Morgan Masapollo

12/14/2024

CS340 Client/Server Development

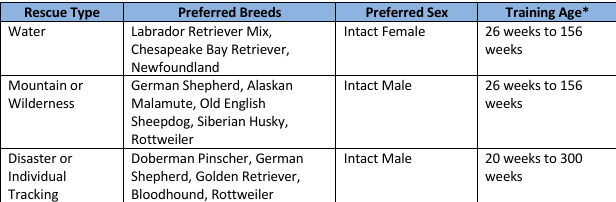
Project 2 Readme

The given assignment that was tasked out to create, was making a web based dashboard for the company Granzioso Salvare in order to aid them in seeing what dogs were available to be adopted and trained for various types of rescue work. The dashboard is constructed through 3 different layers allowing them to be able to interact with the dashboard, gain locations of the animals, and to have accurate up to date information. There is the MongoDB layer which contains the needed information for the dashboards visualization, the python middle-ware layer which is where control is exerted and the plotly, dash, leaflet based visualization layer which is utilized to create the html. The middleware layer is a separate py file that is called on by the main project file and contains the CRUD (create-read-update-delete) actions within mongoDB. This py file contains the methods needed to initiate MongoDB and start the CRUD operations within the code.

MongoDB was chosen for this project as it has a lot of flexibility and is easy to use meaning if a backup was ever to have to be created then a worker could with ease create it even if they are not part of the original team. MongoDB also works more effectively with JSON when being used for creating a web environment. Python was chosen as the language for coding due to its ability to connect with the frontend and backend and having a mass amount of tools to help aid in doing so. In the case of the project python is able to interact with MongoDB with the aid of the CRUD class and middle-ware layer while building the dashboard and other widgets used within the code.

In order to utilize the software the developer will need to set up MongoDb, and input the Austin Animal Center Outcomes dataset into the collection. They then will have to install the dependencies to get the needed python libraries; dash, pandas, plotly, and pymongo.With all of these set up any developer should be able to access the dashboard with their code updated to match their user ID and Password created. From there they will see on the top center the logo for Grazioso Salvare, and the link to the Grazioso Salvare website. There are custom filters making it easy to navigate finding the dogs eligible for water rescue, mountain or wilderness rescue, and disaster rescue/individual tracking along with pie charts for a quick and easy visual marker on the available breeds. Next to the pie chart is a geological map that will show the location of the animal selected.

The filters are predesigned with the provided list from Grazioso Salvare on what animals they prefer for each rescue attempt in order to help filter the lists making it easier to find the perfect match. The graph below helps the filters preselect based on age, sex, and breed of dog.



One of the main challenges I faced was getting the code to properly take in from my CRUD file which was named animalShelter1 as I had utilized the same code in project 1 as I had in Project 2. I was running into issues with it recognizing call backs and that was causing some errors to run. However I was able to fix it by tweaking the code and realizing I had accidentally mistyped some of the words by capitalizing the wrong letters which made it so they did not match.