**CS-330 Module 7**

**Final Project Reflection**

**Christopher Clark**

**August 14, 2022**

For this class I chose to model a candlestick holder, a wooden box, a candle, and a lightbulb sitting on a table. I later realized that the requirements only called for four primitive shapes. I chose to eliminate one of the shapes to meet the requirements and reduce complexity. I chose the objects without the knowledge of what it would take to model them. The candlestick holder seemed complex, but I knew it would take more than one shape to model and had to meet that requirement. The box was a simple cube and seemed not to be a difficult item to model. I wanted to include a cylinder and sphere, so I went with the candle and lightbulb. I realized that the table would be considered one of the shapes as it is a plane. I have learned that items I can model using vertices seem to be easier to bring to life than shapes that require additional .cpp and .h files. I had issues rendering a cylinder and a sphere. I went through the tutorials and examples provided but found that with what seems to be multiple ways to accomplish the same task, I could not render these shapes within the scope of the code I had already generated.

As a requirement of the project, we needed to have camera controls to manipulate the scene. I have provided those controls withing the project. The user can navigate the scene using the mouse to move through and view the image from different angles. I have employed the W, A, S, and D keys to allow more navigation for forward, backward, left, and right. Thew Q and E keys provide Up and DOWN movement withing the scene. The O and P keys are set up to enable ortho. They can change the perspective as required. The SPACE BAR will return the image to zero. It will recenter the image to its original location. The mouse buttons will generate messages in the background denoting which button was pressed. This can be used for future upgrades to the system. The scroll wheel will allow for zoom in and out. However, the zoom out only goes as far as the original position of the scene.

The code employed within this project was developed based on the tutorials we were given in the class. I have managed to customize the vertices and develop individual shapes that can be manipulated and textured independently. Each shape has its own VAO and VBO as well as texture files. This is demonstrated when run. Some of the code provided did not account for camera movements. So additional code was added to provide for keyboard and mouse functionality. During development I had to customize the camera.h file to allow for the additional keyboard functionality. The features for up and down movement were not incorporated. Code was added for additional shapes but was not used and therefore removed. As it sits, this project meets most of the requirements and has been developed to the best of my ability.