Category Two Artifact - Narrative

1. Briefly describe the artifact. What is it? When was it created?

The artifact that I chose for category three is an interactive web application called the Austin Animal Center (AAC) dashboard. It was initially developed as the final project of my CS 340: Client-Server Side Application course taken from July 2024 to August 2024. The motivation behind the AAC dashboard was to streamline the process of identifying suitable dogs for search-and-rescue training based on existing data from five animal shelters. The AAC Dashboard is built using Python, Dash, MongoDB, and Jupyter Notebook. It visualizes data from the Austin Animal Center's animal shelter database, providing insights into animal breeds, demographics, and locations. The dashboard features a sortable and filterable data table, interactive charts (pie chart for breed distribution), and a map displaying the location of selected animals.

1. Justify the inclusion of the artifact in your ePortfolio. Why did you select this item? What specific components of the artifact showcase your skills and abilities in software development? How was the artifact improved?

I chose this artifact because it represents a significant project demonstrating my software development skills and abilities. It is a practical application of the concepts learned in CS-340 and goes beyond simple code examples to create a functional and informative tool.

Showcased Skills and Abilities in Software Development:

* **Data Handling and Processing**: The dashboard demonstrates my ability to retrieve, process, and manipulate data from a database (MongoDB). The use of Pandas DataFrames to structure and clean the data, including handling missing or inconsistent data (like the \_id column), showcases my data manipulation skills.
* **Web Application Development**: Building the dashboard with Dash highlights my ability to create interactive web applications. The use of Dash components (tables, graphs, maps) and callbacks to manage user interactions demonstrates my understanding of front-end and back-end web development principles.
* **Data Visualization**: The pie chart and the interactive map visualize data in a clear and engaging way. This demonstrates my ability to choose appropriate visualization methods and use libraries like Plotly and Dash Leaflet effectively.
* **Responsive Design**: The layout is designed to be responsive, adapting to different screen sizes. This demonstrates my understanding of responsive design principles and the use of Bootstrap for styling.
* **Code Structure and Organization**: The code is also well-structured, with a clear separation of concerns (data access, layout, callbacks). This demonstrates my ability to write clean and organized code.

Artifact Improvements:

* **Code Cleanup**: Removed deprecated code (jupyter\_dash, app.run\_server), cleaned up FIX ME comments, and improved code comments for clarity.
* **Error Handling**: Enhanced update\_dashboard(), update\_graphs(), and update\_map() with try/except blocks to prevent app crashes due to data issues or unexpected errors. Improved error messages provide more helpful debugging information.
* **Responsiveness**: Made the footer and the logo/title section responsive. Charts now responsively resize thanks to the Bootstrap grid layout.
* **Styling**: Styled the data table header (bold, larger font). Added left margin to filtering options.
* **Data Table Enhancements**: Added virtualization=True to the data table for better performance with large datasets and implemented a slider for pagination.
* **Logo and Title**: Added a try block to handle potential FileNotFoundError for the logo. Implemented a Flexbox layout for improved visual appeal and responsiveness of the logo and title.
* **Chart Enhancements**: Added a title to the pie chart. Handled cases where viewData is None or empty.
* **Map Enhancements**: Added a nested try/except block for data conversion and validation. Explicitly raises ValueError for invalid coordinates. Handled missing animal names by displaying "Unknown." Corrected the indices used to access latitude, longitude, and name data. Added checks for viewData, dff, and index to prevent errors. Added a check to ensure the row index is within range. Made the map width responsive. Added meaningful comments.
* **Layout**: Fixed the layout to ensure the chart and geolocation chart are displayed side by side.

1. Did you meet the course outcomes you planned to meet with this enhancement in Module One? Do you have any updates to your outcome-coverage plans?

Yes, I met the course outcomes I had planned to meet with this enhancement. The enhancements directly address course outcomes related to:

* **Course Outcome #1**: Employ strategies for building collaborative environments that enable diverse audiences to support organizational decision-making in the field of computer science.
  + The dashboard is specifically designed to support organizational decision-making at the Austin Animal Center. It provides key insights into animal demographics, breed distribution, and geographic location. Although the dashboard’s purpose is to identify suitable dogs for search-and-rescue training, the dashboard could be used to inform decisions related to resource allocation, adoption programs, and targeted outreach efforts.
* **Course Outcome #2**: Design, develop, and deliver professional-quality oral, written, and visual communications that are coherent, technically sound, and appropriately adapted to specific audiences and contexts.
  + The Austin Animal Center Dashboard serves as a professional-quality deliverable. The improvements made (code cleanup, meaningful comments, error handling, responsiveness) all contribute to the professional quality of the code. The narrative and list of enhancements demonstrate my ability to communicate effectively about technical projects.
* **Course Outcome #3**: 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.
  + I have designed a computing solution (the dashboard) to solve a problem (visualizing and interacting with animal shelter data). The overall dashboard and enhancements I have made demonstrate my ability to evaluate and improve the design based on best practices.
* **Course Outcome #4**: 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.
  + The use of Dash, Plotly, Dash Leaflet, and Bootstrap demonstrates my ability to use industry-standard tools and techniques. The enhancements I have made (especially the interactive filtering, geolocation map, and responsive design) show my ability to apply these tools in a meaningful and effective way.
* **Course Outcome #5**: 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.
  + The mongodb\_crud.py file implements the principle of least privilege. The database user account used by the application is granted only the necessary permissions (read and write) to the specific collections required for the dashboard's functionality. It does not have permission to drop collections, create new databases, or perform other administrative tasks that are not needed. This limits the potential damage if the application were compromised.

1. Reflect on the process of enhancing and modifying the artifact. What did you learn as you were creating it and improving it? What challenges did you face?

The process of enhancing and modifying the Austin Animal Center Dashboard has been a valuable learning experience. I gained a better understanding of the dash framework. I learned how to effectively link different components together to create a truly interactive user experience. I also realized how crucial it is to write code that is not only functional but also maintainable. It made it much easier to understand and modify the code later when revisiting code from earlier iterations. Working with real-world data from MongoDB, including cleaning and transforming it using Pandas, reinforced my data handling skills. I learned how to deal with missing values, inconsistent data formats, and other common data challenges. I also gained a better understanding of data visualization principles. From this artifact enhancement, I learned how to choose appropriate chart types, design effective layouts, and present data in a meaningful way.

Some challenges I faced when enhancing the artifact were callback complexity, data consistency, and debugging interactive elements. Managing the interactions between multiple components through Dash callbacks was challenging at times. Keeping track of the flow of data and ensuring that the callbacks were correctly updating the different parts of the dashboard required attention to detail. Ensuring data consistency between the data table, charts, and map was a challenge. I had to be careful to properly update the data in all components when filtering or sorting was applied. Overall, the process of enhancing and modifying the dashboard was a challenging but rewarding experience. I learned a great deal about web application development, data visualization, and software engineering best practices. The challenges I faced helped me to grow as a developer and improve my problem-solving skills. I am now much more confident in my ability to tackle complex software development projects.