**CS 499 Capstone: Initial Enhancement Plan**  
**Student: Malachi Okongwu**  
**Date: [5/9/252]**

**Overview**

This enhancement plan outlines my strategy for completing the CS 499 Capstone ePortfolio. The plan includes selected artifacts from previous coursework and proposed enhancements in three key areas of computer science: software design and engineering, algorithms and data structures, and databases. Each enhancement is intended to showcase growth in my skills, align with the course outcomes, and support my career goals in software development.

**Artifact Selection and Enhancement Overview**

I have chosen **one comprehensive artifact** that will be enhanced across all three areas. This artifact is a **Craps game simulation project**, originally developed in Unity using C#. It reflects a wide range of skills across software design, logic implementation, and potential database integration.

**Enhancement Plan by Category**

**1. Software Design and Engineering**

* **Original Artifact**: Unity-based Craps Game
* **Created in**: CS 340 – Advanced Programming Concepts
* **Planned Enhancements**:
  + Refactor the existing code to follow SOLID principles and modular architecture.
  + Implement MVC (Model-View-Controller) pattern to improve maintainability and scalability.
  + Add a more intuitive and responsive user interface (UI) using Unity's UI system.
  + Improve game flow and usability through better state management and error handling.
* **Course Outcomes Addressed**:
  + Demonstrate the use of innovative tools and best practices in software engineering.
  + Design solutions that manage trade-offs in software architecture.
  + Employ effective software development practices aligned with industry standards.

**2. Algorithms and Data Structures**

* **Original Artifact**: Game logic and state management code from the Craps Game
* **Planned Enhancements**:
  + Optimize dice roll simulation logic and game state transitions.
  + Implement a robust data structure (e.g., dictionaries or linked lists) to track and analyze game statistics, player history, and probability distributions.
  + Include complexity analysis (Big O notation) and performance benchmarking for improved logic.
* **Course Outcomes Addressed**:
  + Apply algorithmic principles to evaluate and improve computing solutions.
  + Optimize code for performance and efficiency while maintaining readability and usability.

**3. Databases**

* **Original Artifact**: Currently lacks a database component
* **Planned Enhancements**:
  + Integrate a SQLite or MySQL database to persist game results, player profiles, and session statistics.
  + Use CRUD operations (Create, Read, Update, Delete) for player management.
  + Create a small dashboard in Unity to view historical data retrieved from the database.
* **Course Outcomes Addressed**:
  + Demonstrate use of database technologies to implement software solutions.
  + Ensure data integrity, security, and efficient access in computing applications.

**Career Specialization Focus**

My specialization is in **full-stack software development**, with a strong interest in game development and systems integration. Enhancing a game project with modular software design, efficient algorithms, and database functionality demonstrates my readiness to develop scalable, interactive applications—skills that are highly transferable to real-world development teams and enterprise-level software engineering roles.

**Conclusion**

This plan reflects a holistic approach to refining a single artifact in three distinct areas of computer science. The enhancements are designed not only to meet academic requirements but also to provide a compelling and professional showcase of my capabilities. Feedback from my instructor will help further refine the direction of each enhancement and ensure alignment with industry expectations.