Ricky Gerolaga

CS 330

October 15, 2022

Project Reflection

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

When developing the scene, I chose the basic objects such as the boxes in the foreground and the barrel object represented as another box object behind the boxes. To program for the functionality, I initialized the transforms of each object in the unnamed namespace which simplified the rendering function by a lot.

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

To navigate the 3D scene, the user can press W to go forward, A to go left, S to go back, and D to go right. The user can also go up and down with the Q and E key respectively. To switch between orthographic and perspective view, the user can press P. the camera header file that is free to use had to be manipulated a bit to get the up and down function going, but that was quite simple.

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

Utilizing the multiple different functions allowed the code to stay organized. The UInitialize function initialized glfw and glew, it also created the window. UProcessInput allowed the user to input keys and the program calls the proper movement for the keys. UCreateMesh was extremely useful because it was the base for the 3D shapes that were created. Having only one block of code for a create mesh function meant that I did not have to code the same function five times which would have made the code a lot longer than it needed to be. UCreateTexture created textures when needed. URender was also significant because it was the function that drew all of the shapes. These functions could also be reused in a different program simply because of the OpenGL functionality it provides. For example, UCreateMesh has the necessary data to create a cube, which can be used in a different program exactly the same way to create the mesh that it needs to. It could also be used to create any other shape as well.