# Milestone Three: Algorithms and Data Structures Enhancement

## Sydney Porter

## Southern New Hampshire University

## February 9th, 2025

\*\*Introduction\*\*

Hello, my name is Sydney Porter, and this is my code review for CS 499 Milestone Three. In this document, I will review and enhance an artifact that demonstrates my knowledge and skills in algorithms and data structures. I will highlight the current functionality of the code, identify areas for improvement, and explain how I have enhanced this artifact to align with the course outcomes and prepare it for my ePortfolio.

\*\*Selected Artifact: Animal Shelter Dashboard & CRUD Operations\*\*

This project demonstrates my ability to design and implement efficient algorithms and data structures. The artifact consists of a MongoDB-backed CRUD service and a Dash-based interactive dashboard for filtering and visualizing animal shelter data.

\*\*Existing Code Functionality:\*\*

- The `AnimalShelter` class handles Create, Read, Update, and Delete (CRUD) operations using MongoDB.

- The `animal\_shelter\_test.ipynb` dashboard retrieves and displays data interactively, allowing users to filter by specific attributes such as breed and rescue type.

- The system processes large datasets efficiently using MongoDB queries.

\*\*Code Review Findings:\*\*

\*\*Strengths:\*\*

- Efficient use of MongoDB for handling structured data.

- Clear separation of CRUD operations and data visualization.

- Interactive UI elements with Dash, enabling real-time data filtering.

\*\*Weaknesses:\*\*

- MongoDB queries lacked indexing optimizations.

- No caching mechanism for frequently requested queries.

- Limited error handling and logging in CRUD operations.

\*\*Enhancements Implemented:\*\*

- \*\*Query Optimization:\*\* Limited field selection in MongoDB queries to reduce data retrieval time.

- \*\*Caching Mechanism:\*\* Implemented `lru\_cache()` for frequently accessed queries, reducing redundant database calls.

- \*\*Improved Data Structures:\*\* Converted MongoDB results into NumPy arrays for faster processing where applicable.

- \*\*Logging & Error Handling:\*\* Enhanced CRUD operations to handle connection failures gracefully and log critical errors.

- \*\*Dashboard Optimization:\*\* Improved filtering logic for better performance and user experience.

\*\*Skills Demonstrated and Course Outcomes Alignment\*\*

By enhancing this artifact, I have demonstrated the following skills:

- \*\*Algorithm Optimization:\*\* Improved query performance and caching strategies.

- \*\*Efficient Data Structures:\*\* Leveraged Pandas and NumPy for efficient data processing.

- \*\*Software Scalability:\*\* Optimized CRUD operations and data retrieval methods.

- \*\*Technical Communication:\*\* Improved documentation and logging for maintainability.

\*\*Conclusion\*\*

For this milestone, I selected and enhanced the Animal Shelter Dashboard and CRUD Operations to demonstrate my expertise in algorithms and data structures. By improving query efficiency, implementing caching, enhancing data structures, and improving error handling, I have refined this artifact to meet professional software development standards. This submission is an important step toward finalizing my ePortfolio and preparing for a career in software engineering.