**Jessica Millet**

**CS 499: Computer Science Capstone**

**September 19, 2025**

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

My Travlr Getaways Web Application, which I created in the CS 465 Full Stack Development I course, is the artifact I chose. It is a full-stack app that uses MongoDB, Express, Angular, Node.js, and Passport and JWT for authentication, in which users can browse and book trips.

I chose this artifact because it shows my skills in developing a full-stack web application. My skills are shown through routing on the server side in the **app.js** code file, API endpoints, authentication in the **authentication.js** file, and fetching trip data in **travel.js**. I improved my artifact by adding more inline comments where needed and splitting large functions into smaller parts to improve readability and maintainability. In **travel.js**, I split the large async fetch function code block into the **fetchTrips()** helper function code block, which fetches data from the API and gets messages before they are returned to the controller. This improved the code to look cleaner and the web app functionality to remain the same.

I believe I met the course outcomes of the enhancements that I planned in Module One for showcasing my skills in Software Engineering and Design by making improvements to my code to ensure that it is readable and maintainable, because I have clearly explained the purpose of each module, function, and route by adding inline comments. I don’t have any updates to my outcome-coverage plans, which weren’t necessary because my enhancements follow my initial plan and my outcome goals in this course.

As I was making my enhancements, I learned that it is very important to ensure that the comments that I add are easy to understand and read. I believe I figured out how to split large functions, especially complex ones, into smaller logical sections without changing the function of the code. The only challenge that I faced was trying to figure out which parts of the code with large functions could be split into smaller sections without messing up the code functionality. These enhancements follow the coding standards professionally by improving the readability and maintainability of my code.