Madisyn Kuczenski

CS-499

**Enhancement 2 Narrative**

The artifact that I chose for Enhancement 2: Algorithms and Data Structure is a Python file and Jupyter file from CS-340 Client/Server Development. The Python file created a connection to a MongoDB database and used CRUD operations to work within the database. The database held different animal profiles used for a rescue project, so the database needed to be updated frequently to work with the changing information. The Jupyter filed created a dashboard for the information from the database to be presented to clients/admins. The Python file and Jupyter file were created in May-June of 2024.

The reason this artifact was chosen for the Algorithms and Data Structure category is based on the fact that there was no validation for what was allowed to be added to the database, which could impact the data structure already put in place. As algorithms are designed to solve problems, implementing a validation check for the data types will solve the problem of incorrect information being added to the database. Additionally, the data structure of organizing the animals into different categories, like animal type, breed, sex, or age will be upheld to make sure the database is high quality. This artifact was improved by creating code that would be added to mongosh to create a validator for the animal collection. When a user is adding information to the database, it will now be required for all fields to be filled in, as well as with specific data types. This helps to maintain the quality and value of the database.

The course outcomes that I planned to meet with this enhancement were: Design and evaluate computing solutions that solve a given problem using algorithmic principles and computer science practices and standards appropriate to its solution while managing the trade-offs involved in design choices 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 met these course outcomes by using algorithmic principles and a security mindset to create a validator to make sure the information entering the database is relevant and filled out correctly. Thinking of the problems that could occur within the database and the security mishaps that could happen by allowing anything to enter the database and creating a prevention, shows the course outcomes have been met. As of now, the original plan that I created for outcome coverage stays the same.

While I was enhancing and modifying the artifact I learned about validators in MongoDB. Originally, I was going to use IF, ELSE statements in the Python code to help validate the information going into the database. However, after thinking about how the project worked, I realized that all of the information added to the database came from MongoDB and that was where the validation needed to occur. I researched different methods of validation used within a MongoDB database and implemented the best option. By utilizing mongosh, it was possible to make certain fields be required and limited to the correct data types.