Client Server Development

2/16/2023

Jad Alrehaoui

SNHU

Project 2

Reflection & Summary

The required functionality of this project, is connecting the database to a front end interactive dashboard.

The dashboard consists on the Grasiozo Salvare logo and a unique identifier to credit the developer working on this project.



As well, they require widgets like a table to view records displayed in pages and are selectable, a chart giving an overview about the data being displayed and because the data has coordinates, we can use a map to locate the record physically.



*Chart, pie chart

Description automatically generatedMap

Description automatically generated*

Lastly, Grasiozo required a an interactive filter option, to filter AAC outcomes data by rescue type.



In order to complete this task, I first configured the database and set a username and password in order to securely authenticate and have access to the data I need to work with. Then I ran some commands on the database to add the rescue type as Grasiozo required.

Table

Description automatically generated

This command adds the Water rescue type:

Text

Description automatically generated

This command updates the dogs that have a rescue type of Disaster or individual Tracking:

Text

Description automatically generated

This command updates the dogs that have a rescue type of Mountain wilderness:

Text

Description automatically generated

Once the database was ready, I started coding the model AnimalShelter. First I defined the init function to accept a username and a password to be able to connect to the database securely. Then implemented the CRUD operations of the animal collection. (refer to code in AnimalShelter.py)

Then I started my Jupyter Dash application by importing all the necessary libraries needed. To connect python or Jupyter with MongoDB I had to use a module called pymongo. It’s a client for MongoDB. The implementation is straight-forward.

Dash was used as a framework to implement MVC. The view translates python into HTML relying on the react framework. It is the digital low code framework for rapidly building data apps in Python, R, Julia and F#.

The interactivity of between dash components is made through callbacks. Which can be described as controllers.

In this project, the development was straight-forward, except for the map widget, I could not get it to change when selecting a record. That’s the only thing missing in this application.

Thank you for reading this .

**References:**

*Introduction | Dash for Python Documentation | Plotly*. (n.d.). <https://dash.plotly.com/introduction>

*Part 3. Basic Callbacks | Dash for Python Documentation | Plotly*. (n.d.). https://dash.plotly.com/basic-callbacks