**10/26/2025**

**David Parks**

**CS 330 Final Project**

**Design Decisions**

**Overview of the Scene**

For my final project I set out to build a small still life centered on a coffee mug resting on a tiled ground plane. I constructed it from simple primitives to keep the polygon count low and the modeling time efficient. The scene currently includes a ground plane, a mug made from multiple cylinders, a torus handle, a short metal base ring, and a shallow cone that acts like a lid leaning on the rim. My goal was a clean, readable composition that demonstrates the full pipeline from geometry through texturing, lighting, camera control, and projection.

**3D Objects and Geometry Decisions**

**Primitive choices and composition.**

* **Plane:** Serves as the “table” or floor. A single quad mesh that I tile with UV scaling so a high-resolution texture can repeat cleanly.
* **Mug outer body:** Cylinder scaled in height and radius.
* **Mug inner wall:** A second cylinder with slightly smaller radius and slightly different height so the lip reads properly.
* **Metal base ring:** A short cylinder beneath the mug to add a material change and visual interest.
* **Handle:** Torus rotated and scaled to look like an inset handle.
* **Lid:** Cone scaled very shallow and tilted so it leans against the rim.

**Polygon budget.**  
All meshes are low-poly and well under the 1,000-triangle guidance per object. Using basic shapes helped me keep topology simple and evenly spaced. I avoided unnecessary subdivisions and relied on materials and lighting to sell the form.

**Transforms and placement.**  
I composed each object with a consistent transform order: scale, rotate X/Y/Z, then translate. This made it straightforward to sit the base on the plane, sink the body slightly into the base to avoid seams, align the inner wall, and position the handle with a small inset so it looks attached. The lid uses a small tilt in X and Y to feel casually placed.

**Texturing Decisions**

**Texture sources and settings.**  
I used three royalty-free textures at 1024×1024 or higher: a porcelain look for the mug, a brushed metal for the base ring, and ceramic floor tiles for the ground plane. I load them with stb\_image, flip vertically to match OpenGL UVs, and upload as GL\_RGB8 or GL\_RGBA8 as appropriate. I enable mipmaps and set GL\_LINEAR\_MIPMAP\_LINEAR for minification and GL\_LINEAR for magnification.

**Tiling and scaling.**  
On the plane I set the sampler wrap to GL\_REPEAT and apply a UVscale uniform so I can tile the floor texture many times across a large ground without stretching. For the mug parts I reset UVscale back to 1 so the porcelain does not repeat visibly.

**Lighting and Shading Decisions**

**Lighting model.**  
I implemented a Phong shader with uniforms for a material and three light types: a directional light, an array of point lights, and an optional spotlight. For this scene I use two active lights:

* A **directional key light** to define the main shading and specular highlights.
* A **warm point light** as a fill to keep shadowed areas readable and to add a subtle color contrast.

All three Phong components are present: ambient, diffuse, and specular. Specular power (shininess) is higher on the metal base ring and lower on the porcelain to match expectations. I sample the texture once per fragment and combine albedo with the Phong terms so the UV scaling affects both color and lighting consistently. I also compute the normal matrix on the vertex shader path to correct normals under non-uniform scale.

**Camera, Navigation, and Projections**

**Navigation.**  
I support six-degree movement for exploration.

* **WASD:** forward, backward, left, right.
* **Q/E:** down and up along world up.
* **Mouse:** controls yaw and pitch for look-around with proper sensitivity and clamping.
* **Scroll wheel:** adjusts movement speed so the user can traverse the scene comfortably without changing the field of view.

**Projection modes.**  
I provide both perspective and orthographic views and keep the camera orientation consistent when toggling.

* **Perspective:** glm::perspective(fov, aspect, 0.1, 100) for a natural 3D view.
* **Orthographic:** glm::ortho(left, right, bottom, top, near, far) sized to frame the mug and plane without distortion. A single key tap switches between the two.

**Code Organization and Best Practices**

**Structure and modularity.**

* **SceneManager:** creates and draws objects, binds textures, sets per-object transforms, and pushes material and texture uniforms.
* **ViewManager:** owns the GLFW window, camera state, input processing, and view/projection matrix updates.
* **ShaderManager:** loads, compiles, and links GLSL programs, activates them, and exposes type-safe uniform setters. It caches uniform locations in an unordered\_map to avoid repeated glGetUniformLocation calls.
* **ShapeMeshes:** provides reusable primitive meshes for plane, cylinder, torus, and cone.

**Comments and formatting.**  
I follow consistent indentation, group related code, and annotate decisions that are easy to forget later, such as why UVscale resets after drawing the tiled plane or why a body cylinder sinks slightly into the base to hide seams.

**Functional logic.**  
Initialization steps are separated from the render loop. Error messages report texture loading failures or shader compilation issues. I avoid magic numbers by naming key sizes and positions, and I validate texture tags before binding to prevent undefined behavior.

**Reflection on Choices**

**Why these objects.**  
The coffee mug scene is a practical way to meet the requirement of at least four primitives, and it maps well to real objects. The composition lets me demonstrate multi-part modeling on a single item while keeping polygon counts low.

**Why these textures.**  
Porcelain, metal, and tile are common materials with clear visual cues. They also show how different specular levels affect perception under the same lights. High-resolution textures with mipmaps produce a clean look without aliasing.

**Why this lighting.**  
The directional key defines form, and the warm point light adds color variation and keeps the porcelain from looking flat. Using both lights ensures the model reads well from multiple view angles as the user navigates.

**Custom functions and reusability.**  
Utility functions like SetTransformations, SetShaderTexture, SetTextureUVScale, and material setters keep drawing code concise and consistent across objects. The uniform setters in ShaderManager are reusable wherever I need to push GLSL values, which reduces boilerplate and errors.

**Looking Forward**

This project strengthened my understanding of the full OpenGL path from data to pixels and improved my C++ discipline. I plan to extend the scene with shadow mapping, normal maps, and HDR/tone mapping once the base is fully stable. Longer term, these skills support my IT and cybersecurity goals by enabling high-signal visual tools and dashboards, and by building my comfort with GPU-style parallel thinking and performance-minded coding.