Jean Pierre Jacques Toussaint

Professor Sandifer

Southern New Hampshire University

CS-499-19008

November 13, 2025

The artifact I selected for enhancement is the backend API of the Travlr full-stack travel booking application, created in CS-465 (Full-Stack Development). Although the original project included both a client-facing Angular application and an administrative interface, my enhancements in this milestone focused primarily on the backend portion of the system. The artifact consists of Node.js/Express server files, MongoDB models, controllers, routes, authentication logic, and supporting configuration. The original version provided basic trip retrieval, a static admin login, and limited error handling. The API was substantially enhanced in this milestone to meet professional software engineering standards and to prepare the artifact for inclusion in my ePortfolio.

I selected this artifact because it clearly demonstrates my ability to design and implement software using modern engineering practices. Enhancing the API allowed me to showcase skills in backend architecture, database design, secure coding practices, and maintainable code structure. This artifact best represents my growth as a software engineer, especially in the areas of API design, Express middleware, database schema management, and modular organization.

The enhancements significantly strengthened the codebase's professionalism. I improved controller structure, standardized error handling, enhanced validation, added consistent modeling, improved environment configuration, reorganized routes, reinforced database indexes, and improved code readability with high-quality inline documentation. These upgrades demonstrate my ability to refine real-world software based on engineering principles rather than simply adding new features.

The milestone enhancements focused strictly on software engineering and design. The main improvements included:

- Refactoring controllers for clarity, consistent responses, and separation of concerns
- Improving schema design with meaningful indexes, validation rules, and timestamps
- Adding structured request validation middleware to prevent malformed API requests
- Improving JWT authentication flow to support clean authorization logic within routes
- Implementing standardized API error responses with predictable JSON structures
- Cleaning and documenting the codebase using industry-standard commenting practices
- Enhancing environment configuration by isolating secrets and configuration variables in a .env file
- Improving database integrity through unique indexes and controlled update paths
- Ensuring the API is stable, testable, and maintainable

These improvements strengthened the overall architecture and brought the artifact closer to industry standards.

The enhancement directly supports the following CS-499 program outcome:

“Demonstrate an ability to use well-founded and innovative techniques, skills, and tools in computing practices for the purpose of implementing computer solutions that deliver value and accomplish industry-specific goals.”

All enhancements were chosen to improve maintainability, clarity, structure, and professional readiness. During this milestone, I stayed strictly within the planned Software Engineering & Design category. Although some improvements also promote better security and cleaner logic organization, the central outcome demonstrated is software engineering.

No changes are required to my outcome-coverage plan from Module One.

Enhancing this artifact taught me how to approach backend development like a professional engineer. I gained experience designing a REST API that is secure, scalable, and maintainable. Implementing JWT authentication deepened my understanding of modern stateless security models. Adding trip creation and updating required thoughtful schema validation and careful control of allowed fields to prevent mass-assignment vulnerabilities.

The email-based password reset system introduced me to SMTP configuration, mail transports, and cryptographic token generation. I also learned how to handle asynchronous controller logic, manage MongoDB indexes, and maintain clean separation between controllers, models, routes, middleware, and configuration files. During this process, I also encountered

challenges with SMTP authentication. To resolve these issues, I consulted professional developer resources such as Stack Overflow, where several contributors recommended using Mailtrap for secure development and testing. Adopting Mailtrap allowed me to fully validate the password-reset workflow without exposing real credentials, demonstrating my ability to research solutions, evaluate trusted sources, and apply industry-standard tools.

Challenges included resolving index conflicts in MongoDB, configuring a reliable mail transport, implementing secure environment variable management, and ensuring backward-compatible API responses for the Angular front end. Each challenge strengthened my debugging, refactoring, and problem-solving abilities.

Overall, this enhancement process significantly improved my practical backend engineering skills and prepared the artifact for presentation in my professional ePortfolio.