SNHU

7-1 Project: Reflections

Jonathan Sussan

Gene Rodriguez M.S, MBA

### CS 330 Comp Graphic and Visualization

12/12/2021

**Justify development choices for your 3D scene.**

When I originally chose my scene, I didn’t realize just how complicated the scene would be for someone new to modern OpenGL. I selected objects that included a plane, spheres, torus, and cylinder. I didn’t have too many issues drawing the plane but got stuck on drawing spheres that were needed for the light bulb, glass around the light bulb, and a container. Do to spending so much time on figuring out spheres, I was not able to figure out how to successfully create a torus and cylinder in time for the project due date. I also did not figure out how to make the glass around the light bulb transparent so that I could see the light bulb inside, so I moved the light bulb out of the glass for the time being until I figure that out.

For textures, I was able to cut out sections of my original scene and create swatches of them to apply to my objects. This worked well, although I did need to darken a few up because they looked brighter once lighting was applied. I included widely used headers such as *camera.h*, *linmath.h*, *stb\_image.h*, etc. I used 2 source files (*source.cpp* and *ShapeGenerator.cpp*) to create the objects. *ShapeGenerator.cpp* has the necessary calculations for the shapes.

**Explain how a user can navigate your 3D scene**.

Although I had difficulties drawing shapes, I was successful in the navigation of my scene. A user can use input devices such as a mouse and keyboard to move around the scene in any direction, as well as zoom in and out. For example, a user can freely move in any direction in the scene by using a mouse and the *A* and *D* keys on a keyboard. Zooming in and out can be done using the *W* and *S* keys on the keyboard.

Another part of the scene where I felt that I did well on was the lighting aspect. I was able to apply phone lighting to the scene, which is a combination of ambient, diffuse, and specular lighting. Ambient lighting is used so that objects are never completely dark. Diffuse lighting is used as a directional light, depending on what angle the camera is at. Specular lighting is used as a bright spot on where the focus is currently at. Some textures were coming out too bright due to lighting, so I did need to adjust the texture jpegs a bit.

**Explain the custom functions in your program that you are using to make your code more modular and organized**.

I ensured my code was modular and organized by separating the functionality of the program into independent modules, where each module contained everything necessary to execute one aspect of the desired functionality. When you divide programs into smaller manageable pieces, it allows you to be able to concentrate on what aspect of the program you want to focus on. This way the main program, or *source.cpp* in my case, can define a function, call a function, and declare a function that are specified in other header or C++ files. On top of properly organizing my program, I also included easy to follow comments so that if another developer were to look at the code, they would easily be able to understand my intentions. Functions are reusable blocks of code that are called with a short command. Rather than having to type out the same code multiple times, I can call that code with a short command which lets me avoid repeating code.

After taking this course, I realize that it is quite difficult to learn modern OpenGL in an eight-week course. Most of the docs and information I found on the web were outdated or related to using glut which was not apart of this course. It is an fascinating subject and one worth looking more into.