Ryan Miller

CS 499 Milestone Two Narrative

This artifact is a client management system that I originally developed from reverse engineered legacy code in C++, and later enhanced by translating it into Java and converting it from an application interfaced within a terminal environment to a GUI format using Java Swing. This system allows "SNHU Investments," a fictional investment company, to view and edit service options offered to specific clients. The original C++ program was created during my CS 410 Software Reverse Engineering course.

I chose to include this artifact in my ePortfolio under the Software Engineering and Design category because it demonstrates my proficiency in translating a program from one programming language to another and in user friendly GUI development. These enhancements not only improved the usability of the system but also ensured that all functionalities from the original program were retained and even enhanced. There are some things that I need to add or improve as I polish the application including better input sterilization. By translating the program from C++ to Java, I demonstrated my ability to design and evaluate computing solutions to ensure the functionality and performance of the system. Prior to tackling this enhancement, I conducted a code review in which I documented my plan to enhance the original C++ program, I also ensured that the new artifact is well commented and easy to follow. Implementing user authentication and following best practices highlighted the importance of a security mindset during the development process. I still need to finalize the hashing system before fully implementing it in the artifact, but I will complete this aspect of the enhancement as I polish the artifact. My outcome-coverage plans remain mostly consistent, as I still need to finish tackling the hashing facet of my enhancement plans to fully cover the security mindset outcome.

The process of enhancing and modifying the artifact was very informative, I haven't hade much experience creating Java GUIs and I had to relearn a few things. Initially translating the core logic from C++ to Java was frustrating because some parts of the program functioned a bit differently than expected in Java, but in transforming it into a simple GUI, everything became much easier to implement, test, and use. I learned how to use layout managers effectively and how to handle user interactions within the GUI. This project provided excellent practice in problem solving, utilization of best practices, and Java programming concepts. It also highlighted the importance of planning, testing, and user experience in design and software development.