CS 499 Milestone Two Narrative

The artifact I chose for this enhancement is the AppointmentService component from my CS-320 Software Testing and Quality Assurance course project. The original version was a Java-based backend service, created in early 2025 that handled basic CRUD operations for appointment records using in-memory storage. It also included JUnit tests but had no user interface and no structure for scaling into a larger application. I originally created it to practice object-oriented programming and test-driven development.

I selected this artifact because it gave me a strong starting point to show how my skills in software engineering and design have grown. While the original met the basic requirements, it was missing features that are expected in professional applications, such as a scalable architecture, persistent storage, and a user-friendly interface. In this enhancement, I redesigned it as a full-stack web application. I built a RESTful backend with Spring Boot, created a React frontend for user interaction, and used best practices like separation of concerns and clear API endpoint design. These updates demonstrate skills in both frontend and backend development, REST API creation, and form validation on both the client and server sides. The responsive interface, combined with real-time communication between the frontend and backend, shows my ability to create complete, production-style applications.

In my Module One enhancement plan, I targeted Course Outcomes 3 and 4, and I believe this enhancement successfully meets all of them, along with Outcome 2 as well. Outcome 2 was met by creating a detailed, user-friendly README and adding clear code comments throughout the project, ensuring technical communication that is coherent and suited to a developer audience. For Outcome 3, I applied algorithmic principles and industry practices to restructure the code into a modular, decoupled system. I also made design choices with scalability in mind, such as keeping the current in-memory storage but preparing the backend so it can easily be connected to a database. For Outcome 4, I used modern tools and techniques such as Spring Boot for backend development, React for the frontend, and REST API design principles to create a functional and valuable application. My original outcome coverage plan remains the same because the finished project meets the goals I set.

The enhancement process required rethinking the application’s architecture and building connections between a backend service and a modern frontend framework. I learned how to set up RESTful communication, handle CORS configurations, and use both server-side and client-side validation to improve user experience. One of the biggest challenges was keeping the frontend and backend separate but fully in sync. I had to troubleshoot issues with request formats, JSON serialization, and displaying meaningful error messages to users. Through this process, I gained a deeper understanding of full-stack workflows, REST API design, and how to improve usability through better error handling and form validation. Overall, this enhancement transformed a small backend service into a complete web application and strengthened my ability to design and deliver software that meets real-world standards.