CMSC 234/334 Mobile Computing Chien, Winter 2017

Assignment #2:
Quickstart to Creating Gear VR Apps with Unity

***Release**: Jan 10, 2017* ***Due**: January 16 and 17, 2017 (depends on signup)*
Checkoff Location: Eckhart 131

How to Complete Assignment: TA's (Gushu Li or Ryan Wu) will be available for assignment "checkoff" January 16 and 17 in Eckhart.
The Assignment and Checkoff are individual, each student should come with their own computer and mobile device, build the game using their computer and devices and then show the VR app on the mobile devices.
Please signup immediately for 10-minute checkout slot -- http://bit.ly/2hW5RU4

The purpose of this assignment is to expose you to 3D VR development with Unity. You will build a simple VR game. You will add audio effects and more user interaction. You ` also need to add a simple server to make this app an "Online" game.

Work through the Roll A Ball Tutorial:
You will find the Roll a Ball tutorial at:
https://unity3d.com/learn/tutorials/projects/roll-ball-tutorial

Assignment:
You will need to apply changes based on the Roll A Ball Tutorial. You can find a lot of different assets from Unity Asset Store. You can use lab 1 assets and scripts.
After watching the tutorial, you will know that Roll A Ball is PC game but we can make it a VR game by simple modification. You have a circle of collectable objects and put your camera at the centre of your circle over the plane. Add a reticle to your camera. When you are looking at a collectable object and tap the touchpad, you pickup the object.
This link may help you: https://developer3.oculus.com/documentation/game-engines/latest/concepts/unity-tutorial-rollaball-intro/

a. Change the collectable object used in the game. You need to use an asset from the asset store.5
b. Add audio effects. Trigger an audio effect every time you pick up a collectable object.
c. Add a "change object"button on the top of the scene. Every time you look at (gazing at, pointer enter) the button, the button will change color. Then when you look at the button and tap the touchpad (on the right side of the Gear VR headset), all the remaining collectable objects should change to another model. (For example, the first time you click on this button, the collectable objects change from cubes to spheres. The second time you click on this button, the objects change from spheres back to cubes.)
d. **Network and Multiplayer.** Now you need to add network interaction to the game. This can be achieved by the network support from Unity. Based on the game you have

developed, you need to adapt it for network. First, you need to go through Network and Multiplayer tutorial (http://docs.unity3d.com/Manual/UNet.html). You will be able to learn basic concepts of developing an online game. Then you can start to develop your own online VR game. The specific requirements are as follow:

i. You should have a game server program running on your laptop or other machine you can use (e.g. linux1.cs.uchicago.edu).

ii. You should have two types of clients. One is VR client on the phone. The other one is PC client running on your laptop. The PC client is just a normal 3D game - the original roll-a-ball game, and you use keyboard to control a ball and pick up objects. You need to add network support to it so that the PC client can join the online game.

iii. When a client is executed, it should connect to the server and show a list of connected devices in the scene. The server will assign a new **color** to this client. (You also need to show the corresponding color for each client on the connected device list.) Every time a new client enters, the connected device list should be updated and broadcast to those which have already connected to the server.

iv. The server can start the game. After the game is started, no more clients can get in. To start a game, the server needs to generate a “map”. For example, we have 2 clients in this game. One color is red and the other one is blue. The server needs to generate a circle of objects with two colors. The number of the objects with red and blue should be equal. Each client can only collect objects of its own color. After the game “map” is generated, it is broadcasted to all clients. The clients should not show the connected device list during the game.

v. All the clients share the same scene. In any VR device or PC screen, the player should see the same things. So that every time one client collects an object, the client needs to report to the server and the server will broadcast this information to all connected clients. When an object is picked up by one client, it should disappear in everyone’s scene.

vi. The first client who collects all its objects wins. When the server knows the winner (when one client picks up its last object), it should let all clients stop the game and show the winner client.

e. **Network Latency and Skew**. As you have learned, skew is a critical problem in virtual reality and in multi-player coordination, Given a multi-player game, coordinated by a server, what are the limits of the Unity-GearVR system in a wifi networked environment for latency? Design an experiment to measure the latency from a change in the server to it being reflected in the client. Or from one client to another. Explain how it measure the latency, and what the results mean for what can be done in terms of multi-player games?

You need to show a demo with the TA in the checkoff. You also need to turn in a 4-page writeup describing your system (interface and code), and any salient design elements. It should also include your latency experiment and measurements. Think about the performance of your system, and what kinds of interaction might be possible between the server and the client.