**Milestone 2 Enhancements**

**Artifact Description**

The artifact I selected for enhancement is an interactive Animal Shelter Dashboard (animal\_shelter.py and ProjectTwoDashboard.ipynb) developed as part of a previous course in the Computer Science program. This application, created using Python and Dash, integrates with a MongoDB database to display animal data filtered by rescue type. Originally designed as two files using Jupyter Dash, the code contained elements for connecting to the database, filtering, and rendering visual components, including a data table and a map. This dashboard project was first created during CS 340 as an assignment to showcase advanced programming concepts in Python.

**Justification for Inclusion in the ePortfolio**

I included these artifacts in my ePortfolio because it effectively showcases my skills in modular programming, web application development, and UI/UX design. The Animal Shelter Dashboard aligns with the course outcomes by demonstrating my technical ability to design a professional, scalable application with real-world functionality. The enhancements I made are significant because they reflect my ability to refactor, improve user experience, and add structural efficiency to a web-based application. Through this project, I was able to highlight my skills in breaking down complex applications into modular components and implement responsive design while integrating multiple programming frameworks.

**Alignment with Course Outcomes**

In the initial planning phase of this project, I set out to achieve several course outcomes. Specifically, I aimed to:

1. Demonstrate Modular Programming (Outcome 4) by separating the application logic into different modules. This not only improved code readability but also allowed for easier maintenance and future enhancements.
2. Apply web application development skills by setting up a Flask server to support the Dash app to keep enhancing my abilities in web frameworks and server management.
3. Show UI/UX design experience by refining the interface to make it more user-friendly and visually responsive (Outcome 2), which is essential for presenting information in a way that is easily accessible and engaging to the audience.

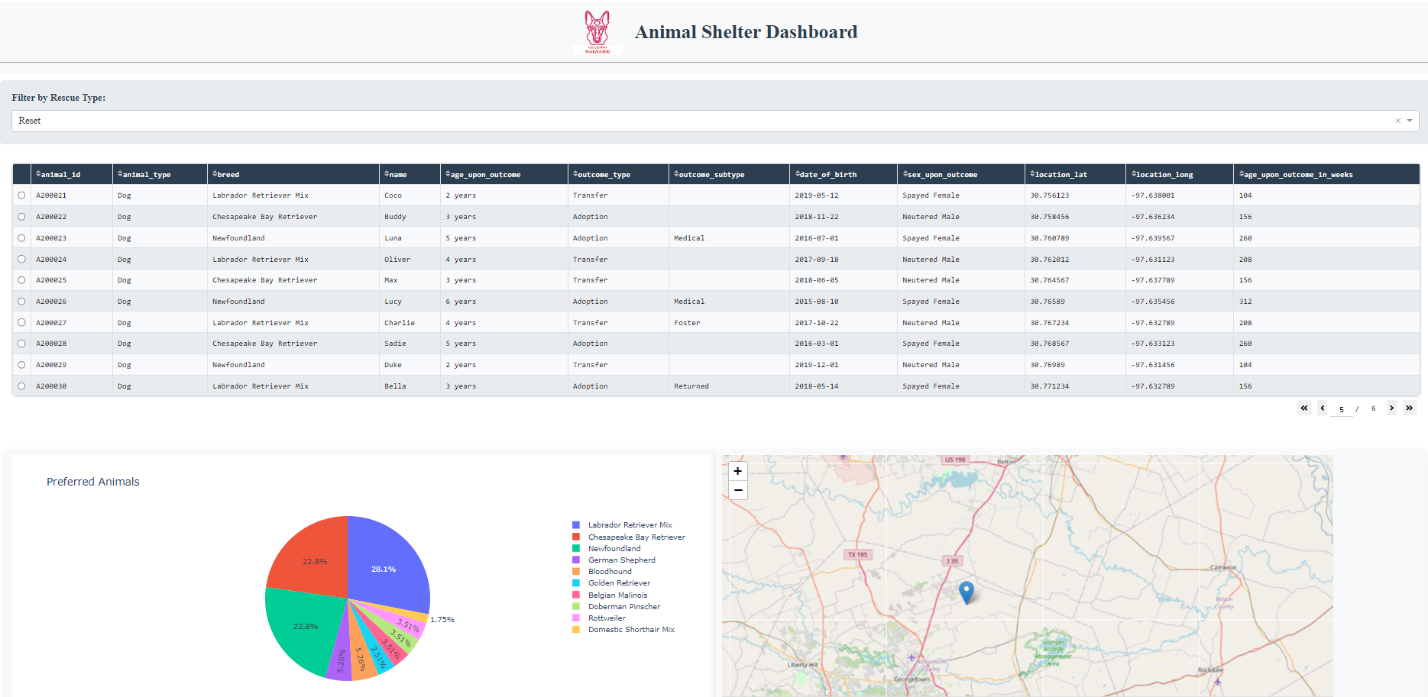
**Enhancement Process and Challenges**

The planned enhancements were focused on modularizing the code, setting up a standalone server, and refining the interface. Here’s a breakdown of the steps I took:

* **Code Modularization** - The dashboard was separated into multiple Python files to support a modular structure:
  + **app.py** - serves as the main application file, initializing the Flask server and registering callbacks.
  + **callbacks.py** - handles all interactive functions, such as data filtering and map updates.
  + **layouts.py** - defines the layout and styling of the user interface.
  + **data\_processing.py** - includes functions for retrieving and processing data from the MongoDB database.

This modular approach enhanced maintainability by isolating logic into specific files to make future updates and debugging more efficient.

* **Flask Server Integration** - I configured a Flask server to run the Dash app, enhancing the app’s scalability and giving it a more professional, web-based structure. This step involved setting up Flask as the main server environment and integrating it with Dash’s layout, which required adjustments in the routing and callback functions.
* **UI/UX Enhancements** - The dashboard layout was redesigned for better user experience:
  + Improved navigation by grouping related components, such as the map and data table, into a unified layout.
  + Responsiveness was implemented to ensure that the application functions smoothly across various screen sizes, using CSS Flexbox for adaptable layout.
  + Minor color and font adjustments were made for improved readability and aesthetics.
  + Realigned the header section to remove empty space.
  + Added sorting to table headers to allow more interaction with records.



**Reflection**

The enhancement process reinforced the importance of modular design in creating scalable and manageable applications. By separating the code into multiple files, it clarified the program structure, making it more intuitive and efficient. I have also found in this assignment and previous ones that a well-structured application is far more helpful to me as a developer to more efficiently isolate and focus on areas that need attention. Additionally, I added very thorough comments to further help myself in this regard.

Debugging was another important aspect of this project, particularly in troubleshooting UI responsiveness and ensuring smooth data retrieval from MongoDB. Adjustments to CSS and Dash layout properties were necessary to address layout inconsistencies and ensure a smooth user experience.