**Reflection on 3D Scene Development**  
Joshua Torres  
CS-330: Computational Graphics and Visualization  
Sherin Aly  
October 19, 2025

**Reflection on 3D Scene Development**

During the development of my 3D scene, I made intentional decisions regarding the objects included and how they would function within the virtual environment. I selected basic geometric shapes such as a cube and a floor plane to demonstrate textures, spatial relationships, and object interaction. These objects were chosen because they provide a clear visual framework and allow for easier debugging and testing of the scene. Throughout the development process, I explored different approaches for object loading and rendering, but I encountered numerous challenges. My code frequently failed to compile, and the errors were difficult to trace, which caused delays and required multiple attempts at adjusting my methods. Despite these issues, the process strengthened my problem-solving skills and helped me better understand the complexities of 3D graphics programming (OpenGL, n.d.).

User navigation was a key focus in designing the scene. I implemented keyboard controls to move forward, backward, and sideways, along with mouse input to adjust the camera orientation. This setup allows the user to explore the scene freely and provides a realistic perspective. Implementing these controls was challenging because it required careful handling of camera transformations and perspective calculations. By iteratively testing and refining the input system, I achieved a functional navigation experience that responds smoothly to user interactions (GLM, n.d.).

Custom functions played an important role in organizing my code and making it modular. I developed reusable functions for tasks such as texture loading and object rendering, which reduced code repetition and made debugging more manageable. Each function was designed to perform a specific task, allowing me to isolate problems and make incremental improvements. While some functions initially caused compilation errors, restructuring and testing each one enhanced my understanding of modular programming. These functions also provide a foundation for future projects, as they can be reused or adapted for other 3D scenes (OpenGL, n.d.; GLM, n.d.).

Overall, this project presented significant challenges, particularly with compilation errors and debugging complex code. Despite these obstacles, I was able to justify my development choices, implement user navigation, and create reusable functions that improved the organization of my program. The experience reinforced the importance of iterative testing, problem-solving, and modular code design. Although the scene is not yet perfect, the lessons learned during this project will guide future development and enhance my skills in 3D graphics programming.

**References**

GLM. (n.d.). *OpenGL Mathematics (GLM) Documentation*. Retrieved from <https://glm.g-truc.net>OpenGL. (n.d.). In *LearnOpenGL*. Retrieved from <https://learnopengl.com>