Liam Breslin

02/22/2025

CS-330

**Design Decisions**

The objects I selected to represent in my scene were a table, a water bottle, a wooden plate, and two half cut loafs of bread. The selected objects were intended to represent a scene of things you may see in a kitchen. In order to make everything look cohesive I aligned things using the x, y, and z coordinates so everything fell into frame on top of the plate, white of course was on the table itself. To illustrate the wooden plate I allocated a wooden texture to it, while having the half-loaves of bread feature a typical bread texture as well. I left the water bottle and table with a standard color, instead of a texture since I think it made the scene more believable as mono colored alternatives are commonplace with those items. I did however raise the shininess of each object so the diffuse light from the directional light source that was implemented bounced off everything better. The scene overall has darker colors so in order to allow for clear visibility the material shininess was raised higher than intended. This along with an orange undertone in the lighting I think showed a nice afternoon lighting affect that didn’t disturb the visibility too much.

When viewing the 3D scene the user will be able to navigate independently using their W, A, S, D keys to move left-right and forward-backward. Additionally utilizing the Q and E keys will allow the camera to go directly up or down in the y-axis. The last bit of user navigating that is possible while in the 3D scene is shifting between 2D and 3D viewing perspectives. There are four of them in total and can be shifted between using the number 1, 2, 3, and 4 on your keyboard. Finally if the user wants to change the sensitivity they are using in order to traverse the scene they can do so by using the scroll wheel on a mouse. This sensitivity change will only affect the user's mouse orientation of the camera, and not the speed of travel using W, A, S, D, etc.

There are a plethora of functions used to bring this project to life. I’ll name three different types that were most useful to the project's efficiency. First starting with the RenderScene() method. Here is where all the 3D shapes used to create the objects come to life. Here several different shapes used such as: Box, Half-Sphere, Torus, Tapered Cylinder, etc were used to create the varying objects seen in the scene. Here the shapes scale, position, angle, color, material, and texture can be established. Between each shape there is a common layout so when making other shapes it’s just a matter of plugging in the proper information into the pre established code. The 3D shapes themselves that the function is drawing from are stored in ShapesMesh.cpp. Next is the SetupSceneLights() method. This method is where the different light sources are set up for our scene. The light sources used for our project was directional lighting and a spotlight. Here the color, ambient, diffuse, specular, range, beam size, and falloff can all be adjusted to our needs. Finally, the DefineObjectMaterial() method is where the configuration of different materials of each object to signify how it interacts with the light being pushed on it. You can control the diffuse and specular color, as well as the shininess of each object. After an object material is made with its intended information a custom tag can be placed on the object material and later attached to a shape/object that's being formed in the RenderScene(). For this project 4 object materials were made in wood, hardwood, plastic, and bread.