**Project Design Decisions**

The first initial object I added to the 3d scene was the base level bottom plane which would serve as the table flooring. I decided to select and use a wooden floor texture to accurately represent my table scene which was also made out of wood. The second object I created for my scene was the box which is made up of five separate planes attached together. The frosted glass texture for the box was selected to give the object a realistic and reflective tone with specular lighting also reflecting off the side of the box. The third object I added to my scene was the 3ds game console which is made up of two parts representing the two screen panels and buttons. The most challenging aspect to creating this 3ds object was transforming and adjusting the UV textures to properly represent the clamshell design of the console and having the top panel link up at the right spot. The fourth object which I decided to add into my scene was the pyramid object. After playing around with triangles and shapes in my scene from a previous module’s assignment, I felt that adding a pyramid to my scene would make it much more interesting. The geometric blue pattern was also added to the pyramid object which I really like as it blends a futuristic style with the traditional pyramid design.

The final object I added to my scene was the cylinder object which I had decided to add as it resembled the cleaning container from my original pictures. Creating and implementing the cylinder required the most time as I had to consider how to efficiently recreate the object without using a large number of triangles and vertices. In the end I was able to design a cylinder layout which does faithfully capture the original image.

The controls for my 3d scene allow the user to dynamically adjust the virtual camera position providing a greater level of immersion within the scene. The user is able to move forward, backward, left, and right by pressing the WASD keys on the keyboard. The user can also easily move up and down the scene using the Q and E keys. The user’s mouse is responsible for changing the camera orientation to look up and down. The mouse scroll wheel can also be used to increase the camera movement in the scene and extend the cameras position. The user is also able to rapidly switch between perspectives in the scene by pressing the P key. By pressing this key, the scene will cycle back and forth between a 2d orthographic and 3d perspective. In the future, additional input devices such as a gamepad or joystick could be added to this 3d scene in addition to the standard keyboard and mouse interface. The GLFW framework provides support for up to sixteen joysticks and can be invoked by using the glfwJoystickPresent method.

While developing this 3d scene I tried to put an emphasis on clean and organized code which could be expanded upon for further modularization if necessary. I created the camera header to contain all of the logic and functionality for the virtual camera within the scene. The camera class contains several different attributes for controlling the movement, positioning, and mouse sensitivity. I also added the stb\_image header to my project to enable proper image loading for all of my textures within my scene. The UCreateTexture function allows a texture to be loaded and generated into the scene with the proper texture wrapping and filtering parameters. The flipImageVertically function is also called before the texture is applied as OpenGL axis needs images to be flipped. I also added separate lighting for my scene which could be positioned and adjusted to highlight objects across the scene. Overall, I was able to significantly expand my own ability and comfort level working with 3d objects in OpenGL and will definitely be using this knowledge for my future career in the world of software development.

**References**

Tutorial, G. L. F. W. (2019). *GLFW tutorial Event processing*. GLFW. <https://www.glfw.org/docs/3.3/input_guide.html#joystick>