**Jeanelyn Roman**

**CS-499 Computer Science Capstone**

**Dr. Troy Hawk**

**Milestone Four Enhancement Three: Databases**

**Artifact: CS-465 Full Stack Database Project – travlr.js**

The artifact I chose for the Databases category is a Mongoose model file called travlr.js, developed during CS-465: Full Stack Development with MongoDB, Express, Angular, and Node.js. It was originally created as part of a RESTful API used for managing travel data, such as trip name, resort, dates, price, and description in a full stack web application called Travlr Getaways. This artifact is a key part of the backend and handles the database schema and validation logic for the “trips” collection in MongoDB.

I selected this artifact because it demonstrates my ability to design and implement a real-world database solution within a full stack application. The travlr.js file shows that I understand how to define schemas, enforce data validation rules, and use Mongoose to connect a Node.js backend to a MongoDB database. These are critical database skills for modern web development.

To enhance the artifact, I improved the schema by:  
• Adding more robust validation for fields such as price, startDate, and endDate  
• Using enums to restrict possible values for certain fields  
• Refactoring the model structure to improve readability and maintainability  
• Adding comments to explain the purpose of each field and how it maps to the front end

These improvements help prevent incorrect data from being stored, improve developer understanding, and make the model easier to maintain in the future.

This enhancement helped me grow in a few key areas. I showed I can use databases and development tools to build something real that works. I also focused on data validation to make sure the information stored is accurate and secure, which is important in database design. I cleaned up the code and added comments to make it easier for others to read and use. I was able to achieve the goals I set at the start of this project, and no changes are needed to my original plan.

Enhancing the travlr.js file gave me hands-on experience with improving an existing artifact rather than building something from scratch. This forced me to think like a real-world developer analyzing what could go wrong, identifying weak spots in the schema, and thinking about how other developers might interact with my code.

One challenge I faced was figuring out the best way to add new validation rules without breaking the existing API routes. I had to test the changes carefully to make sure they didn’t disrupt the client side or cause unnecessary errors in Postman. I also had to research Mongoose documentation to understand some of the best practices around error messages and field constraints.

I learned how important strong data modeling is for application stability and security. This project strengthened my confidence in using MongoDB and Mongoose and helped me better understand the trade-offs involved in designing schemas that are both flexible and safe.