3D Virtual Environment (5 points)

Submission due by Saturday, February 4th at 11:59 pm CT

Purpose

Learn how to use Unity and the Google VR SDK for Unity to develop a 3D virtual environment for VR

Directions

1. Download Unity LTS Release 2021.3.16f1 (https://unity.com/releases/editor/qa/lts-releases?version=2021.3) and install it. Make sure that when installing the editor version, you also install all of the Android build support options. This will be needed to ensure you switch the build platform to Android.
2. Create a new project and name it "CS6334 + *your net id*"
3. Save the current scene as assignment01.unity.
4. Follow the setup guide for the Google Cardboard SDK: (https://developers.google.com/cardboard/develop/unity/quickstart)
    - Follow steps 1-5 under the section titled "**Import the SDK and create a new project**".
    - Under the section titled "**Configuring Android project settings**", follow all up the steps until you reach the subsection titled "**Build your project**". Do not build your project yet.
5. Add an empty game object into the hierarchy. Rename this object to "Cardboard Startup".
6. Select the newly created object, and in the inspector window, press the "Add Component" button. Search for "Cardboard Startup", and select that option.
7. In the hierarchy, right-click the "Main Camera" object, and select "Create Empty Parent". Rename this newly created parent to be "Player".
8. Select the Main Camera. Add the "Tracked Pose Driver" and "Camera Pointer" components in the inspector tab.
9. Create a plane 3D object. Adjust the position of the Player so that you can see the plane in the emulator.
10. Create 3 cube objects, 3 sphere objects, and 4 distinct realistic virtual objects (e.g., a chair, a table, a car, etc.) around the Player object on the plane. The imported models can be created by you or another source (e.g., Unity Asset Store, TurboSquid.com, CGTrader.com).
11. Click on the play button to simulate a 3D virtual world that you created.

12. Click on File -> Build Settings. Click on “Add Open Scenes”. Now, click on “Build” to generate an APK file. Save it as assignment01.apk. If you have your android device connected, you can click “Build and Run” to run the application directly on the device.

13. After building it, save the scene as assignment01.unity.

Submission

1. Clean up your Unity project by removing any unnecessary assets from the “Assets” folder and deleting the project’s automatically generated “obj”, “Library” and “Temp” folders. Your submission zip file must be **500 MB** or less.

2. Create a “Source” document (.pdf) that provides a unique URL for where you obtained each virtual object within your project. If the virtual objects are created by you, please indicate that you created them by yourself.

3. Create a zip file (.zip) that contains your entire “CS6334+ *your net id*” Unity project folder and your “Source” document. Do NOT use any compression fie types (e.g., .rar, .7z, .tar) other than .zip. Such submissions will NOT be graded, which will result in 0 points.

4. Submit the zip file on eLearning under Assignments > Assignment #1. There will be unlimited attempts to submit your work, and the last submission will be graded.

Scoring

This assignment will be scored as indicated below. The maximum possible score is 5 points.

☐ Your virtual environment contains at least ten virtual objects. **0.5 point per object**

Deductions

Deductions will be applied as indicated below. The minimum possible score is 0 points.

☐ Your virtual environment contains inappropriately scaled/placed or unrealistic virtual objects. **0.5 point per object**

☐ Your submission is late. **1 point per day late**

☐ Your submission is not a .zip file. **5 points**

☐ Your submission is larger than 500 MB. **0.5 point per 50 MB over**

☐ Your Unity project does not properly work during initial grading. **2 points**

☐ Your supplementary files are not of the specified formats or do not contain the specified information. **1 point per file**

☐ You did not follow the specified naming conventions. **0.5 point per file or folder**

☐ An APK file is not included. **1 point**

Regrade Policy

For programming assignments, you have a window of 7 days (from when we return your assignment) to ask for a regrade. We will not consider any regrade requests outside this window. Regrade requests should be emailed to TA; regrade requests will not be considered unless they contain a clear explanation on why a regrade should be issued. The TA will respond to your regrade request within 72 hours of receiving it.

Academic Integrity

This is an individual assignment. Each student is expected to complete his/her own work. If found guilty of academic dishonesty, you will receive 0 points on this assignment. Below is a list of things that are considered as academic dishonesty or not:

Considered Academic Dishonesty:

- If you download and copy an already developed scene from someone else or from the Internet, it will be considered academic dishonesty. Copying the scene and making changes in it is still considered academic dishonesty.
- Sharing your actual program code with other students is considered academic dishonesty. You must not share the actual program code with other students. You should not ask anyone to give you a copy of their code or, conversely, give your code to another student who asks you for it; nor should you post your solutions on the web, in public repositories, or any other publicly accessible place.
- You must not look at solution sets or program code from other years. Looking at solution sets or program codes from other years is a dangerous practice. Most assignments change in a variety of ways from year to year as we seek to make them better.
- Copying scripts directly from other students or the Internet is considered academic dishonesty. Copying the script and making slight changes (e.g., variable names) still considered academic dishonesty.

Not Considered Academic Dishonesty:

- Materials, prefabs, and 2D/3D objects downloaded from Unity Asset store or other sources are allowed to use if they are not from other students (including current and previous semesters). However, the source of the assets should be clearly described in a Source document.
- You can refer to scripting solutions on the Internet, try and understand it and then write

your own scripts. However, the source of the scripting solutions should be clearly described in a Source document.

Every submission will be checked for plagiarism. If found guilty, you will receive 0 points for this assignment without any exceptions, and your case will be reported to the department and/or university for further action.

These descriptions and timelines are subject to change at the discretion of the professor.