Craig Rowell

CS-499: Computer Science Capstone

3-2 Milestone Two: Enhancement One: Software Design and Engineering

June 29, 2025

The first artifact I selected for enhancement is a Java-bases appointment management application that was initially developed for Software Test Automation and QA. The original application was bare-bones, using the console to create, view, and manage nondescript appointments, contacts, and tasks. I chose to enhance it by adding a basic GUI built with Java Swing, and add a Model-View-Controller (MVC) architectural pattern to prepare it to support basic CRUD operations on user-generated entries. This artifact was created fairly early on in my academic journey and has since become a foundational example of my skills as a software engineer.

I chose this artifact for my ePortfolio because it captures the core of software design and engineering principles: modular design, user interaction, and controller-based logic. I saw significant room for enhancement, especially in the user experience and system extensibility. As part of this milestone, I’ve added a new tabbed interface to integrate a calendar view and streamline navigation between different management functions – adding, updating, and removing appointments, contacts, and tasks. The enhancements I’ve made have prepared this artifact for integration with the nosql database that will be accessed with my third artifact, the CRUD developed in Client-Server Development.

The enhancements I made directly align with the course outcomes related to software design and engineering. In module one, I outlined a plan to improve modularity and interface design of this application, and I am confident I met or exceeded those goals. The refactored application demonstrates a clearer separation of concerns, improved component reuse, and a more intuitive interface. I don’t have any major changes to my outcome coverage plans at this stage, as the enhancements completed so far have remained aligned with my original strategy.

Throughout the enhancement process, I gained valuable insights int refactoring legacy code and scaling small applications to support broader functionality. One of the main challenges was integrating the new calendar view without disrupting the existing layout and interaction flow. This required careful redesign of the GUI framework and deeper understanding of event-driven programming in Java Swing. I also faced the challenge of maintaining MVC discipline while adding new features, which prompted me to refactor controller logic to better support scalability. Ultimately, these enhancements not only improved the application, but also reinforced my understanding of clean code practices, modular design, and user-centered development, skills that are crucial to my professional growth as a software developer.