**Anthony Nguyen**

**CS 499**

**5-2 Milestone Four**

**Enhancement Three**

**Databases**

**Artifact Overview**

"This Animal Shelter Management System was built using Flask as the backend framework and MongoDB as the NoSQL database. The system allows users to perform CRUD operations on animal records, which means they can Create, Read, Update, and Delete data.

One of the key features of the system is role-based access control. Depending on whether you're an admin or a staff member, you’ll have different permissions within the system. Additionally, the system can generate reports based on the data stored, offering useful insights into the shelter’s animals."

**Reasons for Selecting this Artifact**

"I selected this project for my ePortfolio because it demonstrates my proficiency in multiple areas, particularly database management and full-stack development.

It showcases my ability to work with NoSQL databases like MongoDB and implement features that ensure secure and efficient data management. More specifically, I wanted to highlight the advanced querying techniques I used, such as aggregation pipelines, which allow for dynamic report generation. All of these aspects combined make it a comprehensive demonstration of my skills."

**Skills Demonstrated**

"This project gave me the opportunity to develop and enhance several key skills:

* Database Optimization: I created indexes on fields that were frequently queried, such as 'animal\_id', 'name', and 'breed'. This is a critical step in improving query performance, especially as the database grows.
* Security: I implemented hashed password storage and role-based access control to ensure that only authorized users could perform certain actions. For example, only an admin can delete records, whereas staff can update or view them.
* Advanced Querying: By utilizing MongoDB’s aggregation pipelines, I was able to generate reports on animals grouped by specific criteria like breed or age. This ability to manipulate and extract valuable data is an essential skill for database management.
* Full-Stack Development: Beyond the database, I built the entire application, integrating frontend elements like HTML and CSS with the backend using Flask and MongoDB."

**Process Reflection**

"Enhancing this project came with its own set of challenges. The main one was implementing secure user authentication and proper role-based permissions. To address this, I deepened my understanding of Flask’s session management and learned how to implement hashed passwords to securely store credentials.

Another challenge was optimizing the database for better performance. Initially, as the data grew, the system’s queries became slower. To fix this, I created indexes on commonly queried fields, which resulted in a significant performance boost. This was a key moment in my learning process, as it taught me the importance of database optimization in real-world applications."

**Outcome Alignment**

"This project aligns closely with several course outcomes:

* Database Design & Implementation: I designed a scalable NoSQL database to store animal records, which could grow with the shelter’s needs.
* Security & Access Control: The system includes essential security features like hashed password storage and role-based access control, which are critical in any real-world application dealing with sensitive data.
* User-Friendly Interface: Lastly, the application provides an easy-to-use interface that makes managing animal records efficient, whether the user is an admin or a staff member."

**Conclusion**

"In conclusion, this Animal Shelter Management System demonstrates my ability to design, build, and secure a database-driven application. It not only fulfills the requirements of managing animal records efficiently but also highlights important real-world concerns like performance optimization and security.

Thank you for your time, and I’m happy to answer any questions you may have."