Jorge Torres

SNHU CS340 Client/Server Development

Project Two

Describe the required functionality.

This project is intended as a dashboard to access the data Within a database in a simpler manner.

This and as requested by Grazioso Salvare, who is seeking a software application that can work with existing data from the animal shelters to identify and categorize available dogs.

We begin by accessing the dashboard:

A screenshot of a computer

Description automatically generated

Here we see the starting dashboard, with the logo, and the hyperlink to the requested website.

We also have the dropdown option to filter the data. And with the interactive table we ca choose one cell or one row.

A screenshot of a computer

Description automatically generated

By choosing the first filter, we see the data table restructure itself. We have a geo Map that activates when choosing a row for a singular animal. We also have a pie chart for the different breeds based on the filter selection.

A screenshot of a computer

Description automatically generated

Now when choosing the next filter, we see the data table changes the information and the pin on the map changed as well. We also see the pie chart updated to reflect the different breeds in this filter type.

A screenshot of a computer

Description automatically generated

the last filter type, and we again see the data table update and the pin move in the geo map. We also see the pie chart again updated based on the filter type.

A screenshot of a computer

Description automatically generated

The last option is the reset option in the dropdown menu. This will eliminate the current filter and go back to the starting dashboard.

A screenshot of a computer

Description automatically generated

Here is a picture of the process of hovering over the Logo and the hyperlink it contains.

**Describe the tools used to achieve this functionality and a rationale for why these tools were used**

Mongo DB:

Mongo was chosen for flexibility of use. Has great connectivity with Python.

Dash Framework:

The best way to build a connection and present a dashboard environment that is easy to use and manipulate.

**Explain the steps that were taken to complete the project.**

We started the strategy by first making the connection to the Mongo client and creating the dash in its simplest form.

From there, we then added the dash features one by one, debugging one at a time, until finalizing the project.

**Identify any challenges that were encountered and explain how those challenges were overcome.**

The greatest challenge was the Update dashboard callback function. This was a massive, time-consuming process. It was not filtering the data, only showing when starting the dashboard. Turns out, the queries were not correct and did not make sense to the filters. Once I changed these to match, then it was smooth sailing from there.

The secondary was getting the picture to show with a unique identifier, as I was trying to have them together in the same function. Once I separated them, then they worked.

How to reproduce the project

1. Clone repository

[jtorres789/CS340: CS340 Class (github.com)](https://github.com/jtorres789/CS340)

1. Install dependencies as needed.
2. Run the dashboard (ProjectTwoDashboard JT COMPLETE.ipynb)
3. Access the dashboard using provided hyperlink, example: Dash app running on <http://127.0.0.1:20080/>