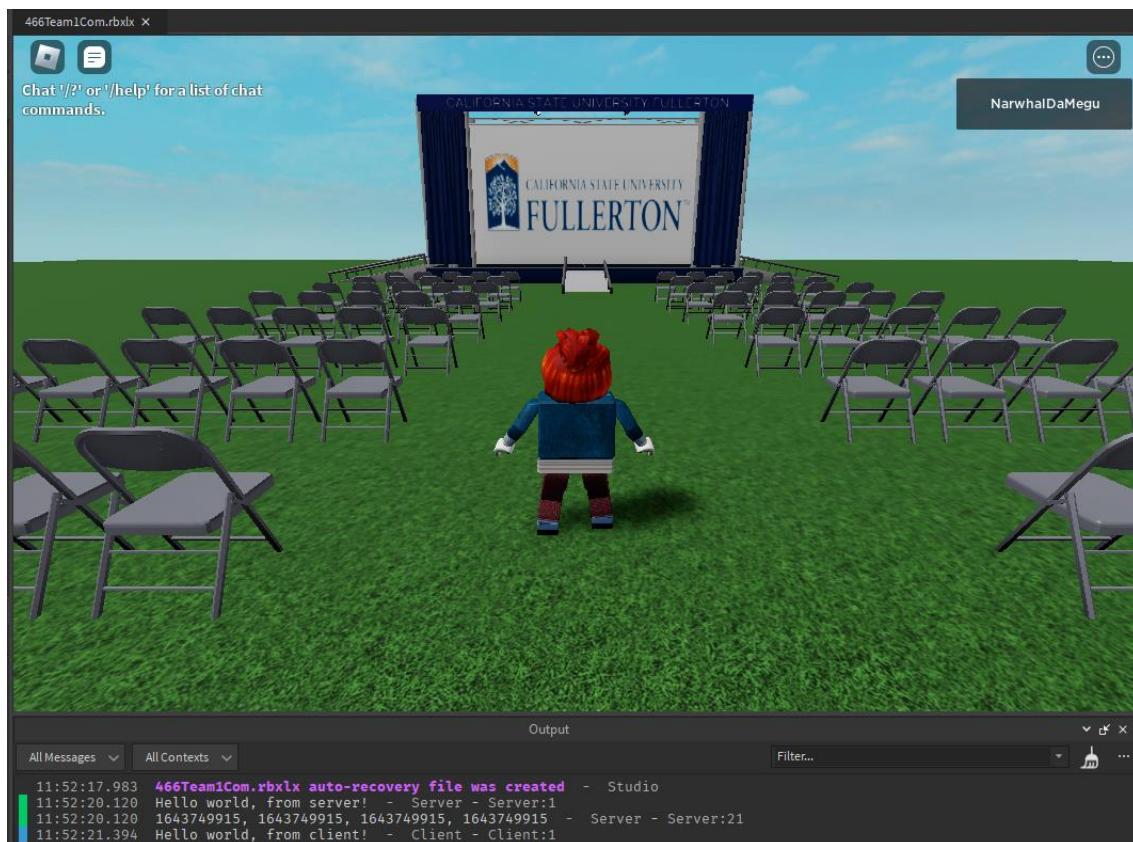
The seating was pulled from the Roblox Model search space, to reduce the need to script animations for seating. Upon walking onto a seat, a user will automatically be attached to the seat in a seated pose.

### ROBLOX-3: Gift Shop model and menu

The Gift shop consists of a shop model, two parts named openPart and closePart, and a ScreenGui. The openPart is used as a trigger to open up a ScreenGui that represents the shop menu, and the closePart is used as the trigger to close the shop menu. Additionally, there is a local script that runs when openPart is triggered. The local script is responsible for making the shop menu visible to the individual users that triggered the openPart. The gift shop transaction is also taken care of by the local script that opens and closes the shop menu. The shop menu has clickable buttons which consist of buttons with item names and an exit button. When the exit button is clicked the shop menu is closed for the user and they can either trigger the shop menu again or return continue exploring the commencement event. However, if they press the buttons with the item names then the local script is in charge of placing the respective item in the users inventory.

#### Deployment and Testing:

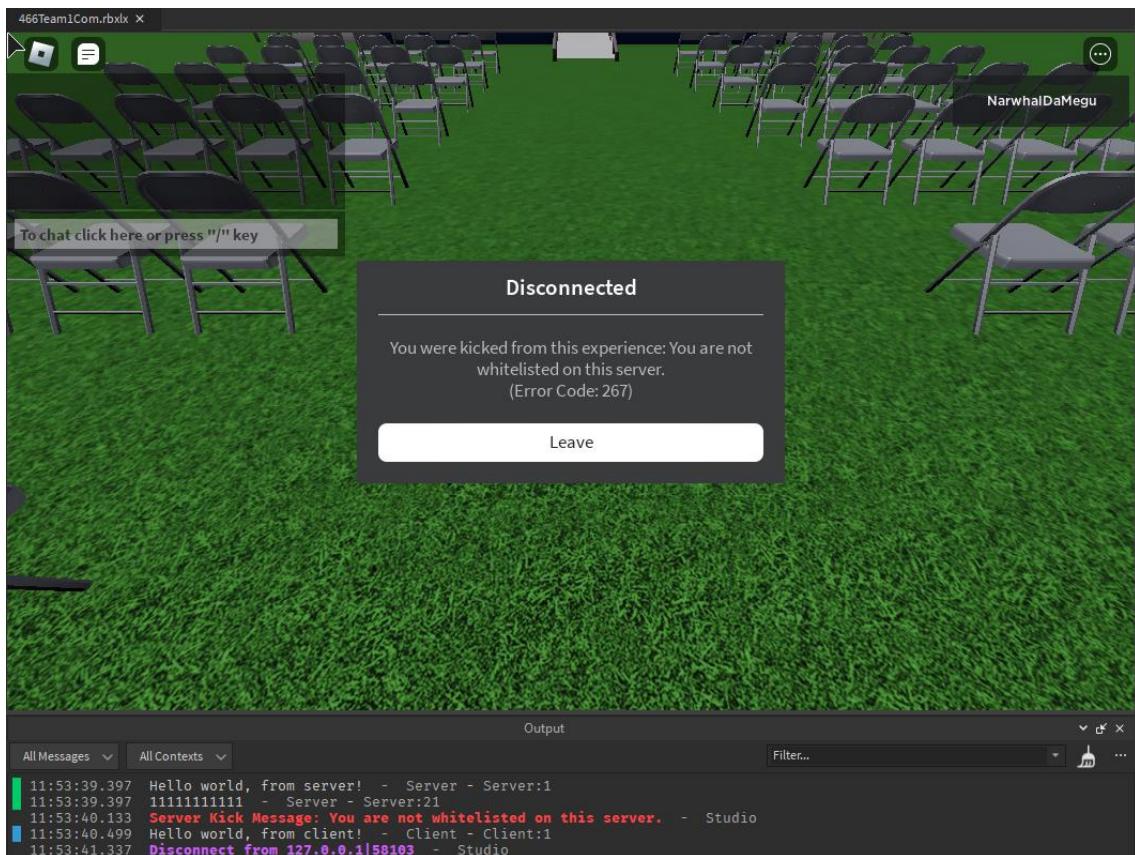
##### ROBLOX-1:



Test Case: Authorized Users in whitelist can join server

Result: Authorized user is able to join server.

PASS



**Figure 9 Client is not on whitelist and server kicks client on join.**

Test Case: Unauthorized Users that is not in whitelist cannot join server.

Result: Unauthorized user is unable to join server.

PASS

### ROBLOX-2: Stage and Seating

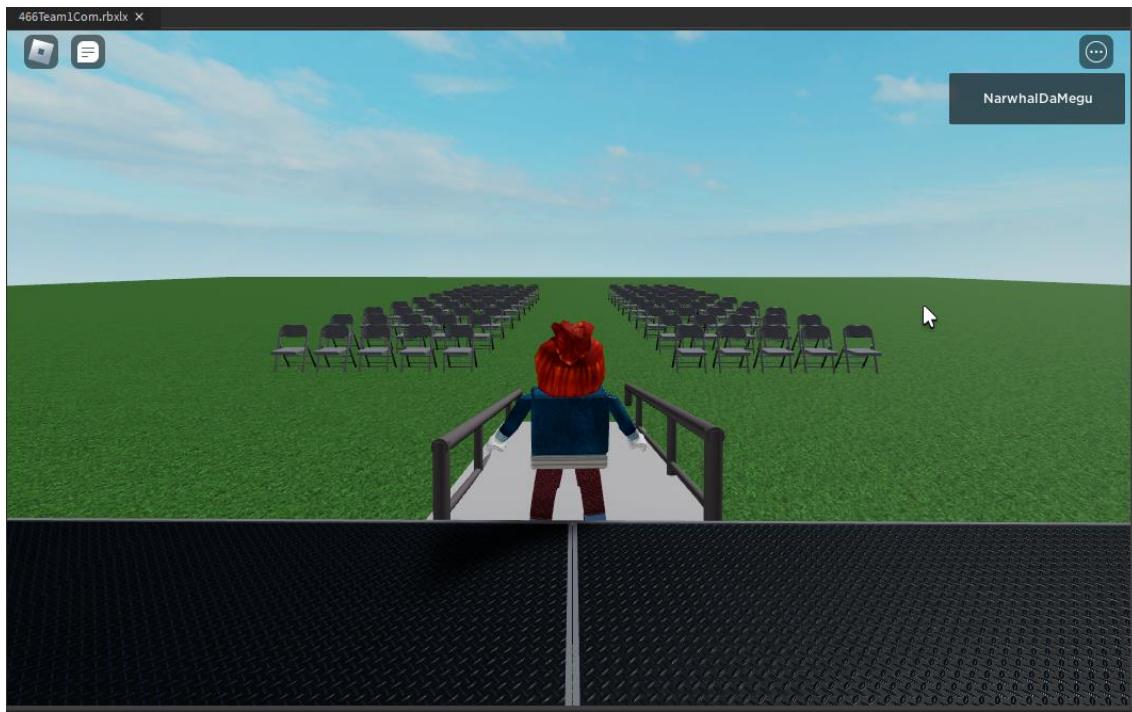


Figure 10 Seating View from Stage



Figure 11 Stage from Seating View

Test Case: Models for Stage and Seating are rendered in the game world, tangible to player, and can be sit on.

Result: Objects appear in world and player can seat in seats.

PASS

**ROBLOX-3: Gift Shop menu and model**



Figure 12 User approaching the Gift shop



Figure 13 The shop menu is triggered when the user approaches

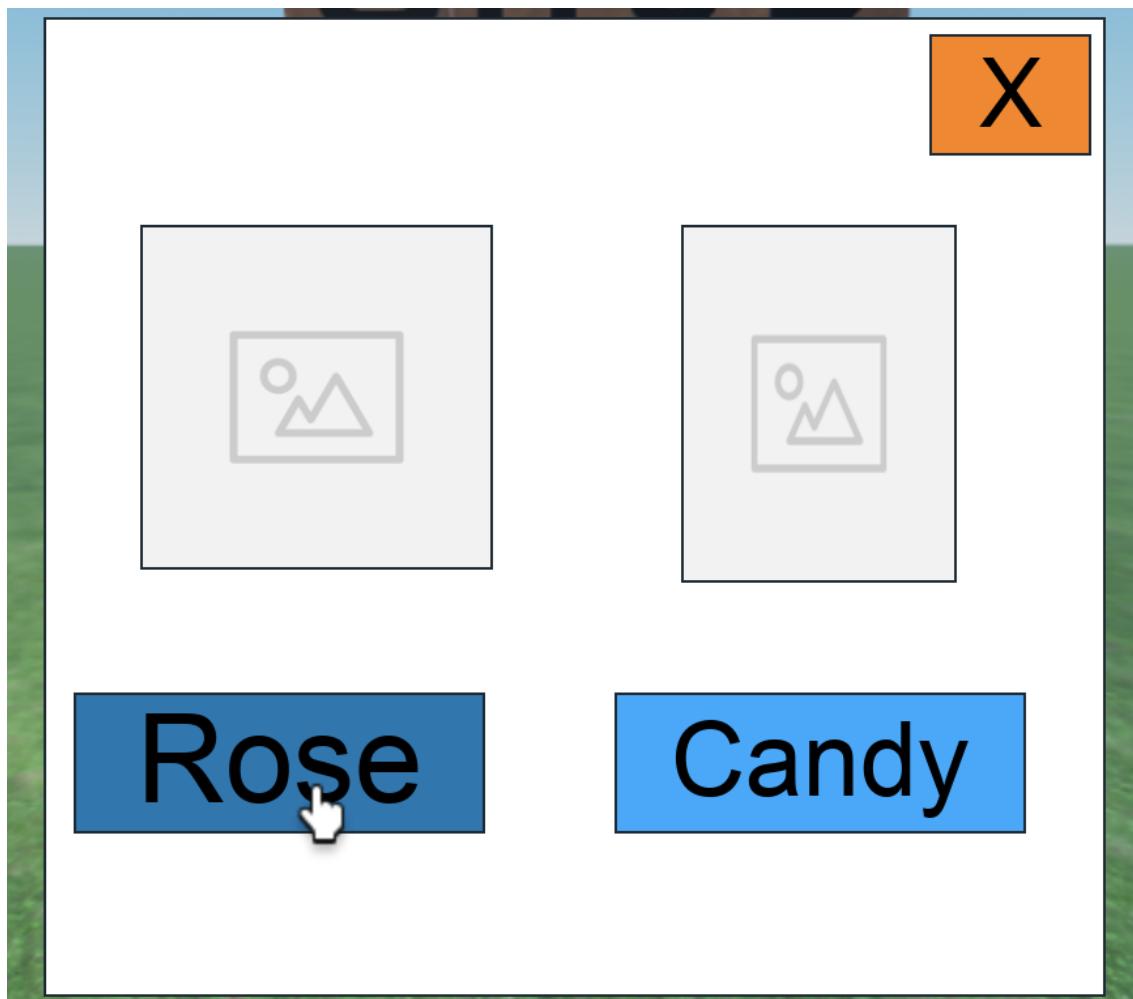


Figure 14 User clicks item button



Figure 15 User clicks on the exit button



**Figure 16 User exits the shop menu and has the items in their inventory**

Test Case: The Shop model is present in the game and the user can approach the shop to trigger the shop menu. Additionally, the user will choose an item and exit the shop menu.

Result: The object menu appears when the user approaches the shop. The user picks an item, exits the menu shop, and they now have the item in their inventory.

PASS

#### ROBLOX-4



Figure 17 Image of the Titan Stadium at CSUF



Figure 18 Replication of the Titan Stadium in Roblox for the Event Ceremony



**Figure 19 Attendee interacting with the environment before the event begins**

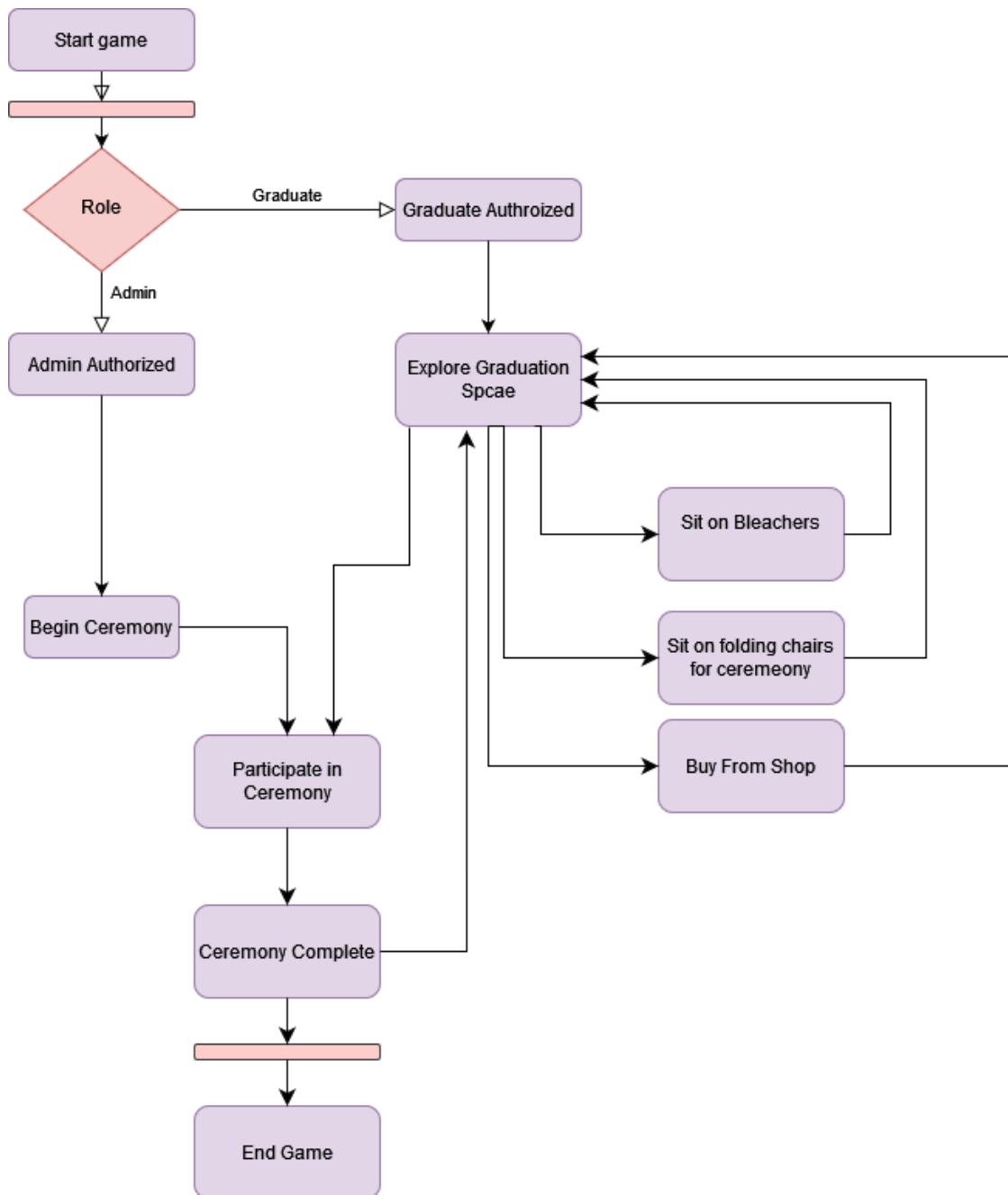


**Figure 20 Animation plays as the event itself is taking place**

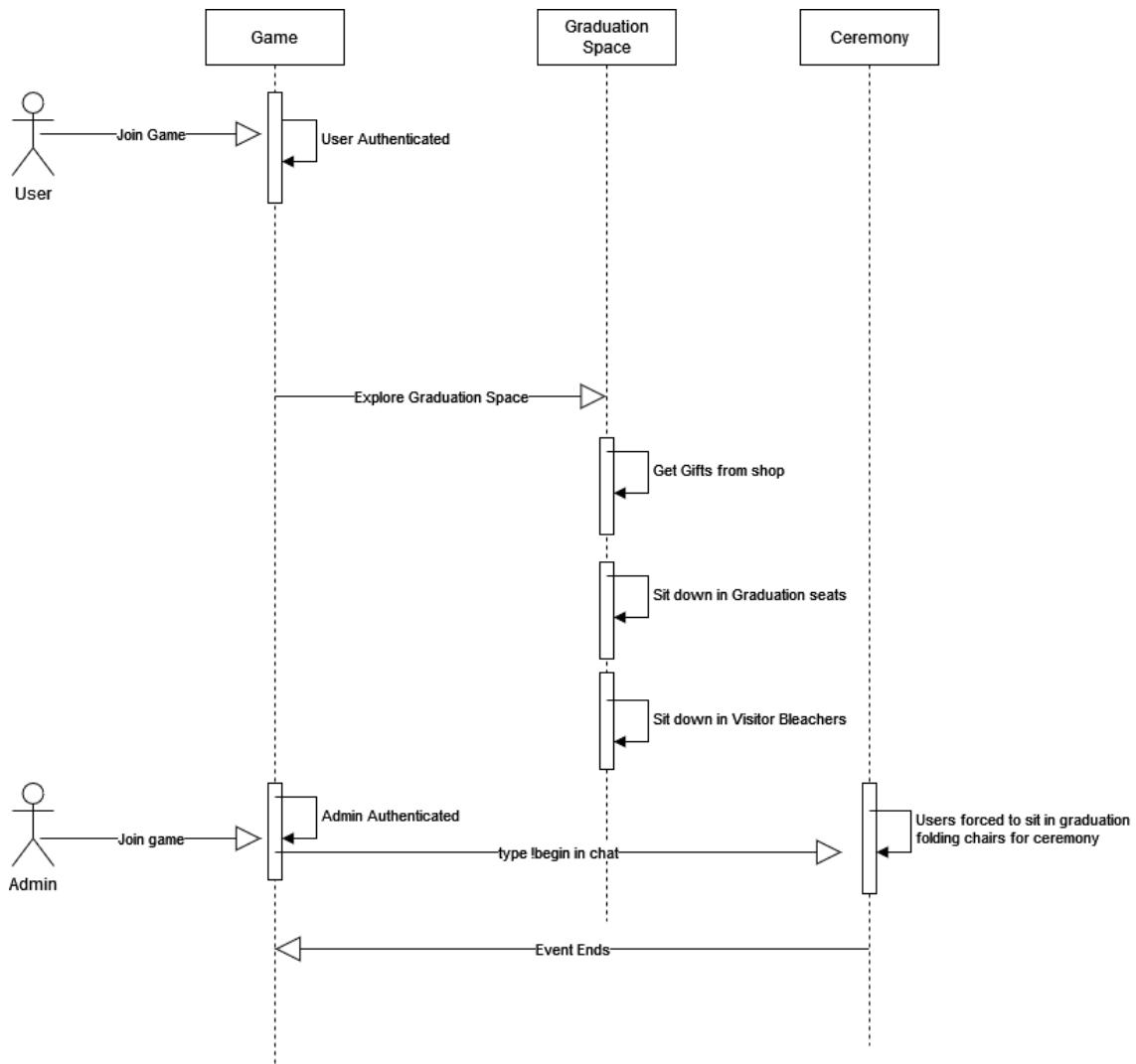
**Test Case:** All objects and models present at the event must be able to have their functionality utilized, while also not allowing the user to move or change the positions of that object or model when the user is exploring the event or when the user is viewing the event ceremony.

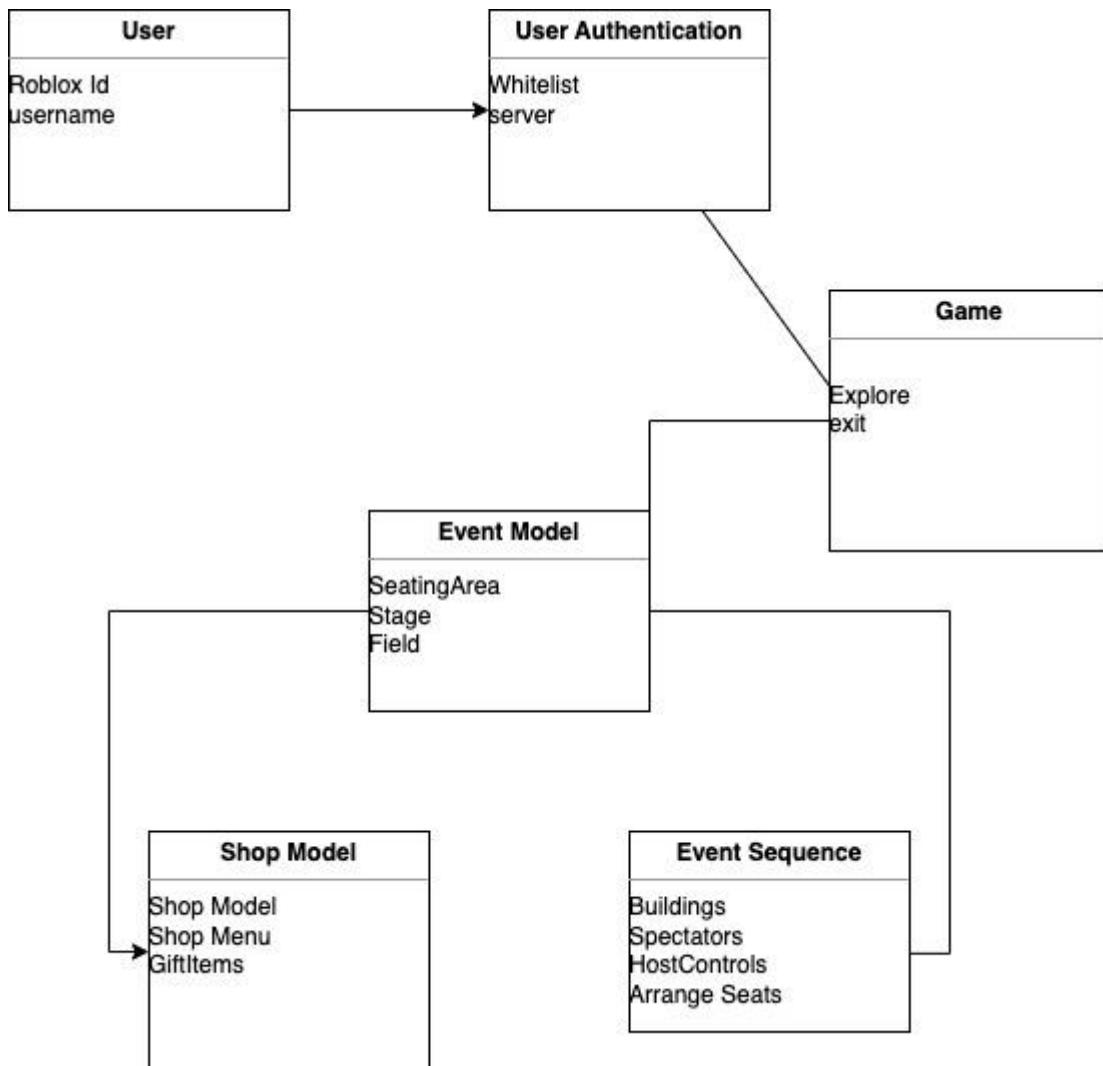
**Result:** There is no object or model, that is either stationary or intractable, that can be moved by the user while they are attending the event.

PASS

**Iteration 2****Analysis and Design:****Activity/Interaction Diagram**

### System Sequence Diagram



**High Level Class Diagram****Figure 21** High Level Class Diagram

**Deployment and Testing:****ROBLOX-1 Unauthorized Users Documentation**

To setup the whitelist for the server, copy this form and send it to the emails of CSUF students who will participate in commencement.

[Link to Google Form Template](https://docs.google.com/forms/d/e/1FAIpQLSebq_JHxbHhwUNtl7uYmqNWEZgCZh1mMD6IMNyLY--g_Z_U0w/viewform?usp=sharing)

**Setting Up the Form**

## CSUF Roblox Commencement

Event Timing: ENTER DATE

Event Address: ENTER ROBLOX GAME LINK

Contact us at ENTER PHONENUMBER or ENTEREMAIL



christopher2melgar@gmail.com (not shared) [Switch account](#)



Resubmit to save

\* Required

CWID \*

1111111112

First Name \*

Kris

Last Name \*

Melgar Morales

Sign into Roblox and click Profile



Copy the 10 digit Roblox ID from the URL Link once in your Roblox Profile



Enter your unique Roblox ID \*



1643749915

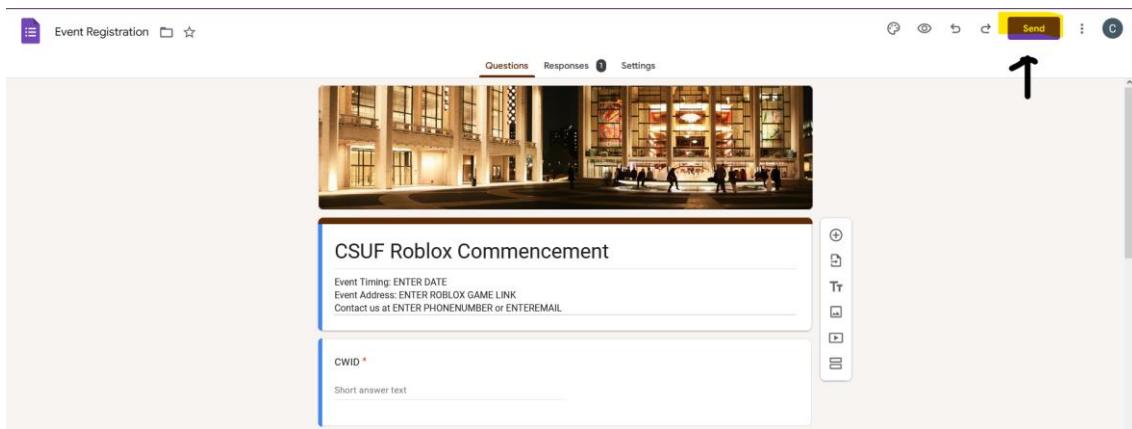


Figure 22 Click send

Send form X

Collect emails

Send via Email Link Share Facebook Twitter

Email

To: cmelgarmorales@csu.fullerton.edu

Subject: CSUF Roblox Commencement

Message

I've invited you to fill out a form:

Include form in email

+ Add editor Cancel Send

Figure 23 Enter the emails of the CSUF Students and hit Send

---



The screenshot shows the Google Forms interface for a form titled "Event Registration". The "Responses" tab is selected, showing "1 response". Below the responses, there are two main sections: "View in Sheets" (with a three-dot menu icon) and "Accepting responses" (with a toggle switch set to "on").

**Figure 24 Click the three dots next to “View in Sheets” hyperlink.**

---



The screenshot shows the Google Forms interface for a form titled "Event Registration". The "Responses" tab is selected, showing "1 response". A modal dialog is open, titled "Select destination for responses", containing two options: "Get email notifications for new responses" and "Select destination for responses".

**Figure 25 Click select destination for responses to set up the sheet where all responses will be stored**

#### Modifying the Results to move to Rojo and Roblox

[https://docs.google.com/spreadsheets/d/1ZRV59cwgBhhSrjMb\\_ql8gmrQywdw17xza\\_Yh\\_x7QU4/edit?usp=sharing](https://docs.google.com/spreadsheets/d/1ZRV59cwgBhhSrjMb_ql8gmrQywdw17xza_Yh_x7QU4/edit?usp=sharing) Link to Google Spreadsheet Results from Form with 1 response submitted.

You will need to Make a Copy of the form if you do not want to delete the information from the Responses

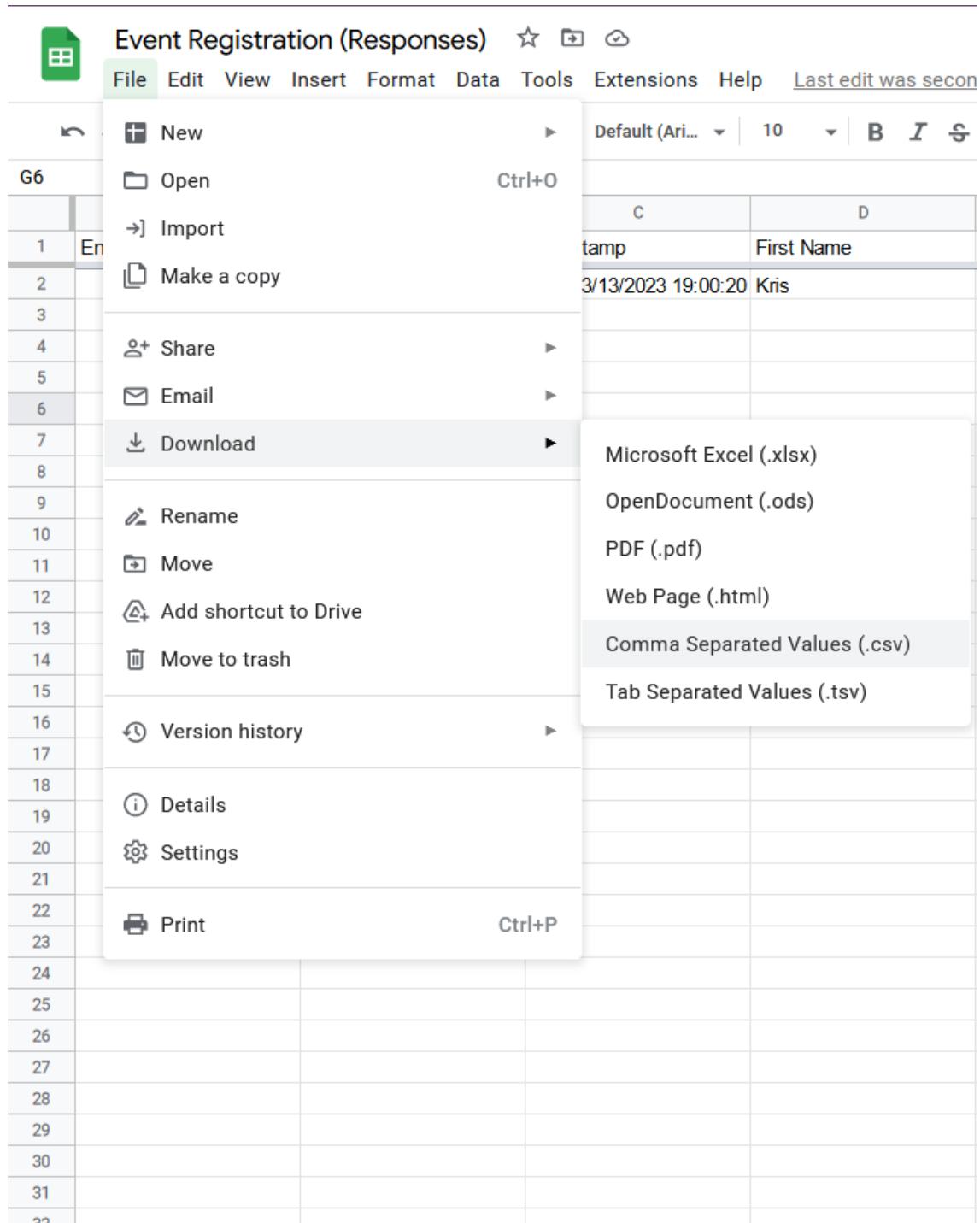
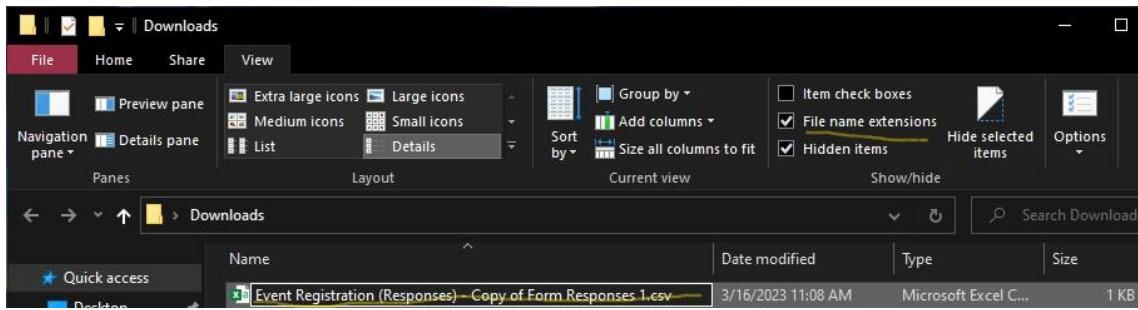
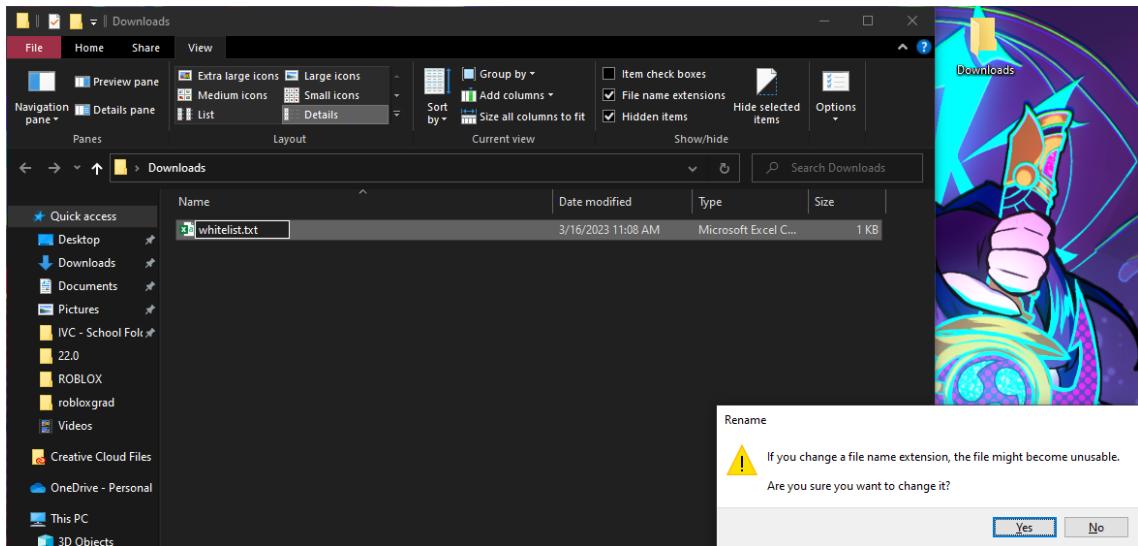


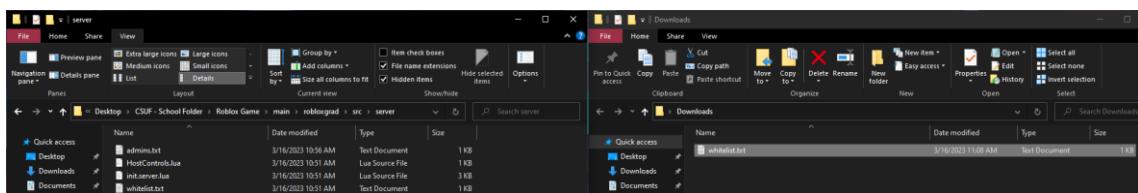
Figure 26 In the Response sheet, download the response as a comma separated values file (.csv), and ensure that the first column of data is the Roblox IDs



**Figure 27** In your file explorer, ensure that file name extensions are enabled



**Figure 28** Rename the file to whitelist.txt and click yes to confirm the change.



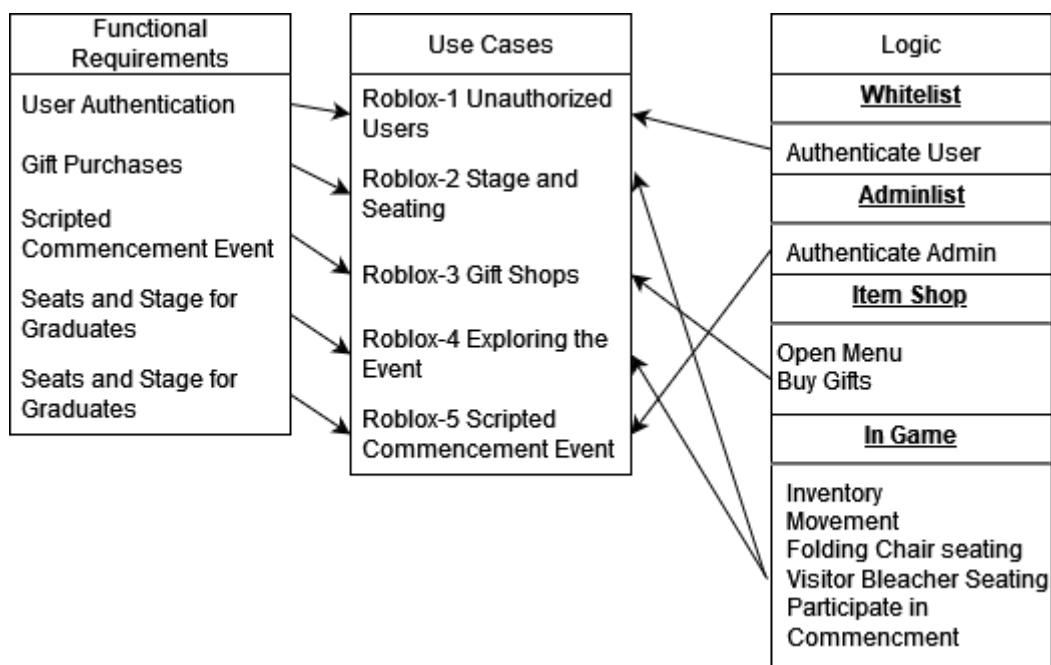
**Figure 29** Before rebuilding the Roblox Game, move whitelist.txt to the file path src/server relative to the repository download of the event.

**Traceability Matrix**

Requirement Details	User Story	Requirement Name	Diagram ID	Expected Result	Status
Graduation area and stage are represented in the Roblox world for users to sit in to wait to and get their diplomas.	Alan enter the graduation event area, and sees the seating area for graduates, accompanied by a stage.	Stage and Seating Models	ROBLOX-2	Models are visible in game, chairs can be sit on	OK
Unauthorized users are not able to enter the commencement event.	Alex has shared a link/invite with his friend to the commencement event. Alex is not authorized to have a link to the event. When Alex attempt to join the event, he is blocked from entering the server.	Unauthorized Users	ROBLOX-1	Regular Flow: Authorized user on whitelist is able to join server  Alternate Flow: Unauthorized users not on whitelist is unable to join server	Regular Flow: OK  Alternate Flow:OK
Attendees are able to purchase gifts from the gift shop	Bob is attending the commencement as a guest to watch his friend David receive his diploma. At the commencement he will be able to buy a gift to give to his friend after he has received his diploma.	Gift Shop	Roblox-3	Shop model is visible and the user is able to interact with the shop menu in order to retrieve a gift.	OK
A scripted movie will play when the commencement begins and graduate will be able to walk up to the stage and receive a diploma	Once all of the players are ready, the host will begin the commencement by playing a scripted movie. Rick is a graduate and he will walk up to the stage and receive his diploma when it is his turn.	Begin Commencement	Roblox-5	When executed, players are assigned to seats and the graduation ceremony begins	OK
Users can explore event and interact with other people before and after the commencement event	Jim is an attendee at the commencement event, viewing David graduate. While waiting for the event to begin, Jim can walk around the buildings surrounding the commencement area and interact	Exploring the event	Roblox-4	Models for event appear in Roblox Studio, and are properly anchored to world.	OK

Requirement Details	User Story	Requirement Name	Diagram ID	Expected Result	Status
	with other students and visitors.				

### Traceability Diagram



**Iteration 3**

We added a new construction phase to maintain the previous report's work. We expect to be fine without any changes to support this addition concerning the transition phase and documentation for setting up the Roblox Environment.

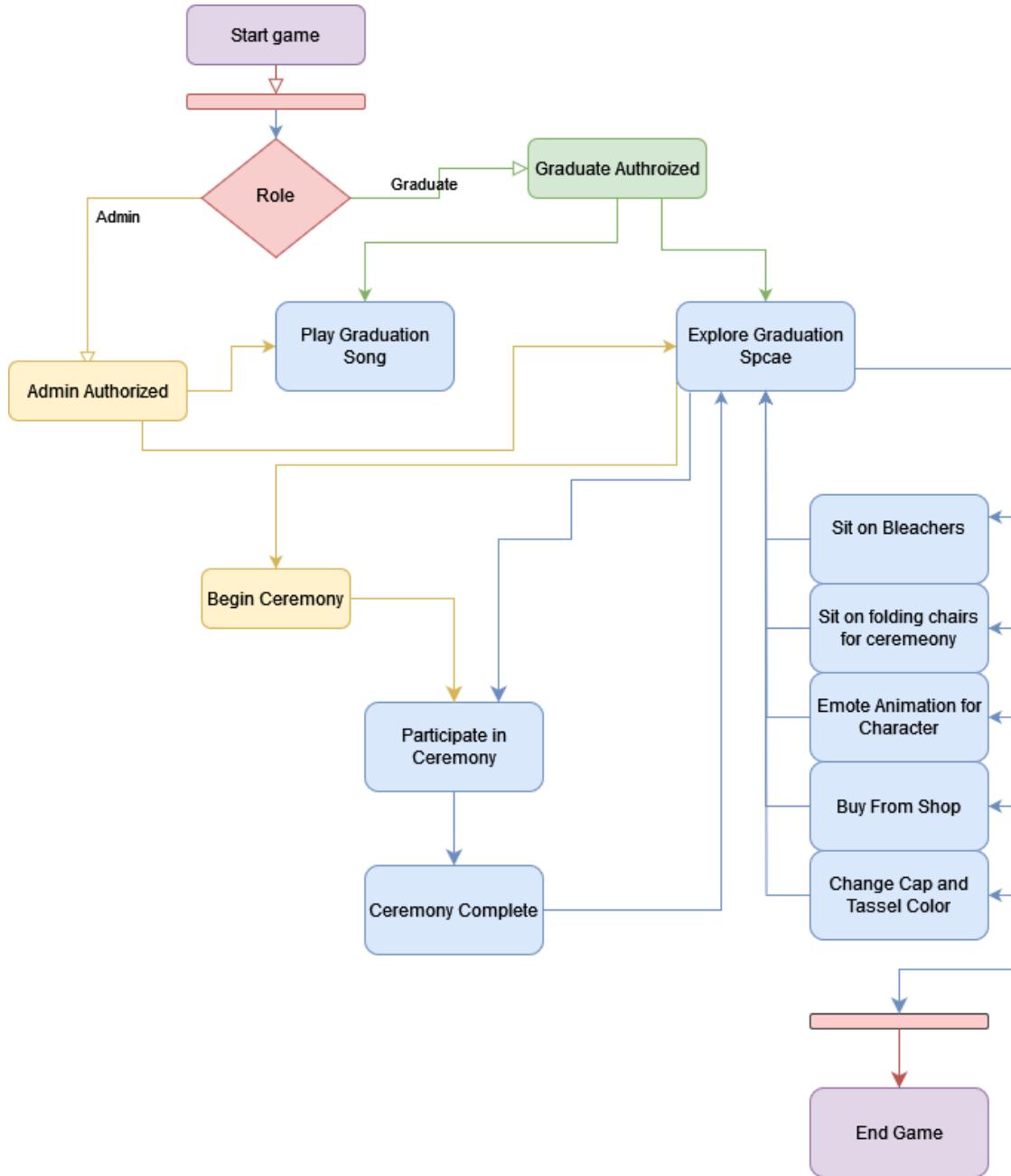
**Requirements:**

Requirement	User Story	Importance	Diagram ID	Taken By
Graduates hear March no. 1 - Pomp and Circumstance when they enter the game world	When Kylie enters the game event, he turns on his speakers/headphones. From the audio device, they hear March no. 1 - Pomp and Circumstance and are at peace.	LOW	ID-ROBLOX-7	 <a href="#">Kris</a> <a href="#">Melgar</a> <a href="#">Morales</a>
Graduates are able to select cap and tassel to wear before the graduation event.	When Ricardo enters Fullerton Commencement from the Roblox Launcher, he is greeted to a screen with colors to choose for their graduation cap and tassel. When he enters the game world, he sees that his roblox avatar is wearing the selected outfit. He also can reselect the cap color at any time.	LOW	ID-ROBLOX-6	Kris
When users get their diploma, they can choose a preset choice of emotes to do.	Bob is about to go up to receive her diploma. She shes a window on her screen that show different emotes that she can prepare when she goes up on stage. She clicks an emote, and successfully preforms the selected emote.	LOW-MID	ID-ROBLOX-8	Mario Linares

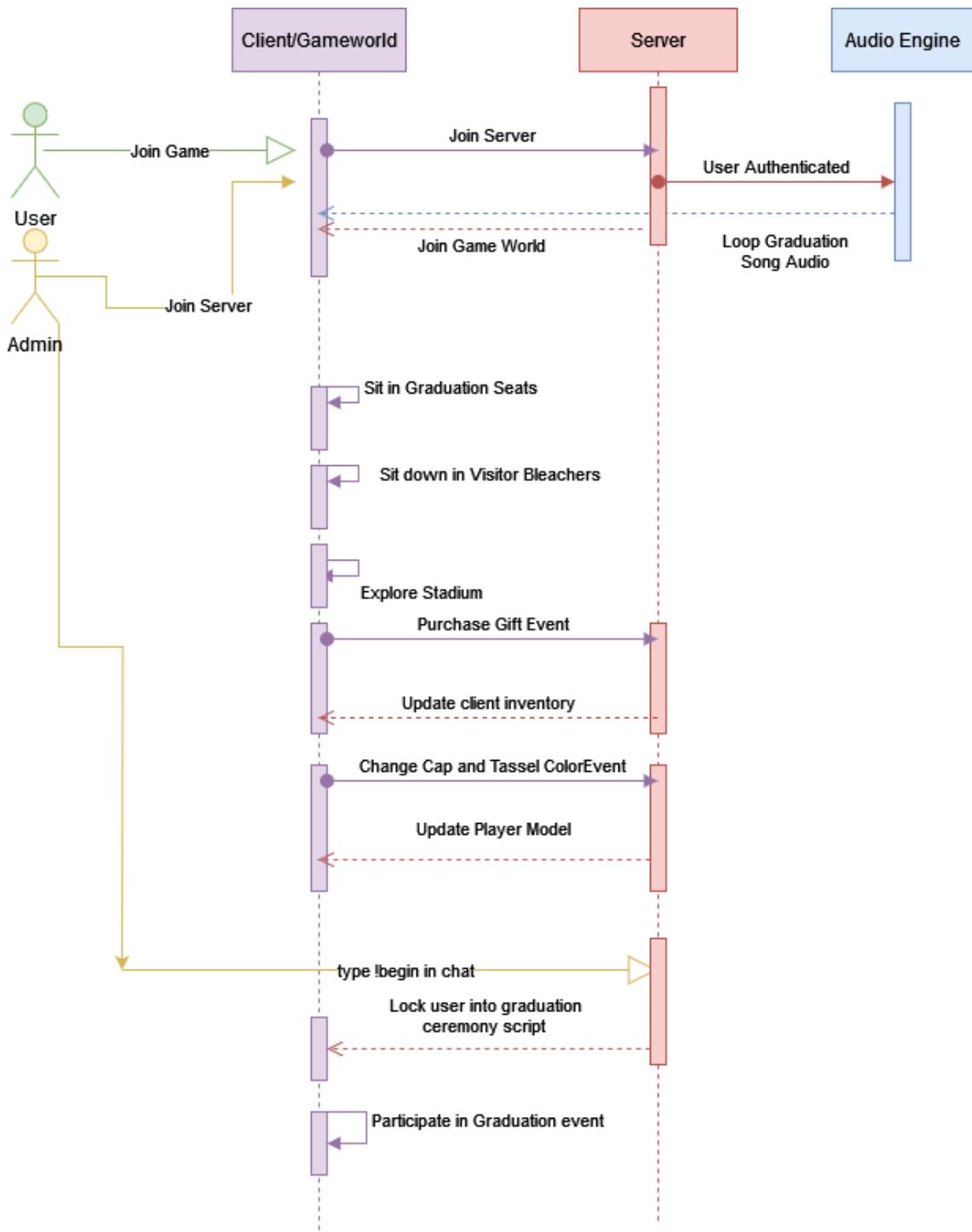
## Analysis and Design:

### Activity/Interaction Diagram

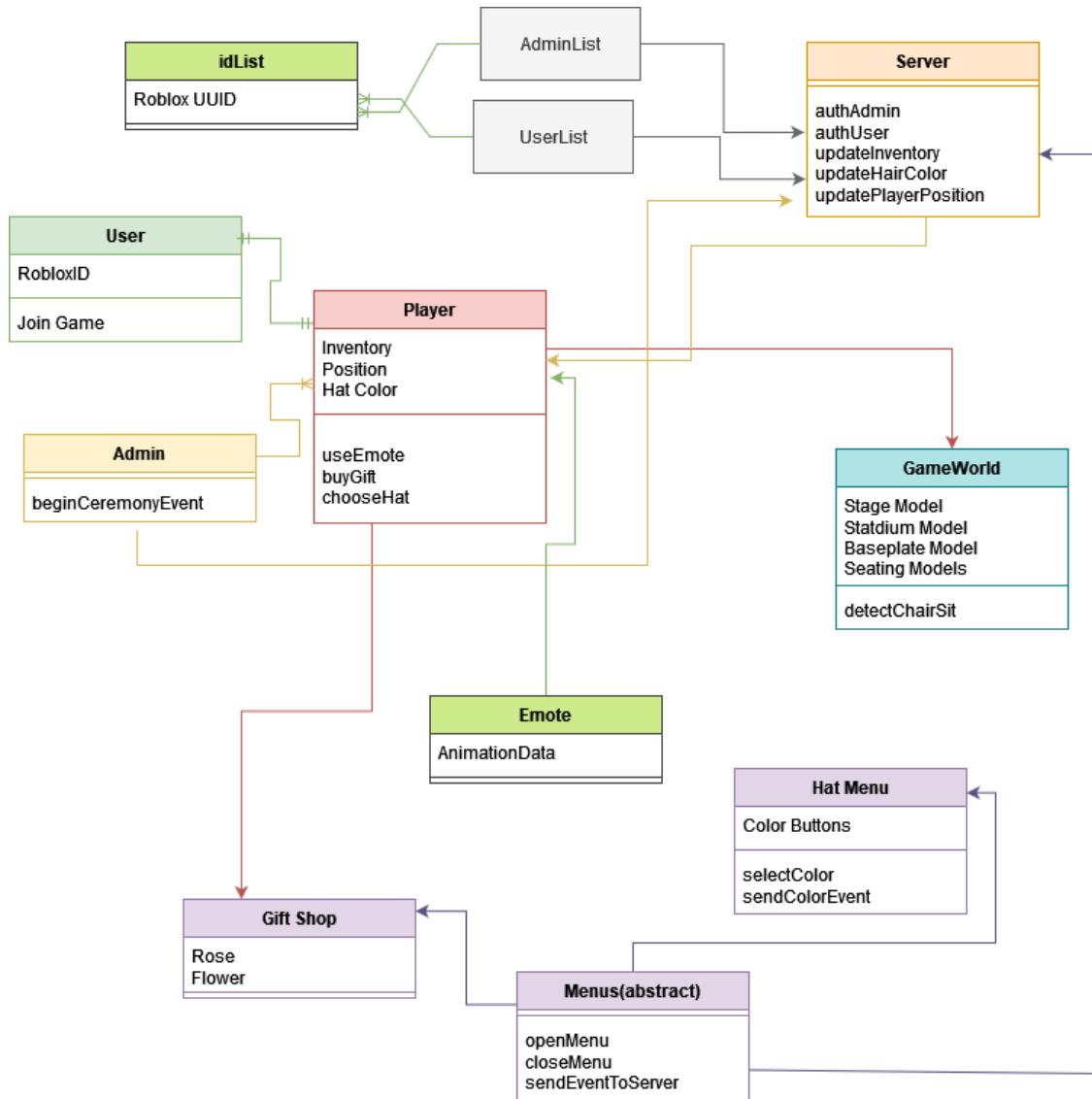
Edits to the original diagram were made, which include adding the new functional requirements as part of the diagram flow, as well as additional colors to clarify the flow of the graph.



### System Sequence Diagram



### High Level Class Diagram



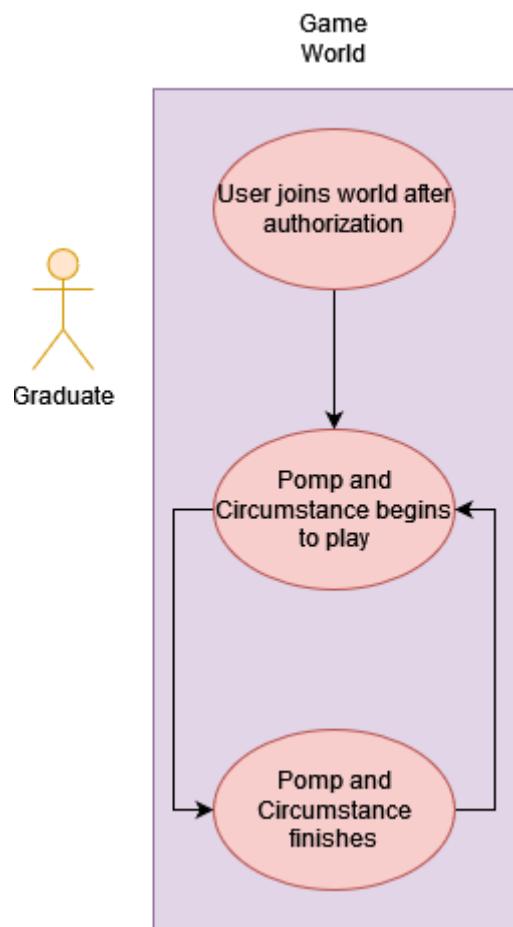
Version	Date Created	Originator	Changes Made
0.1	5/8/2023	Kris Melgar Morales	Created use case for graduation audio

Requirement	User Story	Importance Taken By
Graduates hear March no. 1 - Pomp and Circumstance when they enter the game world	When Kylie enters the game event, he turns on his speakers/headphones. From the audio device, they hear March no. 1 - Pomp and Circumstance and are at peace.	<span style="background-color: red; color: white; padding: 2px 5px;">LOW</span>  <a href="#">Kris</a> <a href="#">Melgar</a> <a href="#">Morales</a>

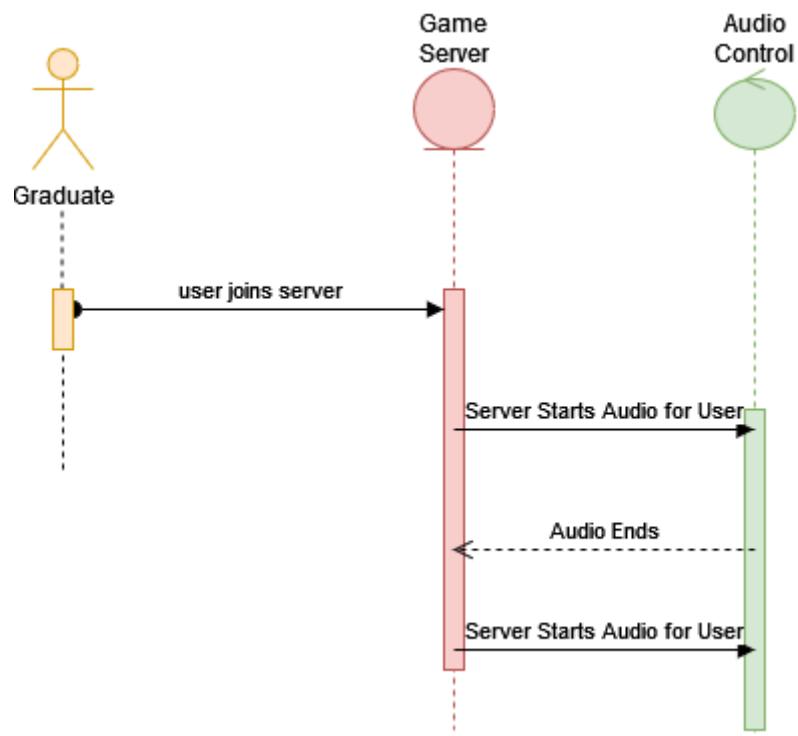
## Component basics

Diagram ID	ID-Roblox-7
Description	For the Roblox commencement event, graduates should be able to hear special graduation music from there speakers
Primary Roles	Graduates
Pre-Conditions	User is logged into Roblox. User has been authorized to join the roblox event.
Trigger	User joins the server.
Basic Flow	1. User enters Game world after authorization 2. If user has audio device connected, user begins to hear Pomp and Circumstance 3. Pomp and Circumstance loops when complete.
Post-Conditions	Audio Loop for Pomp and Circumstance is created.

## Use Case Diagram



## Sequence Diagram



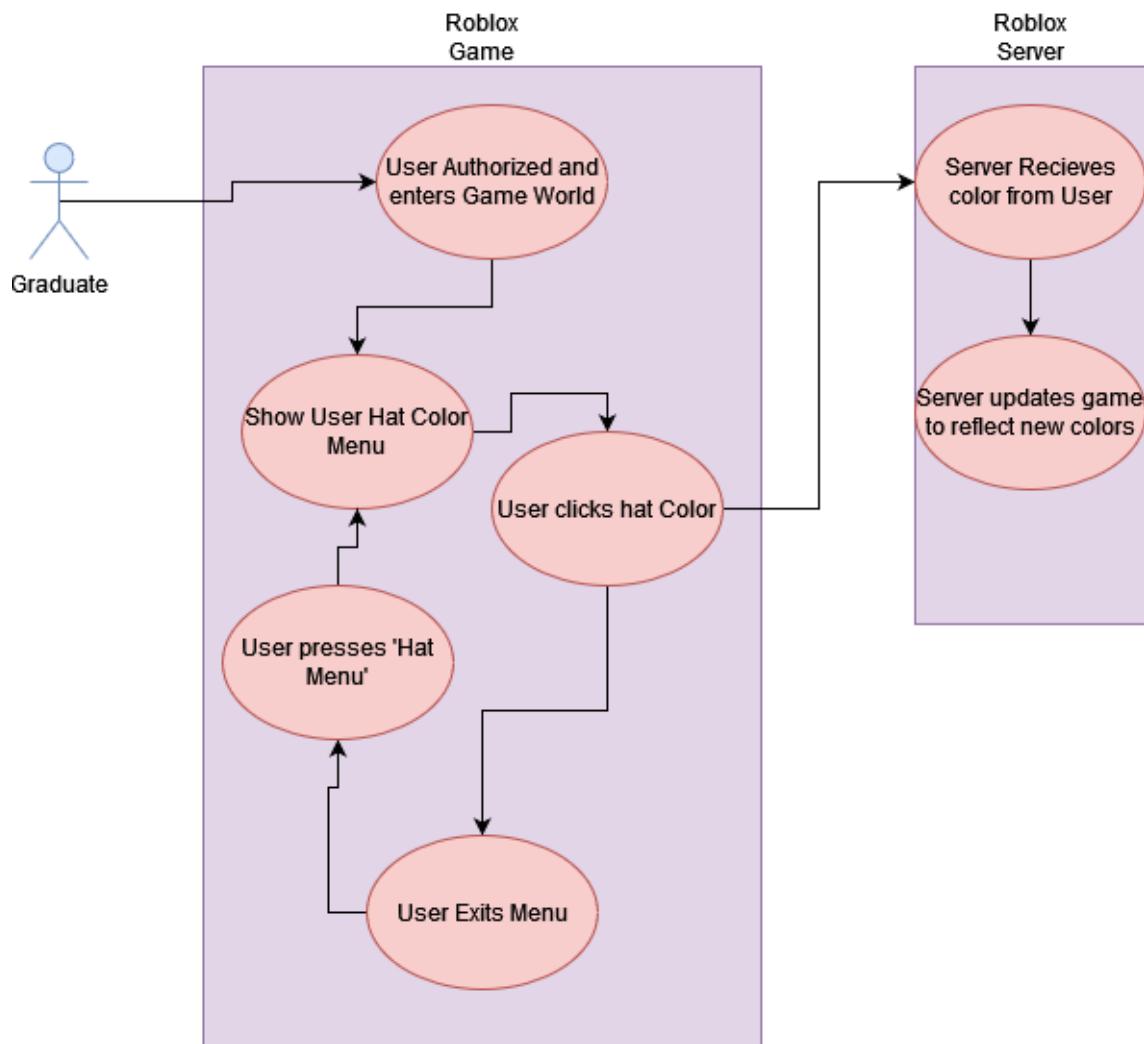
Version	Date Created	Originator	Changes Made
0.1	5/8/2023	Kris Melgar Morales	Created use case for graduation cap on user

Requirement	User Story	Importance	Taken By
Graduates are able to select cap and tassel to wear before the graduation event.	When Ricardo enters Fullerton Commencement from the Roblox Launcher, he is greeted to a screen with colors to choose for their graduation cap and tassel. When he enters the game world, he sees that his roblox avatar is wearing the selected outfit. He also can reselect the cap color at any time.	LOW	 <a href="#">Kris</a> <a href="#">Melgar</a> <a href="#">Morales</a>

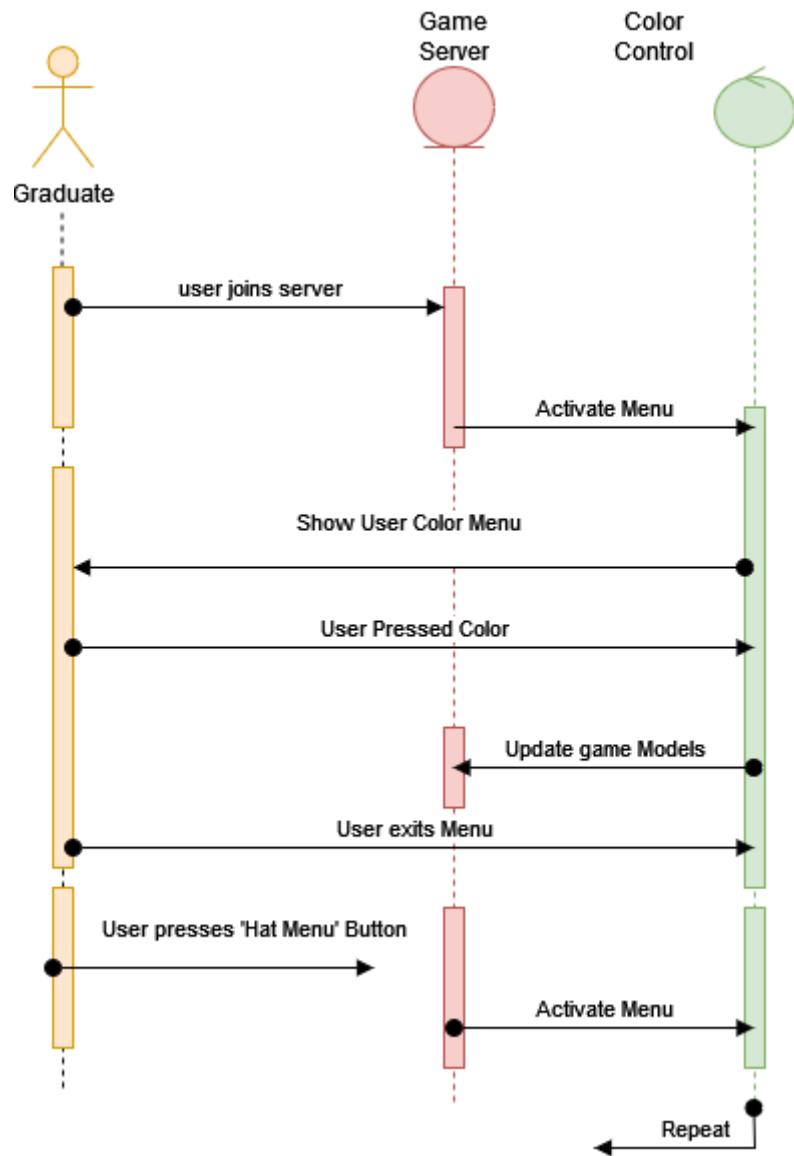
## Component basics

Diagram ID	ID-Roblox-6
Description	For the Roblox commencement event, graduates should be able to wear and cap and tassel, with their desired colors for the event.
Primary Roles	Graduates
Pre-Conditions	User is logged into Roblox. User has been authorized to join the roblox event.
Trigger	User joins the server.
Basic Flow	<ol style="list-style-type: none"> <li>1. User enters Game world after authorization</li> <li>2. User sees menu containing cap and tassel colors to choose from</li> <li>3. User selects a color and exits the menu.</li> <li>4. User's character has updated on server to reflect new model.</li> <li>5. User clicks button that says 'Hat Menu'</li> <li>6. User repeats step 2-4 after menu opens again.</li> </ol>
Post-Conditions	Users who selected a cap and tassel color have their models updated appropriately

## Use Case Diagram



## Sequence Diagram



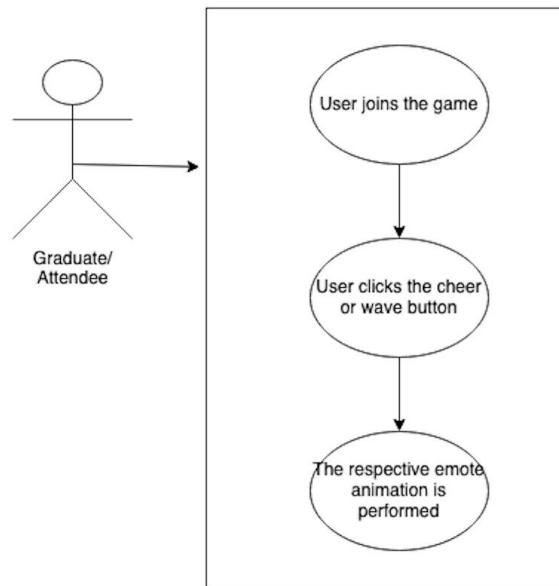
Version	Date Created	Originator	Changes Made
0.1	5/8/2023	Mario Linares	Created use case for Emote

Requirement	User Story	Importance	Taken By
When users get their diploma, they can choose a preset choice of emotes to do.	Bob is about to go up to receive her diploma. She shes a window on her screen that show different emotes that she can prepare when she goes up on stage. She clicks an emote, and successfully preforms the selected emote.	LOW-MID	 <a href="#">Mario</a> <a href="#">Linares</a> Mario Linares

## Component basics

Diagram ID	ID-Roblox-8
Description	When users get their diploma, they can choose a preset choice of emotes to do.
Primary Roles	Graduate and Visitor
Pre-Conditions	User is authorized to be present at the event.
Trigger	User hits one of the buttons on the bottom left hand side of the screen. The button when clicked will play the corresponding emote.
Basic Flow	<ol style="list-style-type: none"> <li>1. User enters the game world after they have been authorized to be present</li> <li>2. User clicks either the cheer or wave button</li> <li>3. The user performs an emote that corresponds to the button pressed</li> </ol>
Post-Conditions	N/A

## Use Case Diagram



**Figure 30 Use Case Diagram: Emotes**

## Sequence Diagram

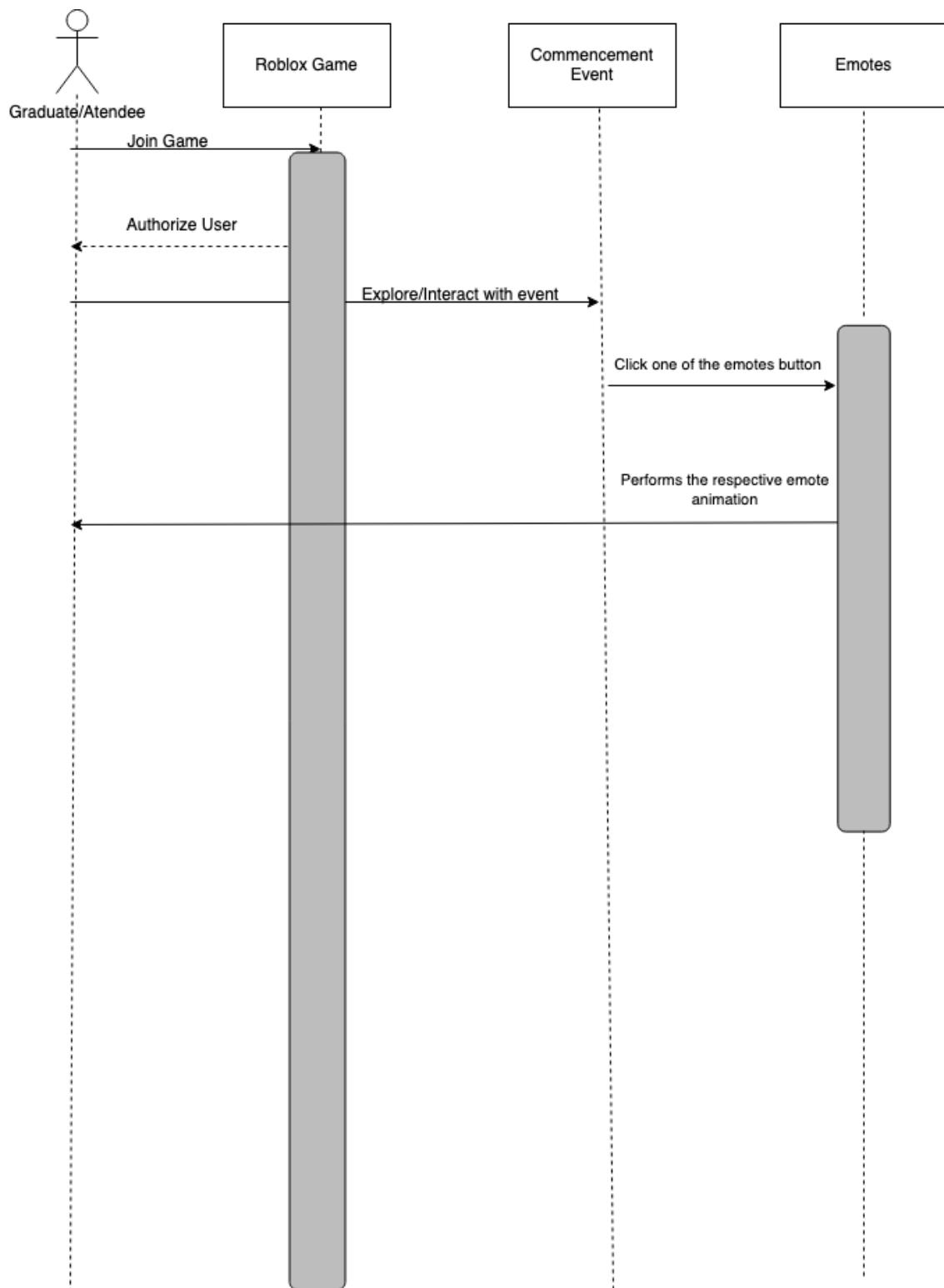


Figure 31 Sequence Diagram: Emotes

**Implementation:****Roblox 6: Graduation Cap**

Here is Layout for the implementation of the Graduation cap functional requirement

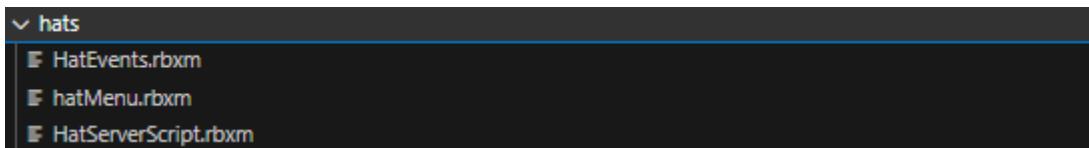


Figure 32 Rojo Format Folder Directory

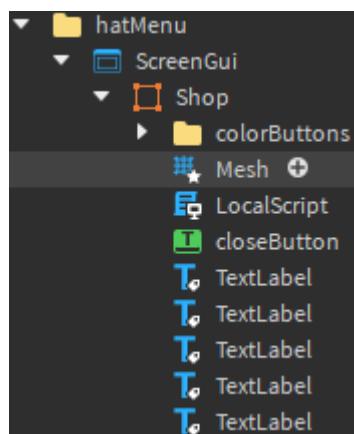


Figure 33 Hat Menu Object Layout

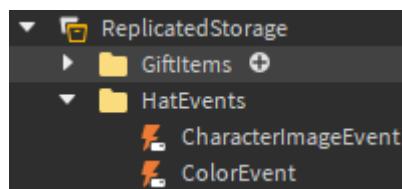


Figure 34 Hat Menu Color Event Layout

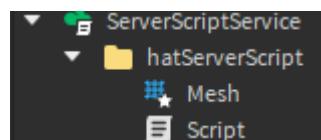


Figure 35 Hat Menu Server Script Layout

```

1 local frame = script.Parent
2 local closeButton = Frame.CloseButton
3 local buttonFolder = frame.ColorButtons
4 frame.Visible = true
5
6 local openButton = Instance.new("TextButton")
7 openButton.Parent = Frame.Parent
8 openButton.Position = UDim2.new(0.056, 0, 0.908, 0)
9 openButton.Size = UDim2.new(0, 50, 0, 50)
10 openButton.Text = "Hats"
11 openButton.Visible = false
12
13 local function toggleVisibility()
14     -- Set the visibility of the open button based on the visibility of the menu frame
15     if frame.Visible == false then
16         openButton.Visible = true
17     else
18         openButton.Visible = false
19     end
20 end
21
22 local function openMenu(nextPart)
23     local player = game.Players.LocalPlayer
24     if player then
25         player.PlayerGui.hatMenu.ScreenGui.Shop.Visible = true
26         player.Character.Humanoid.WalkSpeed = 16
27         toggleVisibility()
28     end
29 end
30
31 local function closeMenu()
32     local player = game.Players.LocalPlayer
33     if player then
34         player.PlayerGui.hatMenu.ScreenGui.Shop.Visible = false
35         player.Character.Humanoid.WalkSpeed = 16
36         toggleVisibility()
37     end
38 end
39
40
41
42 closeButton.MouseButton1Click:Connect(closeMenu)
43
44 -- Connect the openButton's MouseButton1Click event to the openMenu function
45 openButton.MouseButton1Click:Connect(openMenu)
46
47
48 -- Get a reference to the RemoteEvent object in ReplicatedStorage
49 local colorEvent = game.ReplicatedStorage.HatEvents:WaitForChild("ColorEvent")
50
51 -- Iterate over all buttons in buttonFolder and connect their MouseButton1Click event to a function that fires the RemoteEvent with the button's BackgroundColor3 value as an argument.
52 for button in pairs(buttonFolder:GetChildren()) do
53     if button:isA("TextButton") then
54         button.MouseButton1Click:Connect(function()
55             colorEvent:FireServer(button.BackgroundColor3)
56         end)
57     end
58 end
59
60
61

```

**Figure 36 Local Script for Menu**

This script controls the Color Menu for the Graduation Caps on the Client side. Upon loading of this script, the menu is made visible and loaded with controls for colors and a close button. When the user chooses a color, the client sends a 'ColorEvent' to the server with the button's color included. When the user presses the "Close" button, the menu becomes invisible while loading a new button to reopen the menu. When the user presses a new button to reopen the menu, the Color menu opens like before.

```

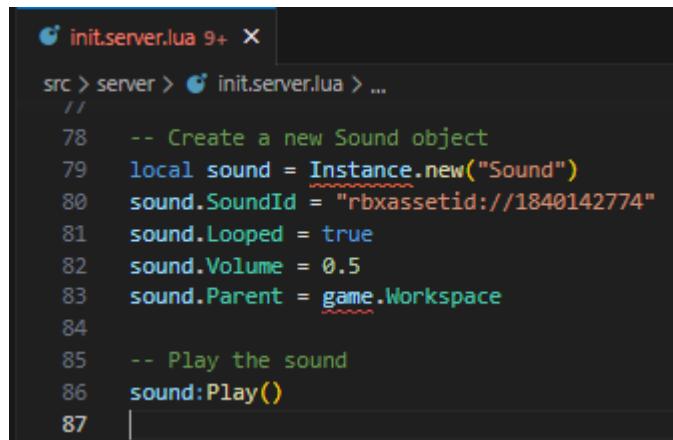
1 -- Server script
2
3 -- Get a reference to the RemoteEvent object in ReplicatedStorage
4 local colorEvent = game.ReplicatedStorage.HatEvents:WaitForChild("ColorEvent")
5
6 -- Connect the RemoteEvent's OnServerEvent event to a function that takes a player and color value as arguments and prints the color value to the console.
7 colorEvent.OnServerEvent:Connect(function(player, color)
8     -- Create a new Hat object
9     local h = Instance.new("Hat")
10    local p = Instance.new("Part")
11
12    -- Remove all accessories from the player's character
13    for accessory in pairs(player.Character:GetChildren()) do
14        if accessory:isA("Accessory") then
15            accessory:Destroy()
16        end
17    end
18
19    h.Name = "Red's Mask"
20    p.Parent = h
21    -- Set the position of the hat to the position of the player's head
22
23    p.Position = player.Character:FindFirstChild("Head").Position
24    p.Name = "Handle"
25    p.formFactor = 0
26    p.Size = Vector3.new(0,-0.25,0)
27    p.BottomSurface = 0
28    p.TopSurface = 0
29
30    p.Locked = true
31    -- Set the BrickColor of the Part object to a new BrickColor created from the received color value
32    p.BrickColor = BrickColor.new(color)
33    -- Clone the mesh and set its VertexColor property to match the received color value
34    local mesh = script.Parent.Mesh:Clone()
35    mesh.VertexColor = Vector3.new(color.r, color.g, color.b)
36    mesh.Parent = p
37    h.Parent = player.Character
38    h.AttachmentPos = Vector3.new(0,-0.1,-0)
39
40 end)

```

**Figure 37 Server Script for Menu**

This script handles the 'ColorEvent' created by a user when they press a button in the menu and modifies the character model. The code looks for any accessories on the player (i.e., hair) and removes them from the character, as well as attaching a model mesh to the player's head with the appropriate color received from the player.

### Roblox 7: Graduation Music



The screenshot shows a code editor window titled "init.server.lua 9+ X". The file path is "src > server > init.server.lua > ...". The code is as follows:

```
src > server > init.server.lua > ...
// 
78  -- Create a new Sound object
79  local sound = Instance.new("Sound")
80  sound.SoundId = "rbxassetid://1840142774"
81  sound.Looped = true
82  sound.Volume = 0.5
83  sound.Parent = game.Workspace
84
85  -- Play the sound
86  sound:Play()
87 |
```

Figure 38 Code added to startup script

For our event, we needed a way to loop Pomp and Circumstance, the song often used in graduation events. Luckily, the audio track was already in the Roblox Asset sound library, so we added a small piece of code to the server startup scripts to play the audio track and loop it until the user chooses to exit the game.

### Roblox 8: Emotes

There will be two buttons on the bottom left side with the “Wave” and “Cheer” text. The game runs a script that triggers the respective emote whenever the user selects either button.

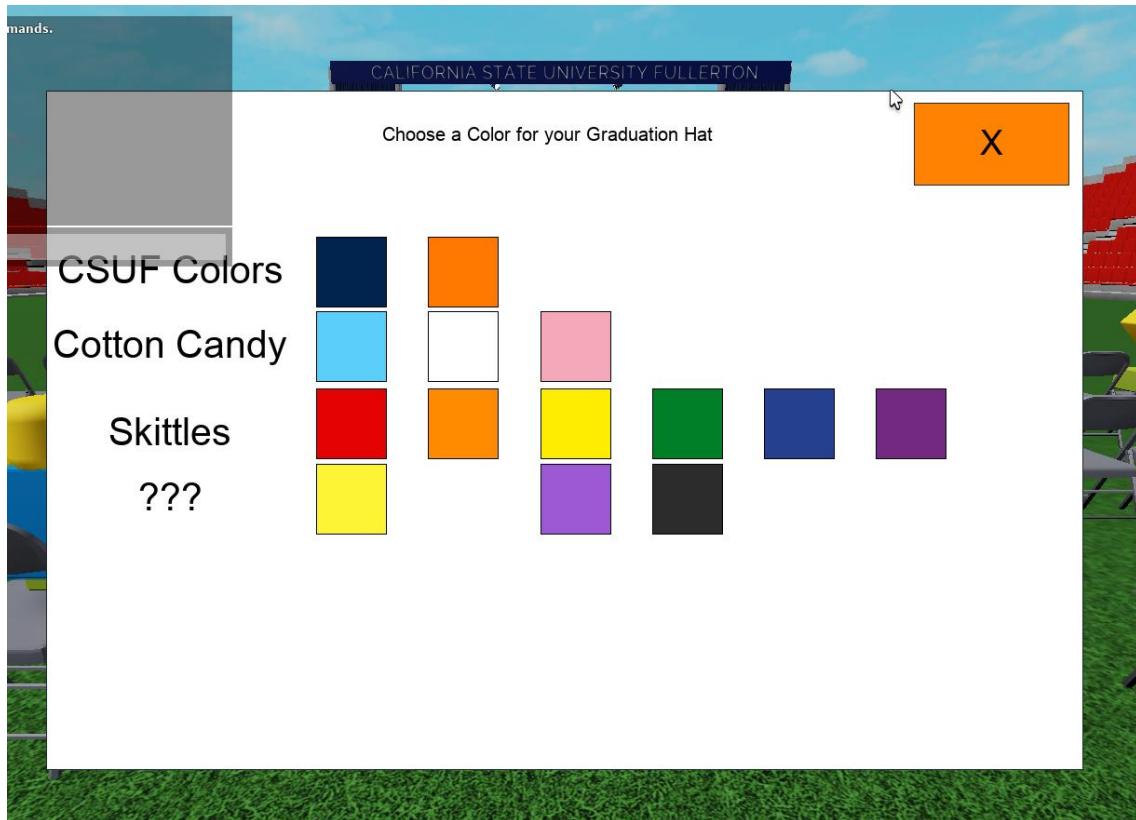
**Deployment and Testing:****Roblox 6: Graduation Cap**

Figure 39 Color Picker for graduation hat that appears on game startup



Figure 40 User model updated with cap and tassel with appropriate color, Hat menu button now visible

**Test Case:** Graduates can chose their cap and gown colors upon entering the game

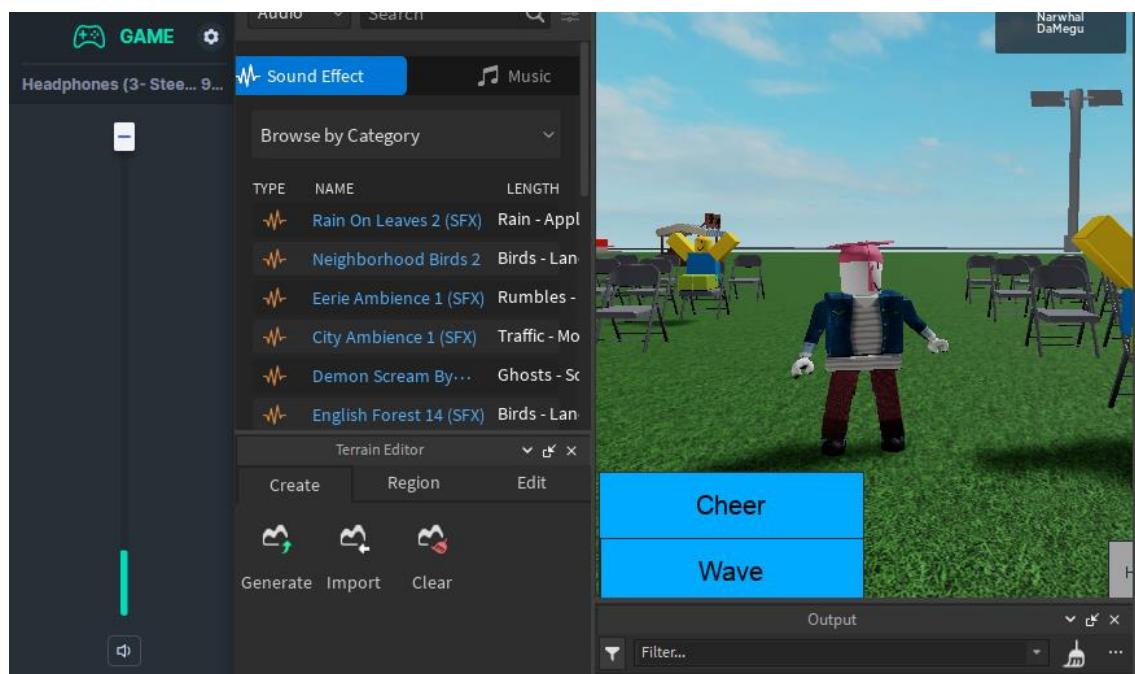
**Result:** User model is updated after user exits color menu

**PASS**

**Test Case:** Graduates can chose their cap and gown colors by clicking the 'Hats' button after closing the color menu

**Result:** Color menu opens after user presses 'Hats' button, button become invisible again.

**PASS**

**Roblox 7: Graduation Music**

**Figure 41 Game audio active(green bar), reflecting that audio is playing on game load.**

**Test Case:** Graduates hear Pomp and Circumstance upon loading into game

**Result:** Pomp and Circumstance plays and loops.

**PASS**

**Roblox 8: Emotes**



Figure 42 The buttons the user can click to perform an emote



Figure 43 The Wave button is pressed and the wave emote is performed



Figure 44 The Cheer button is pressed and the cheer emote is performed

**Test case #1:** Users click on the Cheer button

**Result:** The cheer emote is performed

**PASS**

**Test case#2:** User clicks on the Wave button

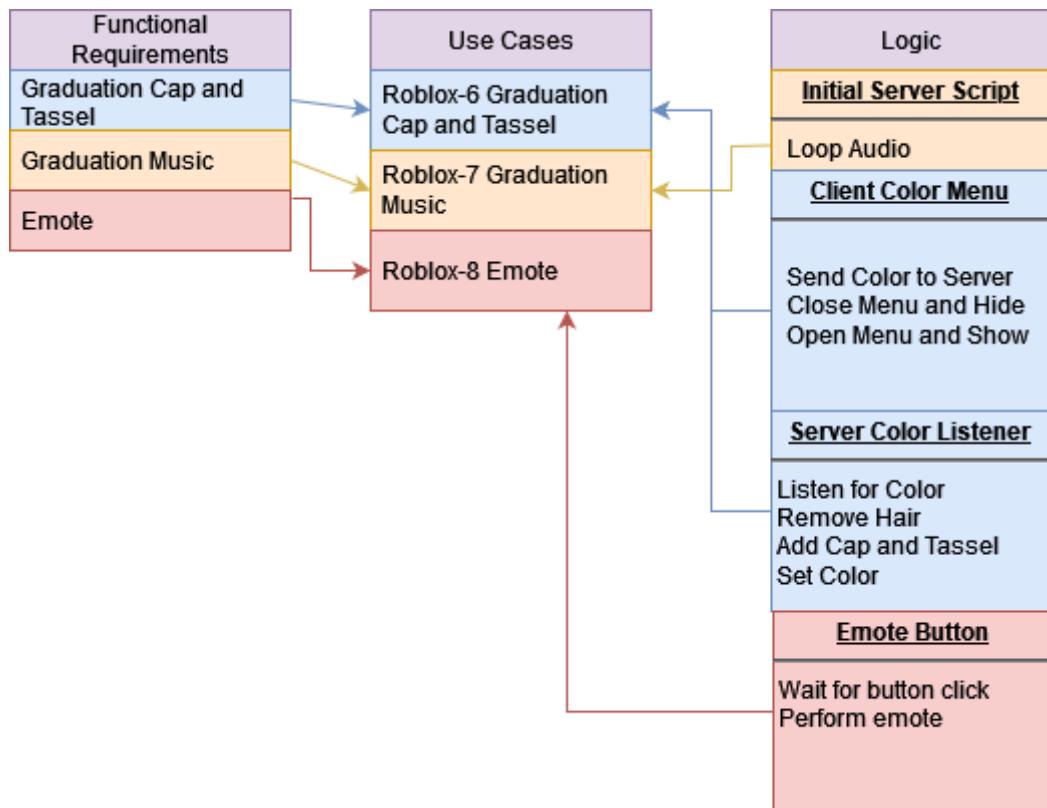
**Result:** The wave emote is performed

**PASS**

**Traceability Matrix**

Requirement Details	User Story	Requirement Name	Diagram ID	Expected Result	Status
Graduates hear March no. 1 - Pomp and Circumstance when they enter the game world	When Kylie enters the game event, he turns on his speakers/headphones. From the audio device, they hear March no. 1 - Pomp and Circumstance and are at peace.	Graduation Music	ROBLOX-7	Audio plays when game loads	OK
Graduates are able to select cap and tassel to wear before the graduation event.	When Ricardo enters Fullerton Commencement from the Roblox Launcher, he is greeted to a screen with colors to choose for their graduation cap and tassel. When he enters the game world, he sees that his roblox avatar is wearing the selected outfit. He also can reselect the cap color at any time.	Graduation Cap	ROBLOX-6	User can change color of cap and tassel, and is able to reenter menu after exiting	OK
When users get their diploma, they can choose a preset choice of emotes to do.	Bob is about to go up to receive her diploma. She shes a window on her screen that show different emotes that she can prepare when she goes up on stage. She clicks an emote, and successfully preforms the selected emote.	Emote	ROBLOX-8	User can click on the “Wave” or “Cheer” button which will activate the corresponding emote	OK

### Traceability Diagram



### Scrum Instead of Agile-UP

If we had chosen Scrum for our Roblox comment event instead of agile-UP, there would have been several notable differences in how we approached the development process. To start, Scrum emphasizes a collaborative, iterative approach to software development. Since Unified Process focuses on a structured and sequential approach to development, Scrum would break the development process down into short, time-boxed sprints. Instead of creating a comprehensive list of requirements upfront as we did in Unified Process, our team would have worked to define a backlog of high-level requirements and features that would drive development throughout the project. We would have prioritized these requirements based on their importance to the overall game, and the team would have worked to deliver the most valuable features first.

During each sprint, the team would have met for a daily stand-up meeting to discuss progress, identify any obstacles or blockers, and plan the work for the day. At the end of each sprint, the team would have delivered potentially shippable increments to the event, with new features and functionality added incrementally throughout the development process. Regarding documentation, Scrum would emphasize less formal artifacts like UML and detailed class diagrams. The focus is on working software and collaboration, although having a shared understanding of the project's goals and requirements is still essential. We would likely have created a product backlog to define the high-level requirements and features, user stories, and acceptance criteria to guide development.

Risk management was an ongoing process throughout development. We applied unit testing throughout our entire development process, with the team identifying potential risks and taking steps to mitigate them during each sprint. The team has been testing to ensure new features and functionality were working correctly. Overall, we would have taken a more collaborative, iterative approach to our development if we had used Scrum instead of the Unified Process. We would focus on delivering working software in short sprints, and we would have reduced the emphasis on documentation and formal artifacts. Using Scrum, we would still have worked to ensure the team understood the project's goals and requirements well.

Scrum may have been more effective for completing functional requirements than Agile-UP. Agile-UP relies on predetermined work products and processes requiring a learning curve. In

contrast, Scrum employs a repeated approach throughout sprints, which may have been advantageous for our team. If we spent less time learning about each iteration and work product, we could have focused more on development. Both Scrum and Agile-UP have essential practices for adopting incremental and iterative development. Agile-UP time-box iterations can vary depending on the phase and iteration, while Scrum recommends a fixed time-box range of 30 days for each iteration. Agile-UP focuses on iterations and work products that are initialized and refined throughout development, whereas Scrum prioritizes daily progress meetings, called stand-ups, and active coding.

Several differences would have affected our workflow if our team had implemented Scrum instead of Agile-UP. We would have had to conduct daily stand-up meetings, focusing on specific progress and problem-solving questions. Workable demos would have been necessary at the end of each iteration, and the ceremony scale would have been smaller, requiring fewer work products. Documentation would only occur in the final phase of Scrum. Additionally, we would have needed to involve customers and clients in our meetings to update them on the project's progress and ensure we met their needs.

Using Scrum, we could have focused more on completing functional requirements during sprints and less on development processes. Daily stand-up meetings would have improved team communication, leading to better retrospective meetings and understanding of the progress. Ultimately, the repeated methods across sprints, stand-up meetings, and lower emphasis on documentation could have been advantageous for our team's development focus.

## Transition

### Deployment and Testing:

We have created a new branch based on our most up to date Bitbucket repository for our Production build. This build has been moved to a Github link and requires a few steps before a person can enter the Roblox game through the standard Roblox site.

### Prerequisite to Build

Prerequisite	Instructions
Must be using a windows computer	Run on windows computer
Project Repository must be downloaded from Github link below	Download Github repository to computer from <a href="https://github.com/ChrisMelgarMorales/RobloxGradFinal">https://github.com/ChrisMelgarMorales/RobloxGradFinal</a>
rojo.exe exist in github repository folder labeled 'rojo'	rojo.exe from the download link listed as <a href="#">rojo-7.2.1-win64.zip</a> . then move the executable from the zip to the folder called rojo in the repository download.
Roblox Studio already installed.	Follow instructions listed for installing Roblox Studio at <a href="https://create.roblox.com/docs/getting-started/setting-up-roblox-studio">https://create.roblox.com/docs/getting-started/setting-up-roblox-studio</a>

### Building and Publishing the Game

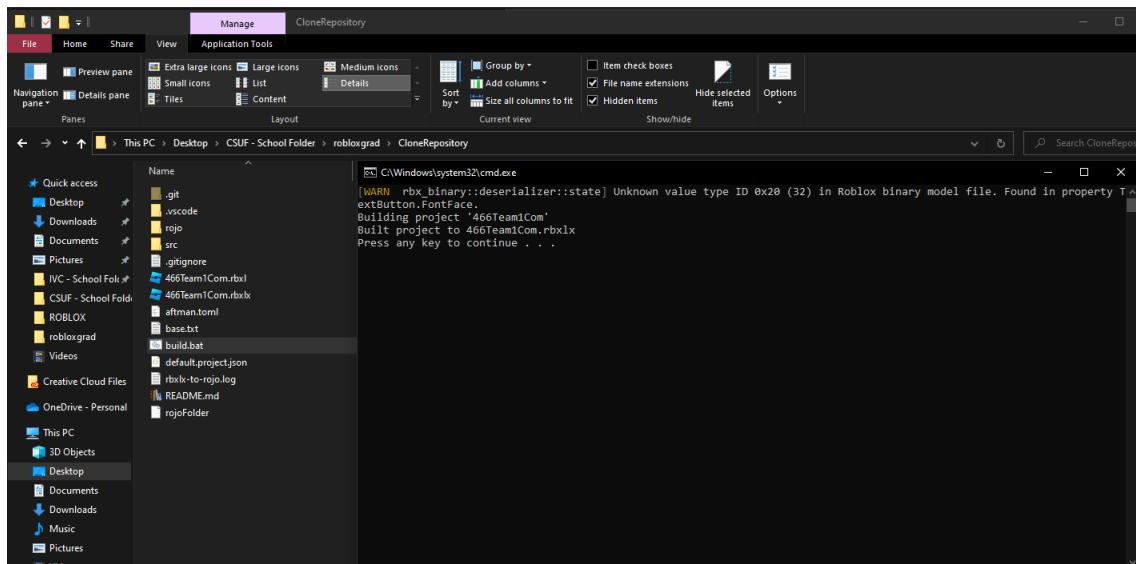
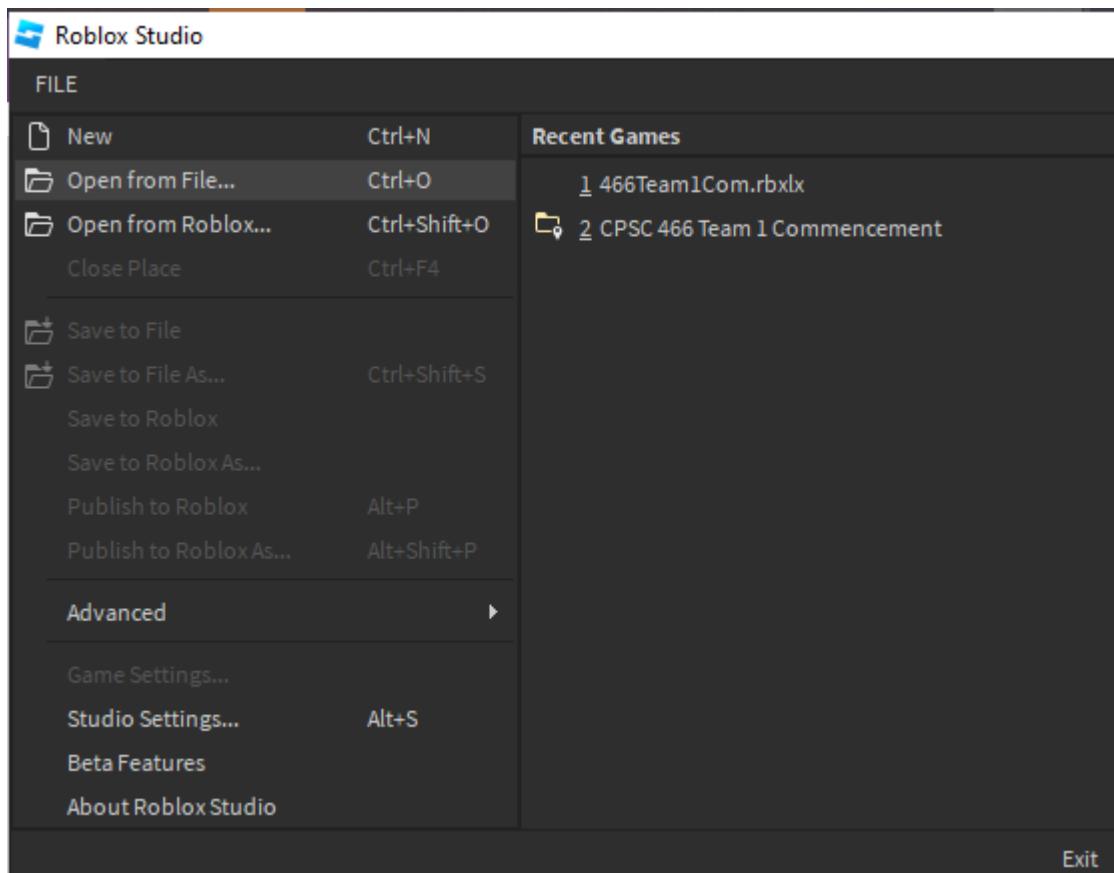
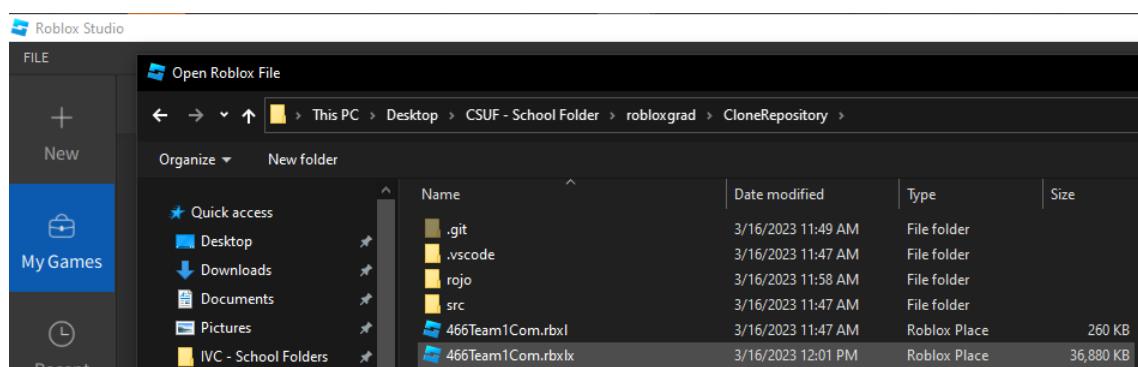


Figure 45 Launch build.bat to generate the Roblox Studio file for the game world.



**Figure 46 Click ‘File’, then click ‘Open from File’ when launching Roblox Studio**



**Figure 47 Select the file generated from rojo.**

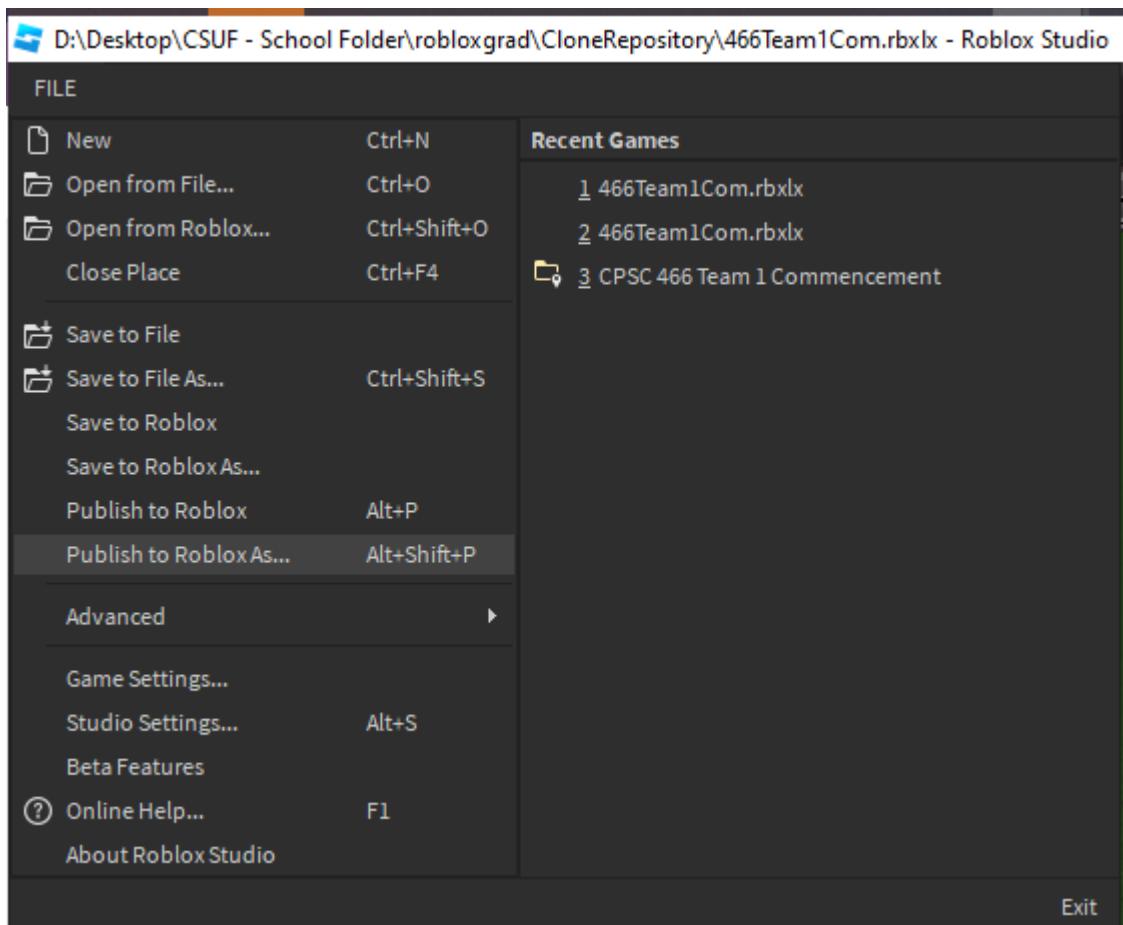


Figure 48 When game opens in Studio, click ‘Publish to Roblox As’

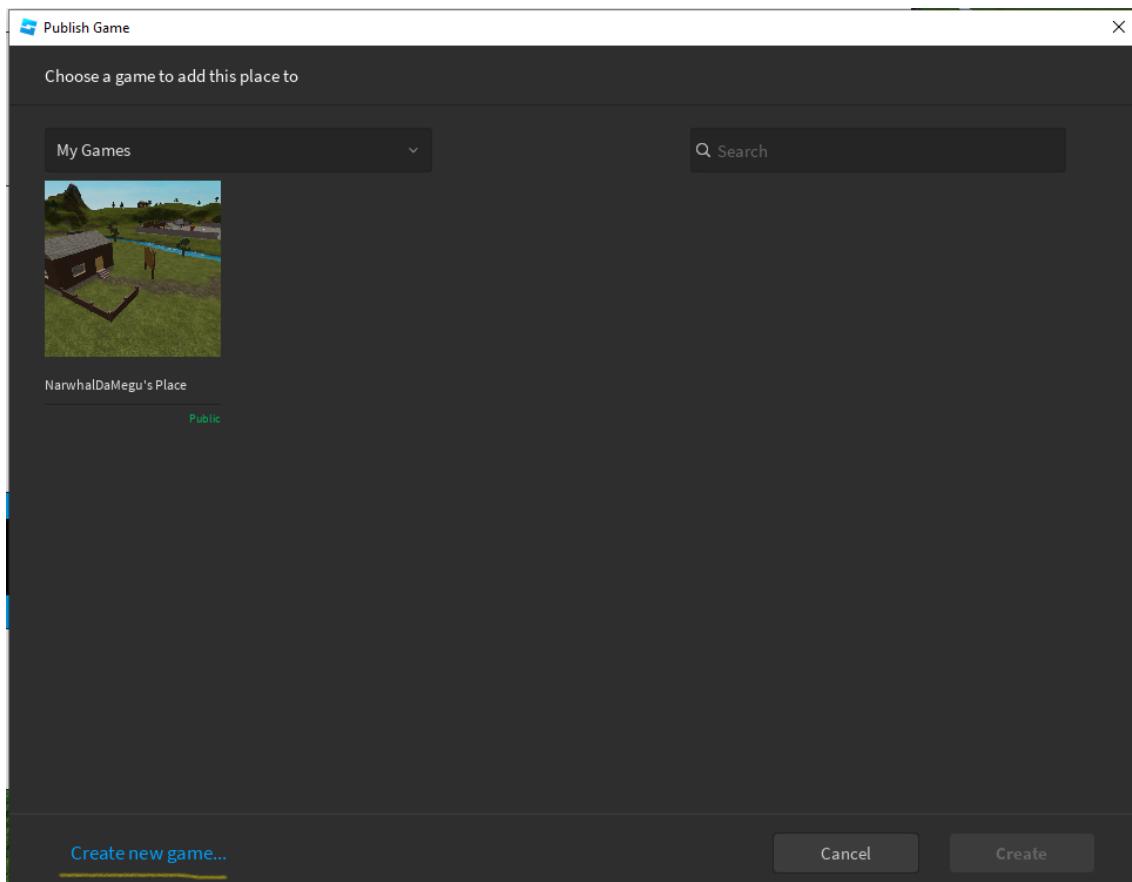


Figure 49 Click 'Create new game...'

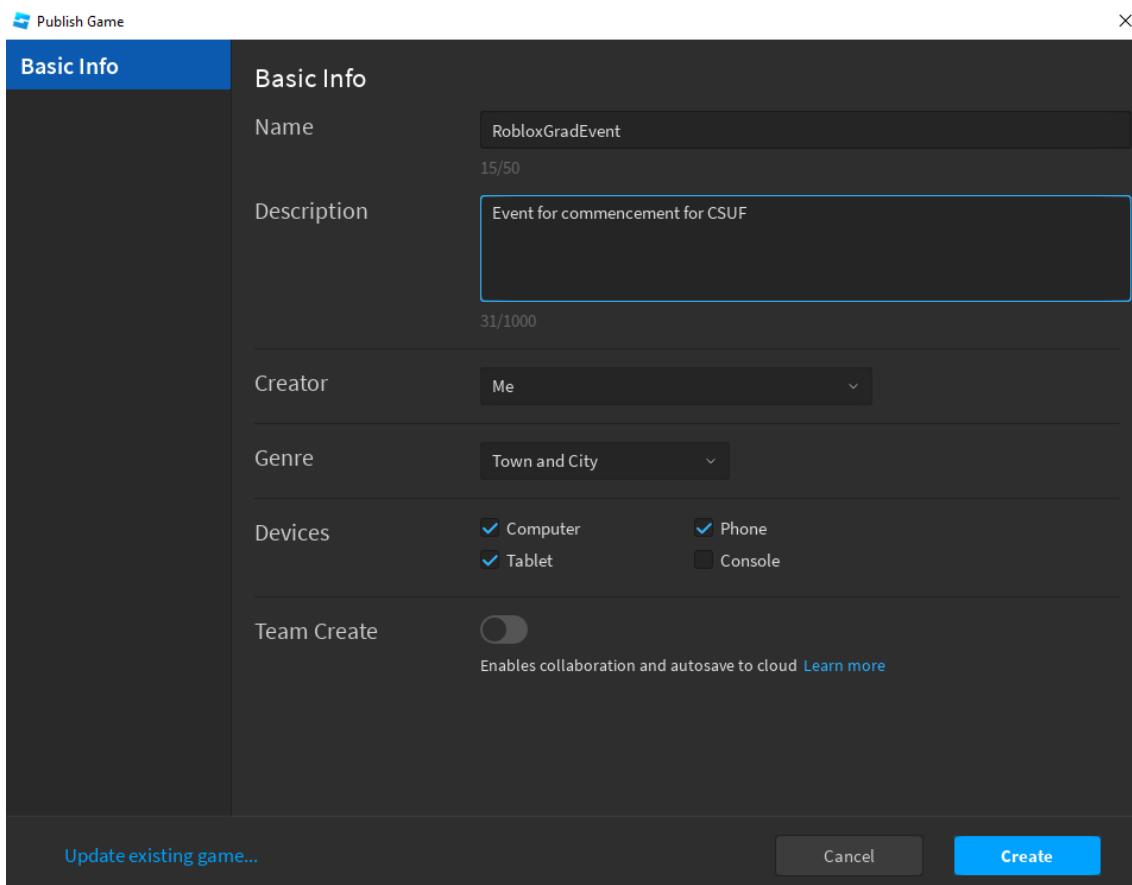
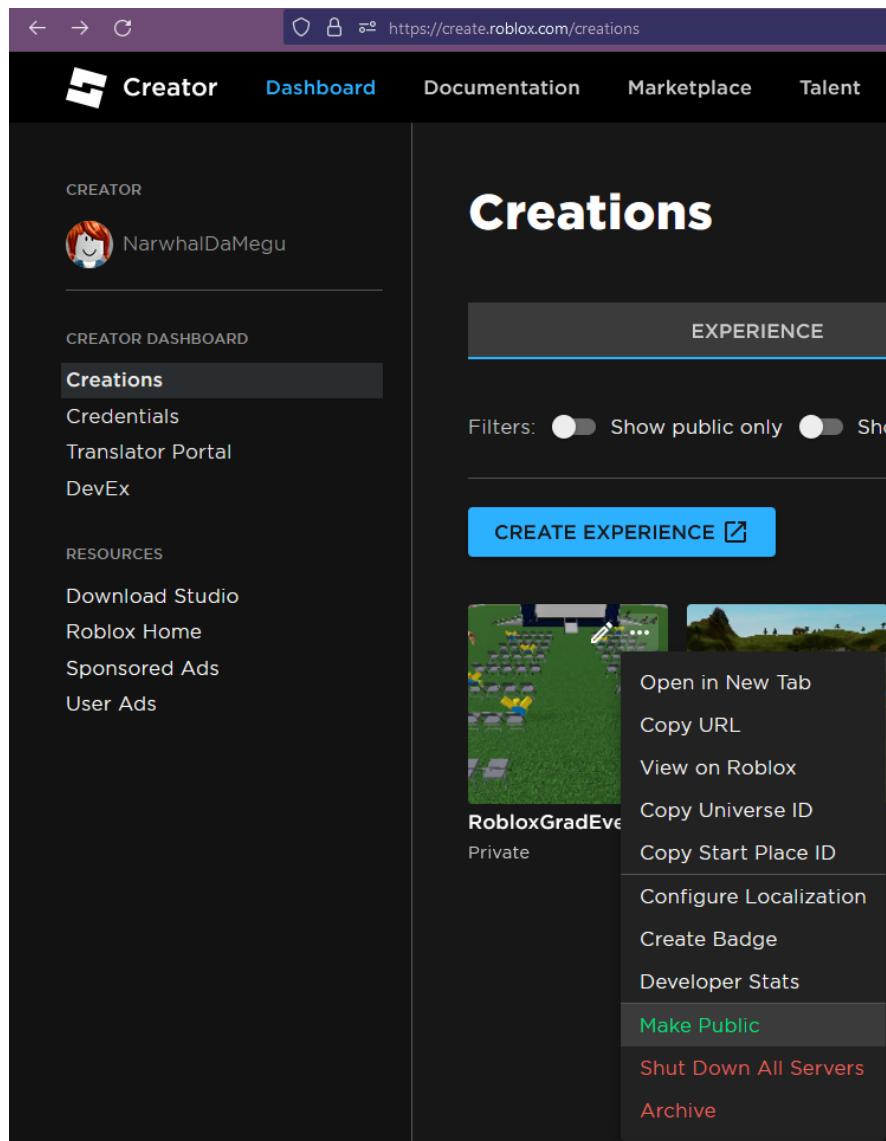


Figure 50 Update the metadata about the game and click create to finalize the game.



**Figure 51** Go to <https://create.roblox.com/creations> and click the three dots on the graduation experience. Then click Make Public

The game can now be shared to CSUF graduates, and they will be able to enter if they are on the whitelist.

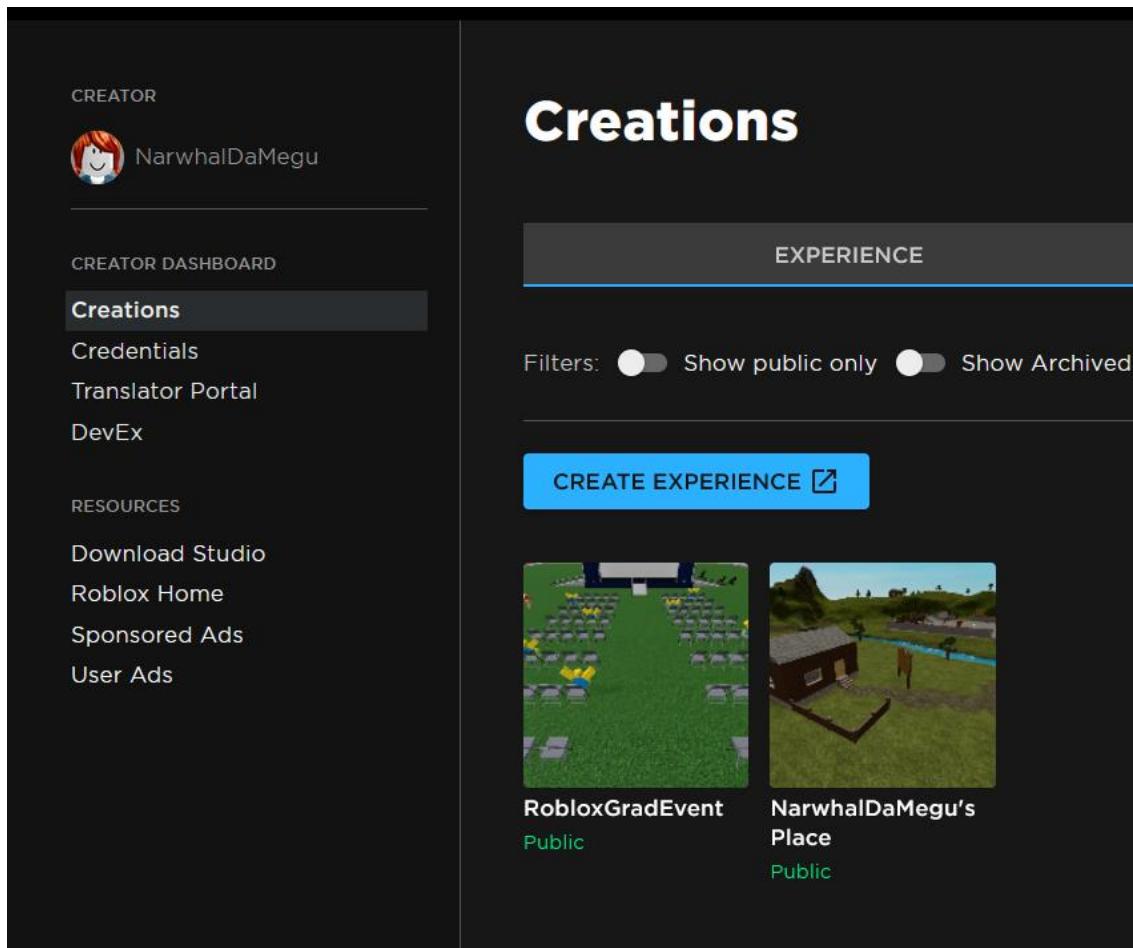
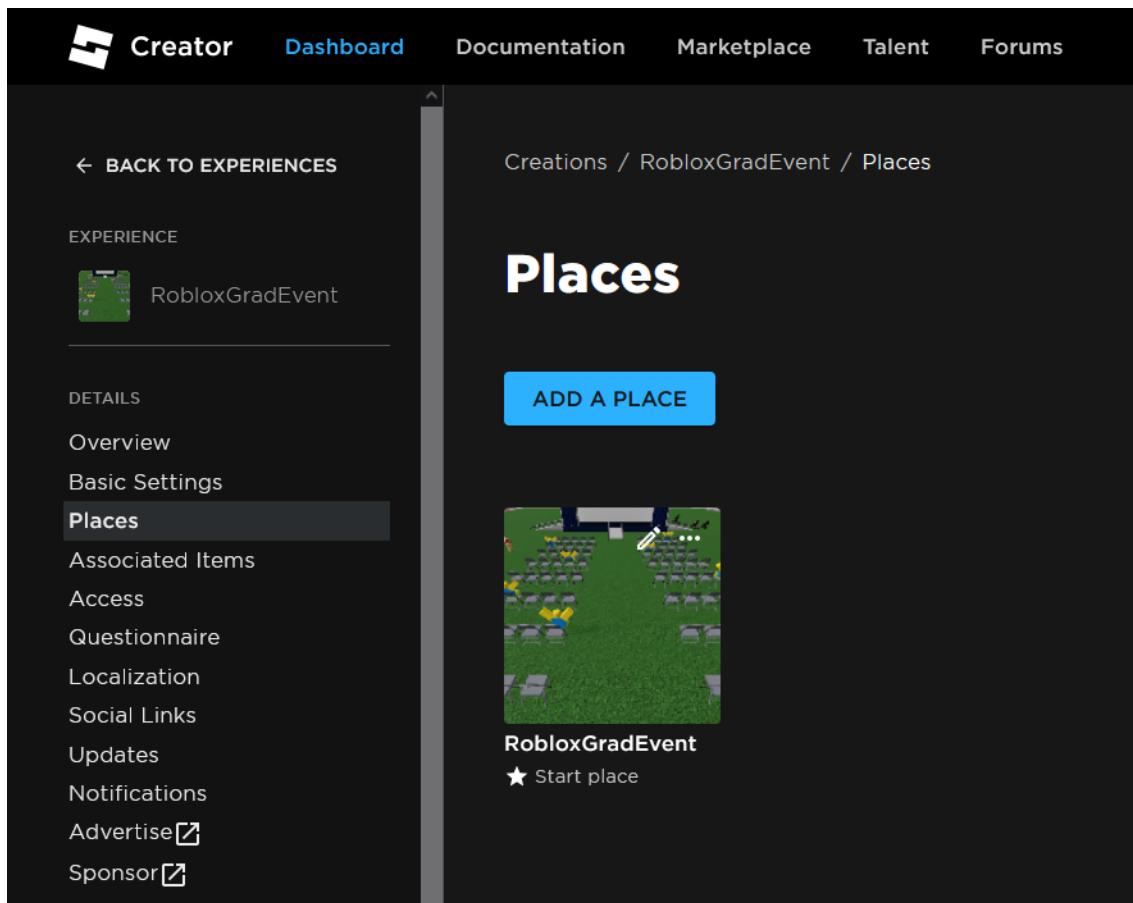
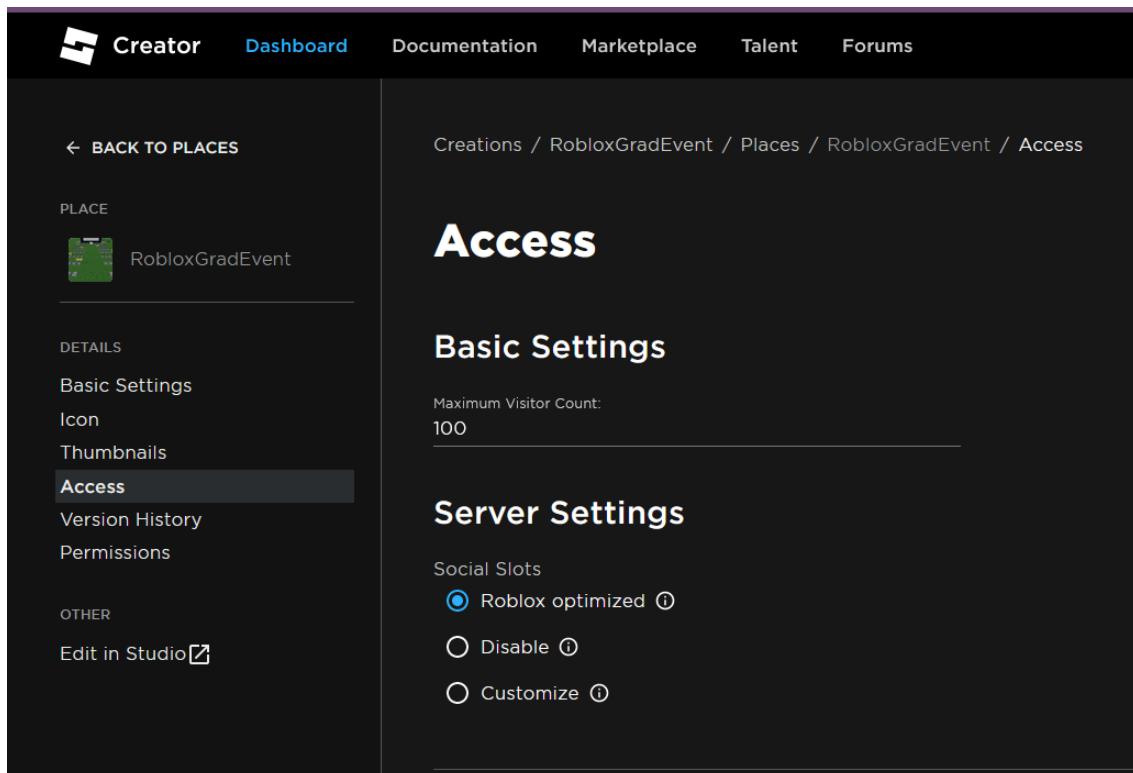
**Increasing Server Size**

Figure 52 Click on the Experience uploaded from Roblox Studio



**Figure 53 Click the Places tab, and click on the place that is shown.**



**Figure 54 Click on the access tab and edit the server size to 100, as that is the max allowed in Roblox as far as we know.**

**Team Evaluation**

<b>Course Title</b>	CPSC 466 Software Process	<i>All team members participated in creating this charter and agreed with its content.</i>
<b>Instructor</b>	Kirti Chaudhary	<b>Date 5/9/2023</b>
<b>Course Dates</b>	1/23-5/18	

**Team Members (Contact Information)**

Name	Address (city, state, country)	Cell	Email
Kris Melgar Morales	Tustin, CA, USA	714-660-3716	<a href="mailto:cmelgarmorales@csu.fullerton.edu">cmelgarmorales@csu.fullerton.edu</a>
Mario Linares	Corona, CA, USA	951-227-8982	<a href="mailto:marioliniaraes@csu.fullerton.edu">marioliniaraes@csu.fullerton.edu</a>
Nicholas Jones	Huntington Beach, CA, USA	714-261-8416	<a href="mailto:nicholasj898@csu.fullerton.edu">nicholasj898@csu.fullerton.edu</a>
Daniel VanDenEykel	Anaheim, CA, USA	714-473-2203	<a href="mailto:d.vandeneykel@csu.fullerton.edu">d.vandeneykel@csu.fullerton.edu</a>

**Team Member Skill Inventory** (Areas individual members can contribute)

Kris Melgar Morales	Docker Desktop and Containers Git and Github using VSCode Python, C++, Java AWS cloud management MySQL and GraphQL
Daniel VanDenEykel	Git and GitHub Branch Management Lua, Python, C++ Firebase/mysql Database Solutions Blender for 3D Modeling
Mario Linares	Git and Github C++, Python, Swift MySQL
Nicholas Jones	Git and Github using VSCode C++, Java, Javascript, Lua

**Team Goals** (Project goals, team process goals, quality goals, etc.)

We have a commencement event that works with automatic pathing and no user control.
Our repositorys remain organized throughout the development cycle
Our final iterations is polished, containing representation of the CSUF commencement as we could afford given the time frame to work on this.
Host of the event is able to understand how to build the game

**Team Roles** (Define roles of each member to achieve goals)

Kris Melgar Morales	Developer Tool Chain Management
Mario Linares	Developer UI Designer
Nicholas Jones	Developer Modeling
Daniel VanDenEykel	Developer Asset Management/Modeling

**Ground Rules** (Meeting schedule/locations, attendance expectations, agenda, assignment completion, communication methods, etc.)

Virtual Meetings every Monday evening: assessment and planning on weekly tasks, debriefing on requirements per sprint, and discussion on possible blocks and solutions.

Instant messaging through Discord: updates on progress for the overall project, questions for how to work with the development environment, and discussion on features to be added.

**Time Commitments/Availability** (Pacific Time)

Kris Melgar Morales	Monday 7
Daniel VanDenEykel	Monday and Wednesday Evenings, Friday Mornings, and Sundays
Nicholas Jones	Monday 7
Mario Linares	Monday 7

**Conflict Management** (What potential conflicts might arise among or between team members during this course? How will team members deal with these and other conflicts?)

Conflict in Desired Requirements	Requirements will be randomly assigned if team cannot come to a compromise.
Conflict on implementation of Requirements	Team will get together to discuss benefits of conflicting opinions on implementation

**Risk Management** (What are potential barriers to achieving these goals?)

Lack of Implementaion	Requirements are detailed and ready to be implemented, but no implementation exist by the final
Source control tools not linked	To manage feature separation through branches and merges, VS code need to be linked through Atlassian. Without synchronization code edits will be hard to manage.
Lack of Time	Project last for about a month and half, meaning their is little time to waste between iterations. So its necessary to stay on top of documentation.

**Team Evaluation Criteria for HW1**

Criteria for Evaluation will be based on completion of assigned Requirements to implement.

While the Unified Process places extreme focus on documentation and work products, having a game with actual features implemented will be prioritized, as documentation can be created in reverse from implementation. However, we still need to have proper design for our project, so we will also track whether members have finished the analysis and design part for their requirements with respect to the Use cases and diagrams.

**Team Evaluation**

Developer	Features Assigned	Analysis and Design Status for Use Cases	Implementations Status	Final Evaluation
Kris	ROBLOX-1 ROBLOX-2	Drawn and Added to Report	Complete and Tested	100%
Mario	ROBLOX-3	Drawn and Added to Report	Complete and Tested	100%
Daniel	ROBLOX-5	Drawn and Added to Report	Complete and Tested	100%
Nicholas	ROBLOX-4	Drawn and Added to Report	Complete and Tested	100%

**Team Evaluation Criteria for HW2**

Criteria for Evaluation will be based on completion of assigned Requirements, work on the project document, or progress on the second part of HW2 focused on the scrum analysis

**Team Evaluation**

Developer	Work Assigned	Status	Final Evaluation
Kris	ROBLOX-6 ROBLOX-7	Drawn and Added to Report Complete and Tested	100%
Mario	ROBLOX-8	Drawn and Added to Report Complete and Tested	100%
Daniel	Misc. Proofreading for Presentation	Review and Edits to Report Complete	100%
Nicholas	Wrote and included “Scrum Instead of Agile-UP” section	Written and Added to Report Complete and Reviewed	100%

### **Works Referenced**

Craig Larman. Agile & Iterative Development: A Manager's Guide. Addison- Wesley, Pearson Education, 2004

Corporation, R. (n.d.). Roblox., from <https://www.roblox.com/home>

Rojo. Rojo RSS. (n.d.). <https://rojo.space/>

The Programming Language Lua. (n.d.), from <https://www.lua.org>