CS-465-J7291 Full Stack Development I 23EW6

7-1 Submit Your Project

Jacob Simmons, 13 Aug 2023

**Justify development choices for your 3D scene. As you write, think about why you chose your selected objects. Also consider how you were able to program for the required functionality.**

The elements that I chose were the fish tank, desk mat, glass of water, and post it notes. These four objects met the criteria of the assignments and allowed for some creative freedom throughout. The complex object was the fish tank as I had to have a sphere and a cylinder work together to form the stable base for the fish tank. These objects were able to be deployed using the simple shapes as available in the supplied OpenGL sample code. Learning translation, VAO objects, Binding methodologies, camera controls, lighting, and textures allowed me to complete the project as needed.

**Explain how a user can navigate your 3D scene. As you compose your thoughts, discuss how you set up to control the virtual camera for your 3D scene using different input devices.**

The navigation of the interface is simple. Both the mouse and keyboard are used. The mouse points the camera and then the WASD keys are used to move in those directions. The QE keys are also used to either translate vertically up or down. The mouse scroll is also used to increase to decrease speed. Also, the 2D scene is viewed when pressing the "P" key. The escape key is also used to exit the program. This is very intuitive and simple to use once the user is able to experience this in action.

**Explain the custom functions in your program that you are using to make your code more modular and organized. Ask yourself, what does the function you developed do and how is it reusable?**

I used mostly the code as supplied in the OpenGL sample code. This had a great framework that efficiently used the needed libraries and directories to ensure the critical command functions were employed. I had to use several camera positions changes (additive or subtractive) for the QE key translations along with multiple texture and VAO objects throughout the program to obtain the necessary functionality.

I was able to capitalize on the existing framework to simplify the needed requests for QE functionality with six lines of code. There was code in this that was note used (some shader details, and VAO objects) that I did not use as they were not necessary for this application.

**References**

None