# Milestone Two: Software Design/Engineering

Hector Maldonado

Southern New Hampshire University

CS-499: Computer Science Capstone

Professor Ramsey Kraya

7/17/2025

## Artifact Enhancement Narrative: CS-499 Enhancement Project

### Brief Description of the Artifact

The artifact is a C++ OpenGL-based 3D scene renderer, originally developed as part of my coursework in October 2024 for CS-330: Comp Graphic and Design. The project features a modular architecture for rendering a toy room scene, including lighting, textured objects, and interactive camera controls. The codebase leverages modern C++ practices and OpenGL libraries, and is organized into several classes such as SceneManager, ShaderManager, ViewManager, TextureManager, and MaterialManager.

### When Was It Created?

This artifact was created in October 2024, prior to my coursework in Current/Emerging Trends in CS, Full Stack Development I, and Mobile Architect & Programming.

### Justification for Inclusion in My ePortfolio

I selected this artifact for my ePortfolio because it demonstrates my ability to design, implement, and enhance a moderately complex software system using advanced object-oriented programming and graphics APIs. The enhancements I made showcase my growth in software engineering, particularly in modularity, maintainability, and collaborative readiness. This project is a strong representation of my technical skills and my ability to apply best practices in real-world scenarios.

### Why Did I Select This Item?

I chose this project because it was a significant milestone in my development as a software engineer. It required me to integrate multiple areas of knowledge, including C++ programming, graphics rendering, and software design patterns. The artifact also provided a solid foundation for demonstrating enhancements that align with professional software development standards.

### Specific Components Showcasing My Skills and Abilities

- **Modular Design:** The refactoring of the codebase to use TextureManager and MaterialManager classes demonstrates my understanding of modularity and separation of concerns.

- **Design Patterns:** Implementing the Factory and Singleton patterns in the codebase highlights my ability to apply industry standard design solutions.
- **Documentation and Maintainability:** Enhanced in line comments found by searching *** UPDATED *** and using Doxygen style documentation throughout the codebase show my commitment to code clarity and collaborative development.
- **Graphics Programming:** The use of OpenGL for rendering, shader management, and interactive camera controls illustrates my proficiency in graphics programming and real-time rendering.

**How Was the Artifact Improved?**

The artifact was improved by refactoring monolithic classes into smaller, reusable components. Texture and material management were moved out of SceneManager and into dedicated manager classes, improving modularity and scalability. Design patterns such as Factory and Singleton were implemented where appropriate. The codebase was further enhanced with improved documentation, consistent naming conventions, and better support for collaborative development.

**Did I Meet the Course Outcomes Planned in Module One?**

Yes, I met the course outcomes I planned to address in Module One. The enhancements demonstrate my ability to design and deliver professional-quality software, support collaborative environments, and communicate technical decisions effectively.

**Do I Have Any Updates to My Outcome-Coverage Plans?**

Currently, I do not have any updates to my outcome-coverage plans. The enhancements made align well with the intended outcomes and showcase my growth as a software developer.

**Reflection on the Enhancement Process**

Enhancing and modifying this artifact taught me the importance of modularity and maintainability in software design. I learned how design patterns can simplify complex systems and make them more extensible. One of the main challenges I faced was ensuring that the refactoring did not break existing functionality, which required careful incremental changes and thorough testing. I also had to resolve type mismatches and header inclusion issues, which deepened my understanding of C++ compilation and linking. Overall, the process reinforced the value of clean code practices and effective documentation, both for individual productivity and for supporting team-based development.

**Challenges Faced**

- **Type Mismatches:** Refactoring required unifying struct definitions across files, which led to several type mismatch errors that I resolved by centralizing type definitions.
- **Header Inclusion Order:** OpenGL and GLEW header inclusion order caused compilation errors, which I fixed by standardizing the included order in all relevant files.
- **Maintaining Functionality:** Ensuring that the new modular design did not disrupt existing features required careful testing and debugging.

**Conclusion**

Completing this enhancement project has been a rewarding and transformative experience. Through the process of refactoring and improving the original artifact, I have deepened my understanding of modular software design, object-oriented programming, and best practices in code maintainability. The challenges I encountered such as resolving type mismatches, managing dependencies, and ensuring the integrity of the codebase have strengthened my problem-solving skills and attention to detail. This project not only demonstrates my technical abilities but also reflects my commitment to continuous learning and professional growth. By including this artifact in my ePortfolio, I am able to showcase both the evolution of my skills and my readiness to contribute to complex software development projects in a collaborative environment. I am proud of the progress I have made and look forward to applying these lessons and skills to future endeavors in computer science and software engineering.