**CS 330 Final Project Reflection**

Tyler Primas

Southern New Hampshire University

CS 330: Comp Graphic and Visualization

Professor Phillips

June 15, 2023

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

When I first started thinking of what to include in the scene, I was given the task of finding objects that were primitive shapes. These shapes are a cube, cylinder, plane, pyramid, sphere, and torus. One of the objects was also supposed to be able to be recreated using two of those objects. Originally, I struggled to find an object to recreate with two shapes without it being too difficult, but then I remembered a previous module showed a spoon so I decided I would try that. The objects I ended up with were a wooden spoon, a tennis ball, a 6-sided die, and a 4-sided die. They were all good choices because they were easy to recreate using primitive shapes and I had an interest in the items. I was curious how I could recreate the items I chose in a 3D environment.

I initially was going to texture all of the objects in the scene, but seeing as the requirements called for two, I settled for leaving the planned D4 and D6 without a true texture. I put a placeholder blue texture there just to make it look a little better, but they did not have their originally planned textures. I included two lights as requested with one being a white light and one being a blue light to give a slightly different lighting depending on the area.

In order to program for the required functionality, I had to work with the tutorial code and look up a lot of other code for creating shapes in OpenGL. For example, the tutorial code made it very simple to create a cube and from that I was able to make a pyramid pretty easily. For the more complex shapes like a sphere and half sphere I had to look up some code to make it actually appear properly. Pretty much everything else in the scene, like the lighting and texturing, was done through following the tutorials which was very helpful.

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

The user can navigate the 3D scene using the mouse and keyboard. The inputs directly tie into the camera header file which makes moving the camera around much easier to implement. The keyboard functions like a normal video game using the W, A, S, and D keys to move around. W moves the camera forward, A left, S backward, and D right. This can be combined with the mouse to move in those directions while looking around. The mouse can be pointed in any specific direction which gives the user the option to look and move in any direction. The Q and E keys can also be pressed to move up and down more easily than pointing the camera up or down with the mouse and moving forward. These controls make it easier for the user to get up close to the objects and view them better from different angles. I think it could be a good idea to add something like a controller to navigate the scene in the future. The joysticks of the controller can be very intuitive and smooth for navigating the scene, which could add to the experience by offering more options to the user.

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

Most of the custom functions in the program are for defining the shapes and how they appear within the program. For example, each of the shapes that are displayed have their own functions. Each of those functions draws the shape which is then called within the render function to render it. Then the texture function is called to put a texture onto the shape and display that to the user. The functions for creating shapes could be reused by calling them again, adjusting their positions, and applying other necessary textures. The code is very well organized and commented to make sure anyone that looks at it will have a good understanding of what each function does. The functions themselves are also named in a way to be self-explanatory so they can easily be remembered and called to create the correct shape.