Matthew Carrillo

CS 499 Computer Science Capstone

2/6/2025

**Milestone 4 Narrative**

This artifact is a full-stack application that was created between October to December of 2024 for the CS-465 course. It was designed to be a travel package shopping service that has a general user front end for selecting and purchasing travel packages and a back end for administrators to login and manage data and packages as a whole. Throughout this course I will continue to focus on the back end portion of the program meant for administrators.

I selected this artifact because it shows my understanding of how data is entered, stored and retrieved in a database, as well as how authentication security works for both a database and a website. It also helped me to further understand in detail how the data structure must be set up to handle that information, more so than the algorithm had. The artifact was improved by this inclusion in that it was not properly storing data prior to this enhancement, which made the insertion of new package options useless on the front end.

I think that I had met the outcomes outlined in module one for this enhancement. The specific outcomes I had listed in module one included “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” and “Develop a security mindset that anticipates adversarial exploits in software architecture and designs to expose potential vulnerabilities, mitigate design flaws, and ensure privacy and enhanced security of data and resources.” I had met theses outcomes by utilizing multiple programs, such as postman and MongoDB Compass, for managing the data within the program in a way that was efficient and followed industry standards. I had also kept a security mindset by considering each step in the process of how the data is shared between the database and the program both ways and how to protect it. I had needed to make sure that the structure of the application was designed to mitigate exploitation and ensure security standards that protect the data within. For example, when a data transfer happens between the website and the database, there are two encryptions on the password. One of the base 64 encryption methods is only for the transfer and applied to both the email and password, while the other is only on the password and is a permanent change. This makes it so that even when viewing the data in the database via Compass, the password remains visibly altered.

I didn’t really have any issues with MongoDB itself. I mostly had some challenges with the response on the website. I also had some issues with the add and edit functions that are accessible after logging in. I learned a lot about authentication and external database altering, and improved the artifact with improved functionality and a cleaned up interface.