Deje Perkins

Client/Server Development

Grazioso Salvare Dashboard README

August 19th, 2025

For this project I implemented a web based dashboard for Grazioso Salvare. It was designed to analyze animal shelter outcomes using a Python based client server application. The required functionality consisted of building a fully interactive dashboard where users could filter data, produce visual results, and test CRUD operations through MongoDB database. Several tools were utilized like how Python was the primary programming language since it was supported by the Jupyter Notebook environment for development and testing. Another tool was MongoDB being chosen to be the database model for its flexible document based structure and its ability to store and query JSON documents. Due to its scalability and compatibility with PyMongo library it’s a natural fit for integrating Python code. For the web application I used the Dash framework. It provides both views and controller elements of the application. It’s super lightweight, integrates seamlessly with Ploty visuals, and follows the model-view-controller structure. I also used documentation from MongoDB, Plotly, Dash, as well as the Grazioso Salvare logo.

In order to complete the project I first started off with importing the AAC dataset into MongoDB and creating indexes for efficient queries. Next was enabling create, read, update, and delete (CRUD) operations on the database. After that the dashboard was built in Jupyter Notebook using Dash and tested by connecting it to the MongoDB instance. After they were tested a visual display produced the filtered results and outcomes. Lastly the application was validated by running the queries and confirming that they were indeed correct in both text and graphical form.

I ran into quite a few challenges like configuring the connection between the dashboard and MongoDB in the Apporto environment. I know that it requires careful attention to ensure the correct URI format and authentication methods. Another problem was debugging queries when my results didn't display properly on the dashboard, I resolved that through refining the translation of user selected filters in the MongoDB query syntax. When fixing any challenge I carefully looked over everything, tested smaller code sections, and used trial and error to refine the dashboards functionality.





