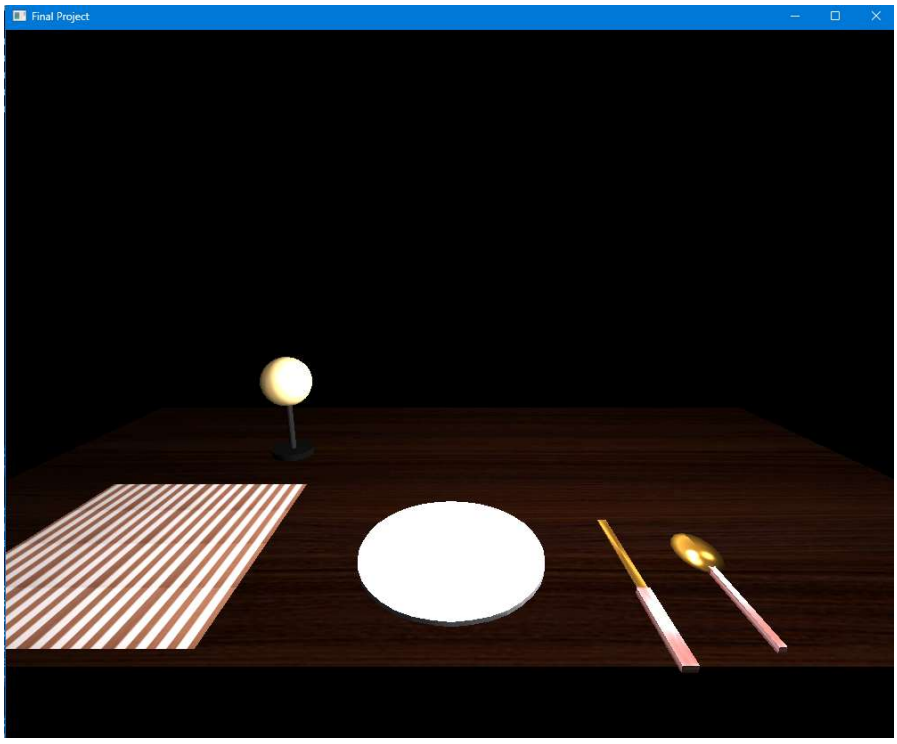
Reflection



For this project, I chose to create a simple table scene. The original reference photo consisted of a napkin, plate, knife, fork, spoon, and water glass. My main struggle with this project was the fork and wine glass. I struggled with beveling the inside of the cylinder and adding a realistic water effect, as well as with the fork, connecting all elements in a manner that appeared realistic. Because of this, I opted to replace both objects with a lamp to still add complexity to the scene while maintaining a realistic look and depiction.

For the table top and napkin, I opted to use a simple plane mesh, as I wanted to make the scene as simple as possible. For the napkin, I used the line m_pShaderManager->setVec2Value("UVscale", 4.0f, 4.0f); to tile the texture 4x4, giving it the appearance of a folded and recently opened napkin. I also searched for a texture image that was striped but somewhat rough. For the plate, I kept it simple

by flattening a cylinder. I considered using a plane, but the cylinder gave it a more 3D look, and the plane had shadows that did not accurately match the lighting of the scene. I created the spoon prior to the knife. The spoon was more complex to build because wrapping the texture around the bowl presented several challenges. Once I determined the correct properties and positions, I was able to duplicate my technique for the knife.

The spoon, knife, and lamp were my most complex objects as they consisted of multiple connected 3D shapes. I used a box for both the spoon and knife handles and a sphere for the spoon bowl. I flattened and connected an elongated, thinner box to make the knife blade. The lamp consisted of three objects: a sphere and two cylinders. The most complex part of making the lamp was setting up the ambient light to shine through only the sphere. I achieved this effect by positioning the light directly in line with the sphere and rotating it to accurately cast a shadow behind the lamp. For lighting, I wanted to emphasize the textures, so I placed a reflective light over the silverware to show the shininess of the materials as well as the subtle color effects simulated in the code.

To navigate throughout the scene, I programmed the camera to move forward, backward, left, and right using the standard WASD keys, with Q and E to move up and down. P and O provide an aerial/overhead view of the scene, and a scroll function was added to alter the speed of the camera.

I used the custom functions in my program to make the code modular and organized. Each function performs a single, focused task, allowing me to reuse the code for multiple objects by simply changing parameters rather than rewriting the function. For example, SetTransformations() handles scaling, rotation, and translation, so any object can use it by passing different values. By keeping functions simple and parameter-driven, I ensured the code is reusable, easier to maintain, and can be applied consistently across all objects in the scene.