Justin Faircloth

CS-330

6/22/2025

Final Project Reflection

**Development Choices**

My scene includes a monitor, a stack of books, a lamp, a keyboard, and a mouse, all sitting on top of a desk.. For the monitor, I chose to use 4 boxes lengthened into rectangles to represent the monitor base, stem, screen, and the border of the screen. On the rectangle representing the screen, I put a texture depicting the homepage of Google. I also put a light emanating from the screen to simulate how it lights up the items in front of it. The desk was a simple plane with a texture applied.

The lamp is sitting on a stack of books to the left of the monitor. To create the stack of books, I simply made one large box. To apply the texture of the books, I made a plane with a picture of the spines of a trilogy of books and put it against the front side of the box like a sticker. I did the same for the top of the box, showing the cover of the top book. I then applied a paper texture to the box itself, showing at the sides and back. Doing it that way was easier than making three separate boxes, sizing them, placing them, and applying the textures of each book to them individually.

The lamp was the most complex item I created, using two spheres, a cylinder, and a taper cylinder for the stem of the lamp and an upside-down tapered cylinder for the lampshade. I placed a light at the lampshade and had it pointing down, simulating the light illuminating the books and desk below it.

Finally, I created the keyboard with a box lengthened into a rectangle and applied a picture of my own keyboard to it. I then used a cylinder and a sphere to approximate the shape of my mouse.

**Navigation**

To navigate the scene, I applied functionality that uses WASD to go forward, left, backwards, and right, respectively. I then used Q as the button to raise the camera and E to lower the camera. The mouse itself can be used to pan the camera around. Finally, the mouse wheel can be used to quicken or slow the movement of the camera.

**Custom Functions** Great examples of modularity within my project are the methods SetTransformations(), SetShaderTexture(), SetShaderMaterial(), and others of that nature. These methods are created separately and only need to be called and handed simple values to create, size, place, and texture shapes. All that needs to be done in the main code is to call a few methods and the object appears in the scene. All the heavy lifting and unwieldy code is done when the method is first described and it is left out of the main code. They greatly increase both the speed of the code and the readability and these methods can be easily used wherever needed.