**CS-330 Project 1: Design Decisions**

**Jason Farrell**

**Southern New Hampshire University**

I selected simple objects that I could render in the space that best represent the objects that I took a picture of. Unfortunately, my vertices mapping wasn’t as strong as I would like and was unable to get complex objects to get to render correctly as I would have liked. However, I do feel I learned a lot of the various OpenGL libraries and various APIs used for projects like this. Using the Camera header from LearnOpenGL really helped and I was able to implement various movement techniques like a 3D game would have. I implemented various functions that would create the models using normals (for lighting) and texture coordinates of each object that I wished to create. I then was able to create textures for each object and lastly apply lighting to the entire scene which used the normals of the objects to reflect light in different directions. The scene I was trying to portray was from an outside picture with the sun overhead. I used a simple plane to shine light down on the objects and the plane that represented the concrete ground.

The user can navigate with simple W,A,S,D keyboard controls as well as vertical controls with Q and E. I also implemented the orthogonal and perspective controls using the P key which the user can switch back and forth between the two modes. I feel really great about the camera movement implementation, however sometimes I had to full screen the application to get the true movement implementing the mouse.

I used a very modular functional programming pattern throughout the entire program which could easily be transitioned more to classes and methods more with a C++ style. I had functions for creating all the individual models, Initializing the application as well as destruction of the models and textures used throughout the application. I kept all my input processing to various functions based on the device used for paring the input device. I also had a render function that the sole purpose was of rendering the models, textures and lighting of each used throughout the scene as well as providing any scale, rotation or translation of the models rendered throughout the scene. Each of these functions could be reusable in other applications due to each being modular based on a specific functional programming aspect of each feature of the application.