Michael Isenhour

January 26, 2025

CS 499 – Computer Science Capstone

**Enhancement 1 Narrative**

**Briefly Describe the Artifact**

The artifact I chose for this enhancement is the backend of my Full-Stack Development 1 project, Travlr Getaways. This application was originally created using the MEAN stack (MongoDB, Express.js, Angular, and Node.js) and was designed to allow users to browse trips while administrators could create and edit trip listings. It was initially built as part of a full-stack development course that taught me the fundamentals of Full-Stack Development. However, I noticed flaws while developing this MEAN Stack application, such as no login page on the website or a way for administrators to delete trips to name a few, that I couldn’t fix during the course since we had to follow a PDF guide, so I decided to transition the project to Flask and PostgreSQL to enhance its overall structure, security, and scalability while showing my ability to adapt to new technologies.

**Justification for Inclusion in ePortfolio**

I included this artifact in my ePortfolio because it represents a major step in my growth as a software developer. This project not only demonstrates my skill to build full-stack applications but also highlights my ability to move an application from one technology stack to another. The enhancements I implemented highlight my skills in backend design, modularization, security, and database management. By refactoring the project to use Flask instead of Node.js, I improved its organization and overall readability by using Flask Blueprints to organize different components, making the code more scalable. I also implemented Flask-Login for session-based authentication and bcrypt for secure password storage, which improved security significantly. In addition, I replaced MongoDB with PostgreSQL, allowing me to take advantage of relational database features such as foreign key constraints and indexing for better performance. These improvements not only strengthened the application’s security and efficiency but also demonstrated my ability to use industry-standard best practices.

**Meeting Course Outcomes**

Through this enhancement, I successfully met the course outcomes I set out to achieve in Module One.

**Collaborative Environments**  
I successfully structured my Flask application using blueprints, making it easier to manage different modules like authentication and trip management. I also implemented an RBAC system to ensure proper role management, allowing admins to add, edit, and delete trips while restricting regular users to only viewing them. This setup improves the application's security and makes it easier for future developers to understand and maintain.

**Professional Communication**  
While I am still finalizing documentation to explain RBAC and database design, I have focused on writing clear comments and structured code. I also implemented flash messages to provide users with real-time feedback on actions like logging in, registering, and submitting forms, improving user interaction and responsiveness.

**Computing Solutions**  
I enhanced the application by implementing search and filtering functionality, allowing users to find trips based on destination, budget, and start date. To optimize performance, I applied PostgreSQL indexing for faster searches and pagination to limit the number of trips displayed per page. This ensures smooth performance even with large datasets while maintaining a user-friendly interface.

**Innovative Techniques**  
I designed the application using Flask blueprints to separate modules, improving organization and maintainability. Additionally, I integrated Flask-WTF for form validation, bcrypt for secure password storage, and PostgreSQL instead of MongoDB for better relational database management. My implementation of pagination and RBAC further enhanced overall performance and security, while Jinja templates improved the front-end experience. I also implemented caching with cookies to allow users to stay logged in when selecting “Remember Me.”

**Security Mindset**  
From the beginning, security was a major focus. To make my application secure I enforced password hashing with bcrypt, implemented CSRF protection with Flask-WTF, and applied strict RBAC policies. All database inputs are also validated to prevent SQL injections, and only admin users have permission to modify trip data. Additionally, newly registered users default to a regular user role, which can only be changed manually in the database, further improving security.

At this point, I am finalizing unit testing, implementing query caching, refining the front end, and completing documentation on RBAC. These final steps will ensure the project is maintainable and aligns with industry standards.

**Reflection on the Enhancement Process**

The enhancement of this artifact was both challenging and rewarding. Before starting this project, I had no experience with Flask-WTF, bcrypt, or Flask-Login, and I had never worked with Flask Blueprints to organize an application. To gain familiarity with Flask I decided to take a week off work and spend that time working 14hour sessions daily, working on this application from 8am-12am. It was in these long sessions that I learned how to properly structure a Flask application, implement secure authentication, and enforce RBAC, while there were frustrating times, it was a significant learning experience. One of the biggest challenges I faced was ensuring that only admin users could access certain routes while still maintaining a smooth user experience. It took some trial and error to get the authentication and authorization systems working correctly, but through persistence and research, I was able to implement a secure and scalable solution. Additionally, transitioning from MongoDB to PostgreSQL required me to rethink how I structured and queried data, but in doing so, I gained a much stronger understanding of relational databases.

Overall, this enhancement process pushed me to develop new skills, improve my problem solving abilities, and reinforce best practices in software development. Not only did I gain hands on experience with a focus on security design and modular application architecture, but I also built confidence in my ability to move between technology stacks. This project is a demonstration of my ability to adapt, learn, and apply new concepts successfully, and it shows a representation of my skills in software design and engineering.