

Brief Description of the Artifact:

The artifact is an interactive animal shelter database dashboard that was developed through using Python and Dash. This dashboard enables users to filter and visualize shelter animal data that are based on specific rescue types which offer an intuitive interface for analyzing available animals. The project was originally created for an academic assignment that was focused solely on database integration and interactive data visualization. Over time, the artifact has been enhanced to improve security, efficiency and user experience to make it a valuable demonstration of my database management and software development skills.

Justification for Inclusion in ePortfolio

This artifact was selected for my eportfolio because it highlights my ability to integrate databases with an interactive web application while also implementing best practices in software engineering. The project demonstrates my proficiency in connecting a MongoDB database to a web-based dashboard using the Dash framework. Additionally, it showcases my ability to handle user input dynamically, retrieve and process data efficiently, and visualize key information using interactive charts. These enhancements were made to the artifact to reflect my growing expertise in database security, performance optimization, and front-end improvements. Having this artifact in my ePortfolio allows me to showcase my technical competency in database management and application development to potential employers.

Course Outcomes Achieved and Updates to Outcome-Coverage Plans:

With the enhancements that were made to this project, I believe that I have successfully met different outcomes. Initially, I didn't think that I was meeting security outcomes as well or at all, but as I started to make these enhancements, I realized that I was improving the database security as I started to improve the code. I think I have also demonstrated the ability to design and

evaluate computing solutions that are based on algorithmic principles, particularly in data processing and query optimization. The project improvements that were made align with the outcome of employing well-founded techniques in software engineering, as I have implemented security measures and performance optimizations to enhance the application. Lastly, I have tried to refine the user interface and interactive visualizations that contribute to my ability to develop professional-quality communications that are tailored to a specific audience. While I have tried my best to meet these outcomes, I plan on continuously refining the artifact by further improving database indexing strategies and enhancing the dashboard's interactivity to provide an even better user experience.

Enhanced Code and Explanation of Changes:

The original code was improved in several ways to enhance security, data handling, and overall application performance. The primary changes included securing database credentials, handling missing data, optimizing code performance, and refining the user interface for a more seamless experience.

Key Enhancements and Their Benefits

One of the most critical changes was replacing hardcoded database credentials with environment variables. In the original code, the database username and password were directly embedded, which posed as a security risk if the code were shared or exposed. The enhanced version now retrieves these credentials from environment variables (`os.getenv('DB_USERNAME', 'default_user')` and `os.getenv('DB_PASSWORD', 'default_password')`), ensuring that sensitive information remains secure and prevents any unwanted access. Another important improvement that was made was handling missing data within the DataFrame. Previously, any missing values could cause crashes or display inconsistencies in the data table. Through implementing

`df.fillna("Unknown", inplace=True)`, any missing values are replaced with a placeholder, preventing any errors and improving the robustness of the application. This change ensures that even incomplete databases would be displayed in the correct order, which offers a better experience for the users. Performance optimization was also a major focus, the data retrieval function was refined to process queries more efficiently before converting them into a DataFrame. This adjustment is especially beneficial when it comes to handling large datasets, as it reduces memory consumption and speeds up data processing. For UI improvements, minor refinements were made to ensure consistency and readability. The layout structure was cleaned up to maintain a more organized appearance, which makes it easier for users to navigate the dashboard and interact with the displayed data. The enhancements that were made collectively contribute to a more secure, efficient and user-friendly application. These improvements address security vulnerabilities, enhance data integrity, and optimize performance while refining the visual experience for the end user.

Reflection on the Enhancement Process:

Enhancing and modifying this artifact was a valuable learning experience that deepened my understanding of database security, performance optimization, and user experience design. One of the most significant changes was that replaced the hard hardcoded database credentials with environment variables. The original implementation that stored the username and password directly in the script, originally, posed as a security risk, hence, the updated version ensures that sensitive credentials are not exposed in the source code, which significantly improves security. Another improvement that was made involves handling missing data within the DataFrame. Beforehand, any missing values would cause inconsistencies in the displayed data, which leads to potential errors or crashes. But through the enhancements, I had ensured that missing fields

are properly handled, which results in a more robust and user-friendly interface. This change prevents any disruptions in the data visualization process and ensures that the application remains functional even if we were to work with an incomplete data set. Performance optimization was another critical area of focus. The original data retrieval function did not efficiently process queries which led to slower performance when handling the large datasets. Through refining the query execution and optimizing the conversion of query results into a DataFrame, I was able to improve the application's efficiency, which reduces memory consumption and speeding up data retrieval. This enhancement will ensure a smoother user experience, especially when dealing with extensive shelter records. Overall, the process of enhancing this artifact has strengthened my skills in secure database management, performance optimization, and user interface design. I encountered an array of problems such as debugging data retrieval issues and refining security measures while maintaining the application's functionality. However, overcoming these challenges helped me to deepen my understanding of best practices in database integration and web development. These enhancements make the artifact a strong representation of my technical abilities.