Janai Williams

CS330: Comp Graphic and Visualization

Professor Omar Aaziz

June 22, 2025

Final Project Reflection

For my 3D scene, I chose several types of buildings, a café, a hotel, a check-in building, and a police station, to give the scene variety and make it feel like a real place. I used basic shapes (boxes, prisms, spheres, cylinders) for the building structures, which made the code simple and easy to manage. I added roofs with prisms and domes to make the buildings look more realistic, since real buildings usually have more than just flat tops. Stairs, walkways, and railings were included to show how people would move around the scene and to break up empty spaces. I put lamp posts around the scene so the lighting would look interesting at night, with warm glows in certain spots. The large moon in the sky not only explains why some areas are brighter at night but also adds atmosphere. Each of these objects was programmed using loops and reusable functions so that adding or adjusting objects is quick and not repetitive. For example, I created many stairs with just a loop, instead of copying and pasting code for each one.

Users can explore the scene using both the mouse and the keyboard. The mouse lets you rotate the camera around, looking in different directions. The scroll wheel allows you to zoom in and out, changing the speed and how close or far away you are from the scene. The keyboard (WASD keys) lets you move the camera forward, backward, left, or right. This way, anyone can move through the environment and look at the scene from any angle. There are also two view types, perspective and orthographic, that the user can switch between for different visual effects.

I used a mix of lighting types to make my scene look like nighttime, just after sunset. I added a directional light to act like soft moonlight. This gives everything a faint blue tint and soft shadows, making the scene feel cool and calm. I placed several point lights in the scene to act as lamp posts. Each lamp post has its own light source, which makes bright spots on the ground and on buildings nearby. I set their color to be cool and whiteish, like real streetlights. I gave the moon its own point light and a glow to make it more visible and help explain why some parts of the scene are brighter, even at night. For each light, I set values for how strong they are (ambient, diffuse, specular) and how far their light spreads (attenuation), so the lighting looks realistic and fades naturally. This combination of moonlight and lamp post lights helps guide the viewer’s eye and makes the scene feel more like a real place at night.

I also used these custom functions to keep my code organized and easy to change.

The **SetShaderColor()** function lets me set an object’s color by just changing four numbers.

The **SetShaderTexture()** function assigns a texture to an object so I don’t need to write texture code over and over.

The **SetShaderMaterial()** function lets me give each object material properties like shininess and reflectiveness.

The **SetTransformations()** function takes care of moving, rotating, and scaling any object so I can put things exactly where I want them.

I also use drawing functions like **DrawBoxMesh()** or **DrawSphereMesh()** to create each shape.

These functions make the code much easier to read and reuse. I can quickly add or change objects by calling these functions instead of rewriting code.