# CS 499 Milestone Two: Software Design and Engineering Enhancement

## Introduction

Hello, my name is Sydney Porter, and this is my code review for CS 499 Milestone Two. In this document, I will review and enhance an artifact that demonstrates my knowledge and skills in Software Engineering and Design. I will highlight the current functionality of the code, identify areas for improvement, and explain how I have enhanced this artifact to align with the course outcomes and prepare it for my ePortfolio.

## Selected Artifact: Task & TaskService Classes (CS 320 Project)

This project demonstrates my ability to design and implement software solutions using object-oriented principles.

### Existing Code Functionality:

- The `Task` class represents a task with three attributes: `taskId`, `name`, and `description`, with encapsulated access.

- The `TaskService` class manages tasks using a `HashMap` and provides methods to add, delete, and update tasks.

### Code Review Findings:

\*\*Strengths:\*\*  
- Clear separation of concerns between data representation (`Task`) and business logic (`TaskService`).  
- Use of `HashMap` ensures efficient retrieval and modification of tasks.

\*\*Weaknesses:\*\*  
- Lacks error handling for invalid task IDs.  
- No input validation for `taskId`, `name`, or `description`.  
- Minimal comments and Javadoc for maintainability.

### Enhancements Implemented:

- Implemented error handling, throwing exceptions for invalid operations.  
- Added input validation to ensure attributes are non-null and non-empty.  
- Enhanced documentation with Javadoc and inline comments.  
- Replaced `HashMap` with `ConcurrentHashMap` for better concurrency support.

## Skills Demonstrated and Course Outcomes Alignment

By enhancing this artifact, I have demonstrated the following skills:

- \*\*Secure and Efficient Coding Practices:\*\* Implemented input validation and error handling to ensure data integrity.

- \*\*Software Scalability:\*\* Improved concurrency management by replacing `HashMap` with `ConcurrentHashMap`.

- \*\*Object-Oriented Design Principles:\*\* Ensured encapsulation, maintainability, and modular code structure.

- \*\*Technical Communication:\*\* Improved documentation through Javadoc and inline comments for clarity.

These enhancements align with the CS 499 course outcomes by ensuring that the code is scalable, maintainable, and secure, which are essential attributes of professional software development.

## Conclusion

For this milestone, I selected and enhanced the Task & TaskService Classes from my CS 320 project to demonstrate my expertise in Software Engineering and Design. By improving error handling, input validation, concurrency management, and documentation, I have refined this artifact to meet professional software development standards. This submission is an important step toward finalizing my ePortfolio and preparing for a career in software engineering.