EECS 581 Initial Architecture

Team Number: 21

Team Members: Duy Vu, Ethan Osorio-Cortes, Aidan Bowen, Luke McCumber, Matt LaTendresse

Project Name: VRClassroom

Synopsis: An online virtual reality application that provides a classroom setting for teachers to instruct students.

Project Design

When designing this software we asked ourselves what is a problem that we have and how can VR be applied to that problem and fix it. The idea of having virtual office hours or a virtual classroom was one of the first things that we all thought of. All of us have attended school during the COVID-19 pandemic and seen and participated in an online class. We thought that what if we could create another experience for people going to class besides on Zoom or in person. Thus, the idea of a VR classroom was born. The software is designed to provide an online virtual reality experience that closely resembles that of a school classroom. Desks, chairs, windows, and a whiteboard are all present in an attempt to simulate a class-like structure. Teachers and students can freely roam around or sit down in their respective desks. The users will interact with one another by talking or typing in the text chat. Additionally, the teacher is able to draw on the whiteboard to assist with their instruction. Students can view this and take notes of their own with the option to save them for later usage. For our development purposes, we will use the Oculus Quest 2 headset. However, we could possibly expand this to other VR headsets as well. With all of these in mind, it is our goal to make the experience the best for the student. We want the user to prefer this method over a Zoom call and ultimately allow for more teaching and creative spaces around the world.

One major component of the software is the multiplayer aspect. Upon starting the application, users will be prompted to either start a meeting as a teacher or to join one as a student. Teachers will be spawned into the classroom with an access code. The code can be distributed to other users who can then join the same session as students. Another option is through invites. The teacher can also send an in-game invite to other users who can accept or decline the offer to join. All users will be able to see and interact with one another through voice or text chat. Everyone has the ability to mute themselves, but the teacher has some more features to control the classroom. At their discretion, the teacher can mute the entire room or specific people if need be. Anybody can leave the meeting at any time, but the option to remove users from the meeting is also available. Students and teachers will have their own display name so that people can refer to each other. The option to change their name will be in the settings, along with optional adjustments to other audio and visual needs.

There are many different types of networking frameworks for virtual reality games. Since virtual reality is still relatively new, software is often evolving and the standards are always changing. As a result, a lot of frameworks become deprecated or outdated frequently, which makes it difficult to learn or stick to a structure. Despite that, we found a framework that most virtual reality games utilize today. The main networking framework that we are planning to use is called Photon, more specifically PUN2. This is a software development kit designed specifically for Unity and works for VR multiplayer apps. This allows us to host a server on a cloud network, meaning we do not have to code any server hosting. All we would need is for the user to send an app key to the cloud and they would be able to connect. Along with low latency, optional matchmaking, and lobbies, Photon will allow us to host our VR classroom with ease.

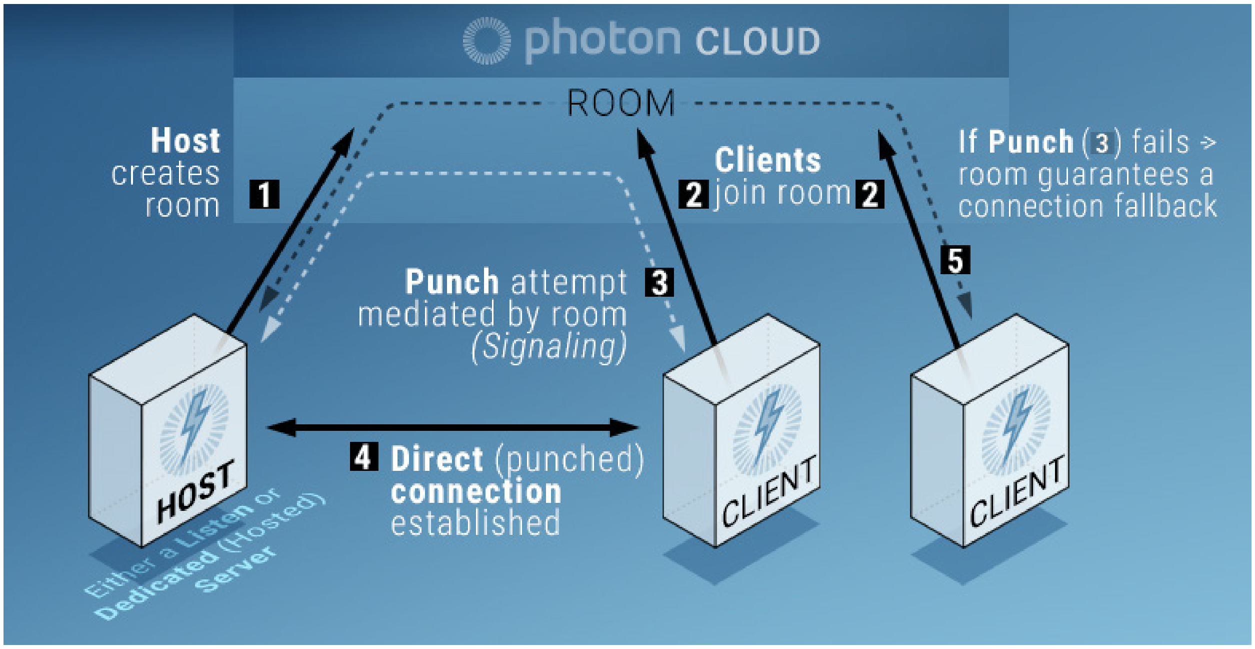

photon CLOUD
ROOM
Host creates room 1
Clients 2 join room 2
If Punch (3) fails > room guarantees a connection fallback
5
Punch attempt 3 mediated by room (Signaling)
4 Direct (punched) connection established
HOST
CLIENT
CLIENT
Either a Listen or Dedicated (Hosted) Server

Figure 1: Chart explaining the Photon Cloud Network of Servers

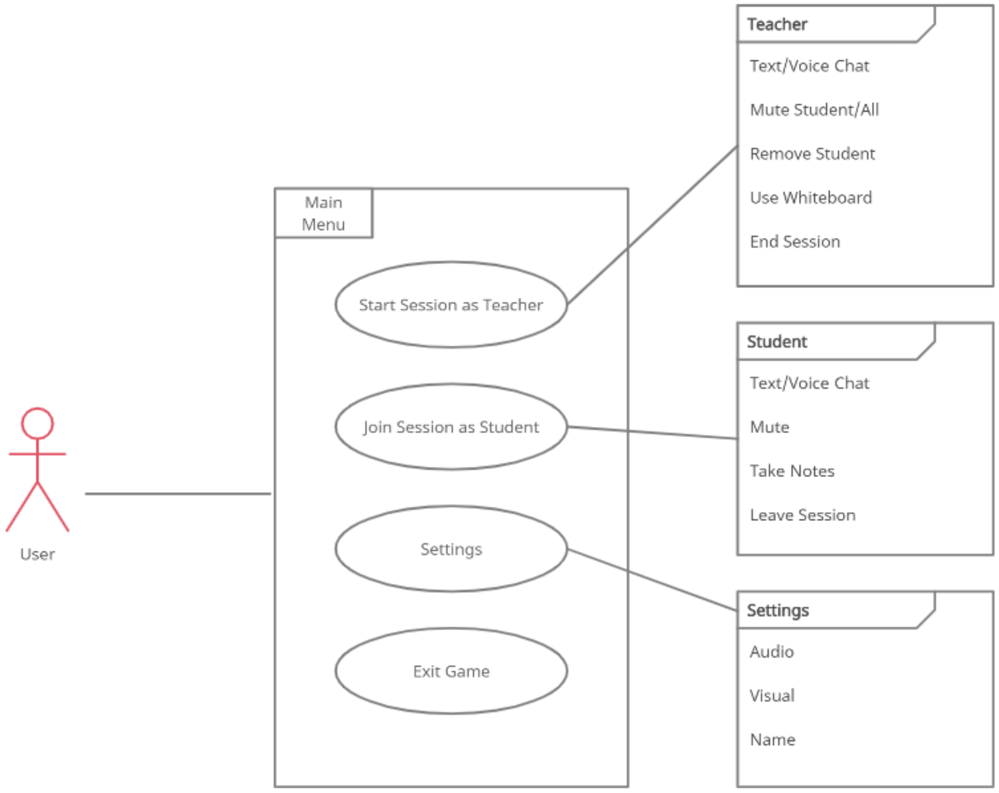
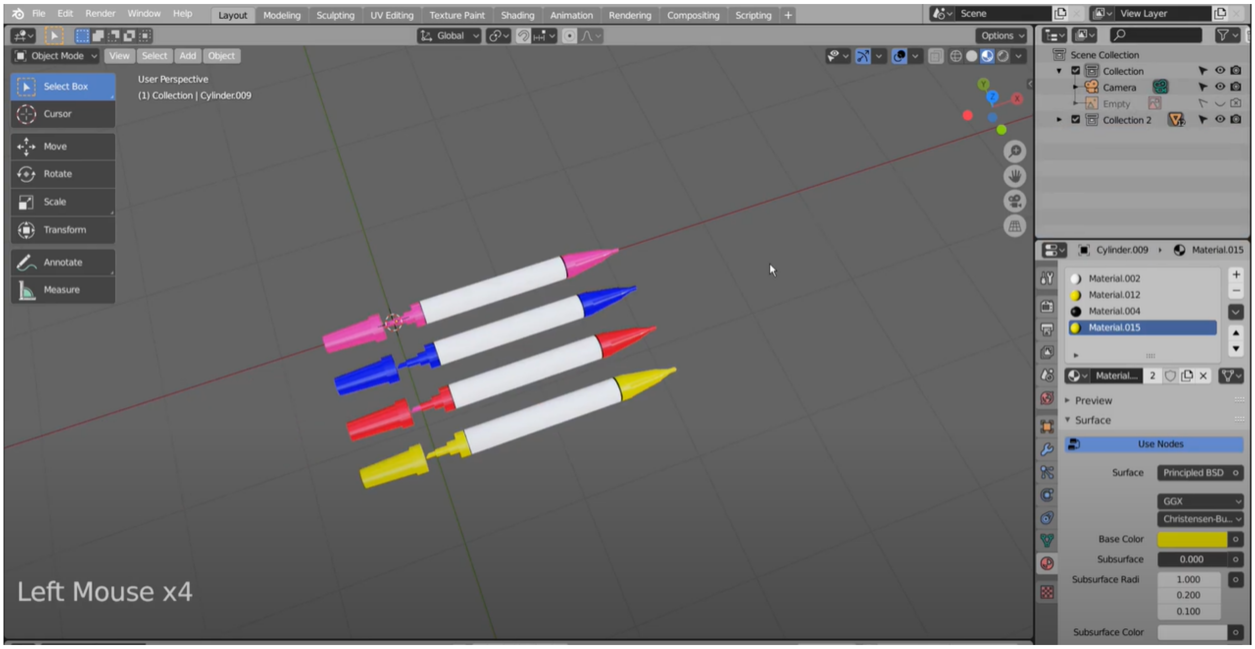


Figure 2: Player Use Case Diagram

For the classroom setting and player interactions, we intend to use Unity, a developmental game engine. The user is the most important thing to us and we want them to have a great experience using our product. We want them to be fully immersed in this experience and be able to attend a lecture or have a conversation with a professor that could possibly last for many hours. With this in mind we chose Unity for the accuracy and realism that it allows us to make when we are creating our virtual environments. Unity is very development friendly and works very well with virtual reality. The engine is separated into objects in the game scene and the C# scripts that come attached with it. With this, we are able to fully customize how the classroom will look. Using Blender, a free 3D modeling software, we can create and import creations of our own, as seen in Figure 3. Blender allows us to create meshes for 3D objects. We can attach different textures to the meshes for different visual effects that would change the look of the object or how light is reflected from the surface. Once we create our models in Blender we can export them and import them into Unity for use in our project. For example, we could make a desk and import it into the setting. Then we can code the scripts to enable users to sit at that desk. The scripts also apply to the interactions that users will have with each other. Players will be able to see their hands and pick up objects such as pencils and markers to write with. Through this, we can also create a user interface or accessible menu in-game. Users can adjust certain settings like their in-game name or volume preferences.



Surface: Principled BSD
GGX
Christensen-Bu...
Base Color
Subsurface: 0.000
Subsurface Radi: 1.000, 0.200, 0.100
Subsurface Color

Figure 3: Example of 3D model of markers using Blender

One example of an important object in our app are the markers and whiteboard. We plan on creating the 3D models through Blender and then importing them into our Unity environment. The markers will be available for all the users inside the server. Each student and teacher can use them to complete the tasks or assignments that they are working on. A variety of fonts and sizes will be made available for customization. Students can take notes on their desk and save their progress while the teacher can instruct by using the whiteboard. Another feature that could be added is the implementation of a desktop screen in the classroom. This would allow the teacher to display their own personal Desktop screen for everyone to see. Online lectures, slideshows, or videos could be displayed for learning.