**MongoDB CRUD with Python and Dash Framework**

**Purpose**

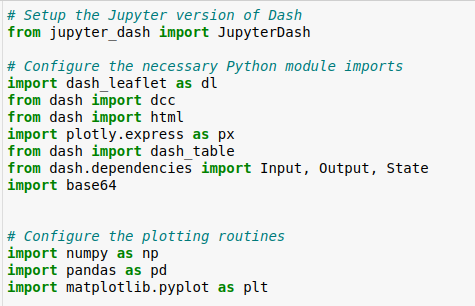
The purpose of this project is to implement python and mongodb database for C-R-U-D (create, read, update, delete) using python function for querying Mongo databases to communicate with the dash framework

**Motivation**

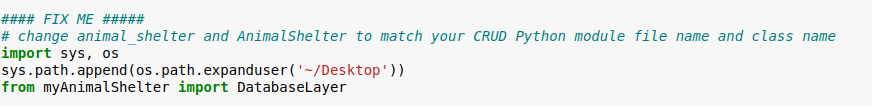
This project exists to give user to be familiar with dash framework as well as learning how to build the data frame for mongodb query using python function.

**Getting Started**

This project will require numerous of python library such as pandas, jupyter\_dash, dash, plotly…



Import the finished python file from project one in module 5



**Installation**

MongoDB package

Python package

Jupyter Notebook for Linux

The script required pymongo package for Python application, dash for dash framework, and pandas which is required by Plotly Express

python -m pip install pymongo

python –m pip install dash

python –m pip install pandas

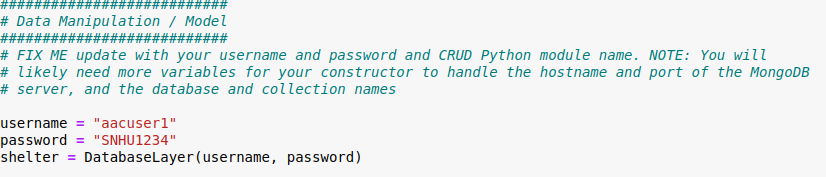
**Usage**

**Code Example: (for mongodb credentials in the imported python file)**

****

**Tests Script:** (assume user named their python file.py in ~/Desktop directory of a Linux system

1. For this particular project, user will hard code their credentials for the purpose of testing and instantiate python class object



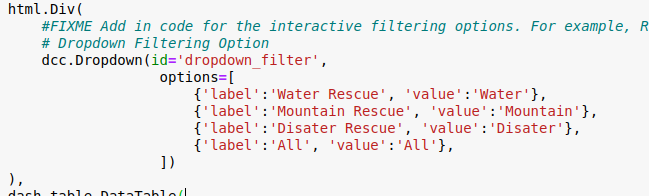
2. Database ‘aac’ is successfully accessed by user created credentials, now we move on to set the filtering option for later used



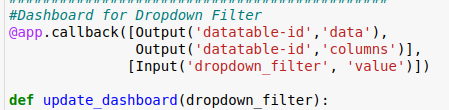
3. Project Two require user choose either radio buttons, or drop down, or checkbox option for the interactive dashboard

4. Using @app.callback for a complete interaction between components and controller. For example:

Dropdown Menu:



Calling @app.callback and set the filtering option using if/else statement. The id has to be matched.



4. Pie Chart and Map is created from app.layout

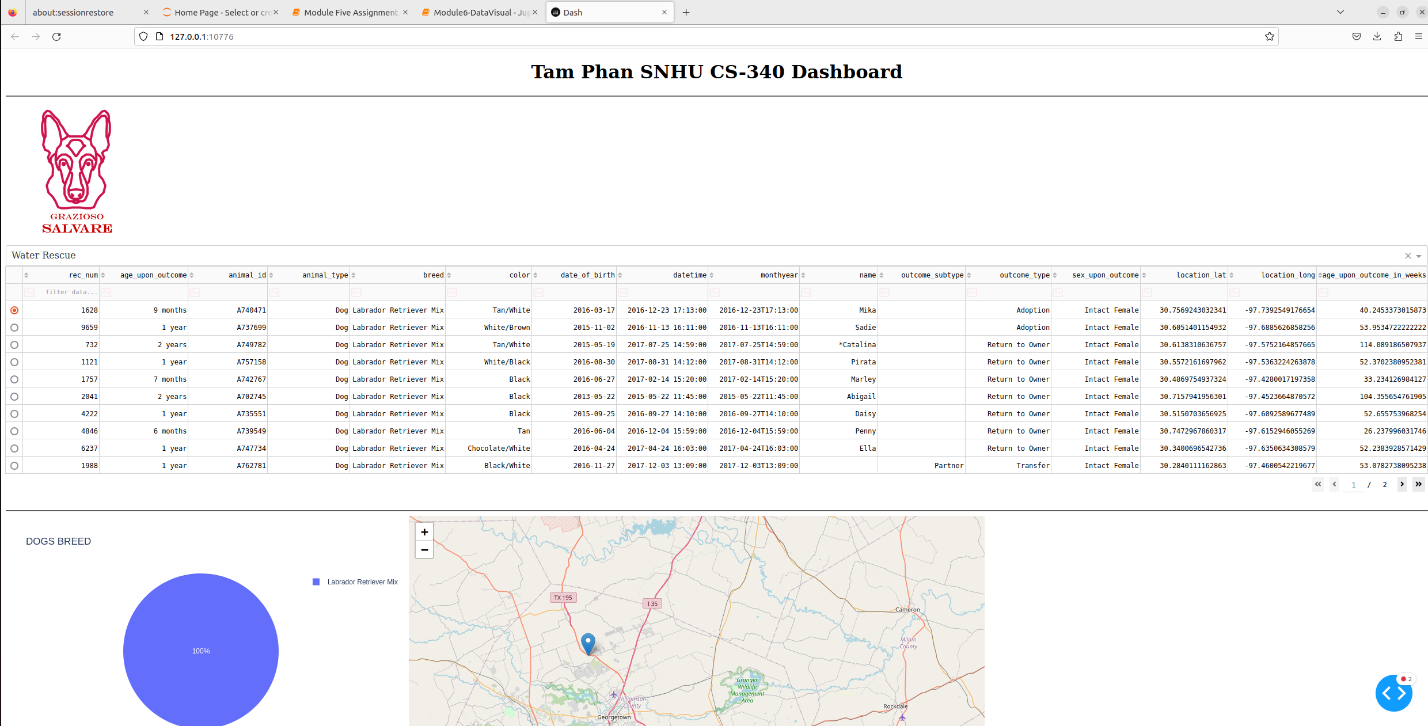
We have graph-id for pie chart, and map-id for map



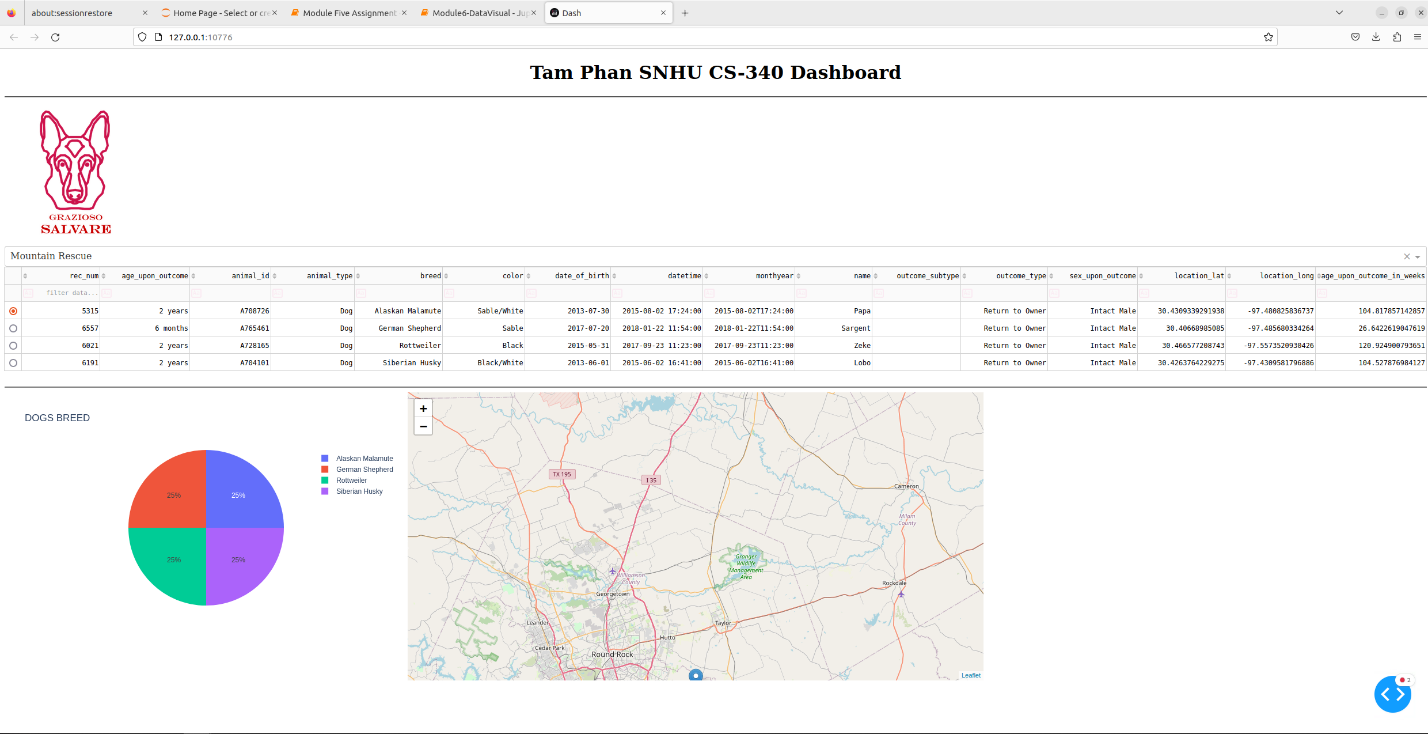
5. Also need to use @app.callback and create python function to generate pie chart and map

**7.Final ouput:**

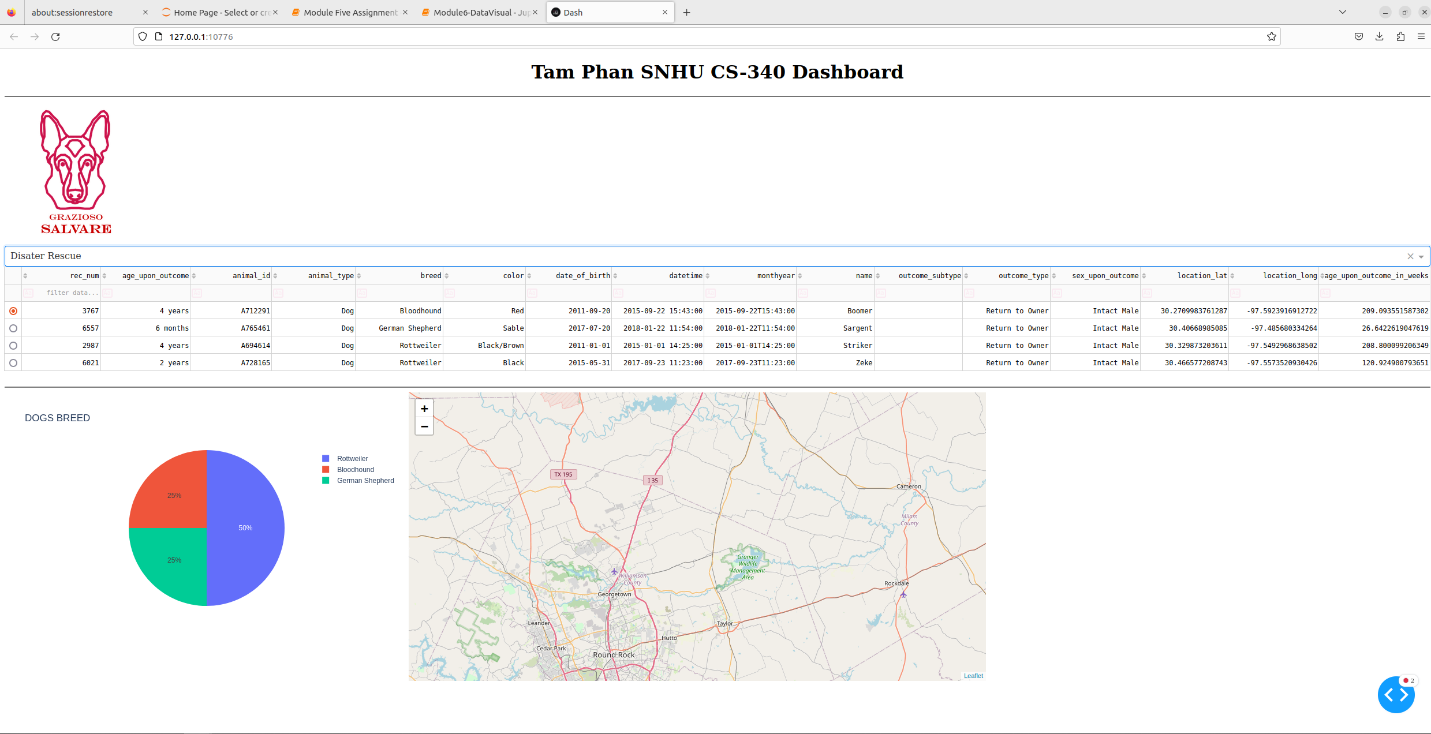
*Dropdown Menu with Water Rescue Option*



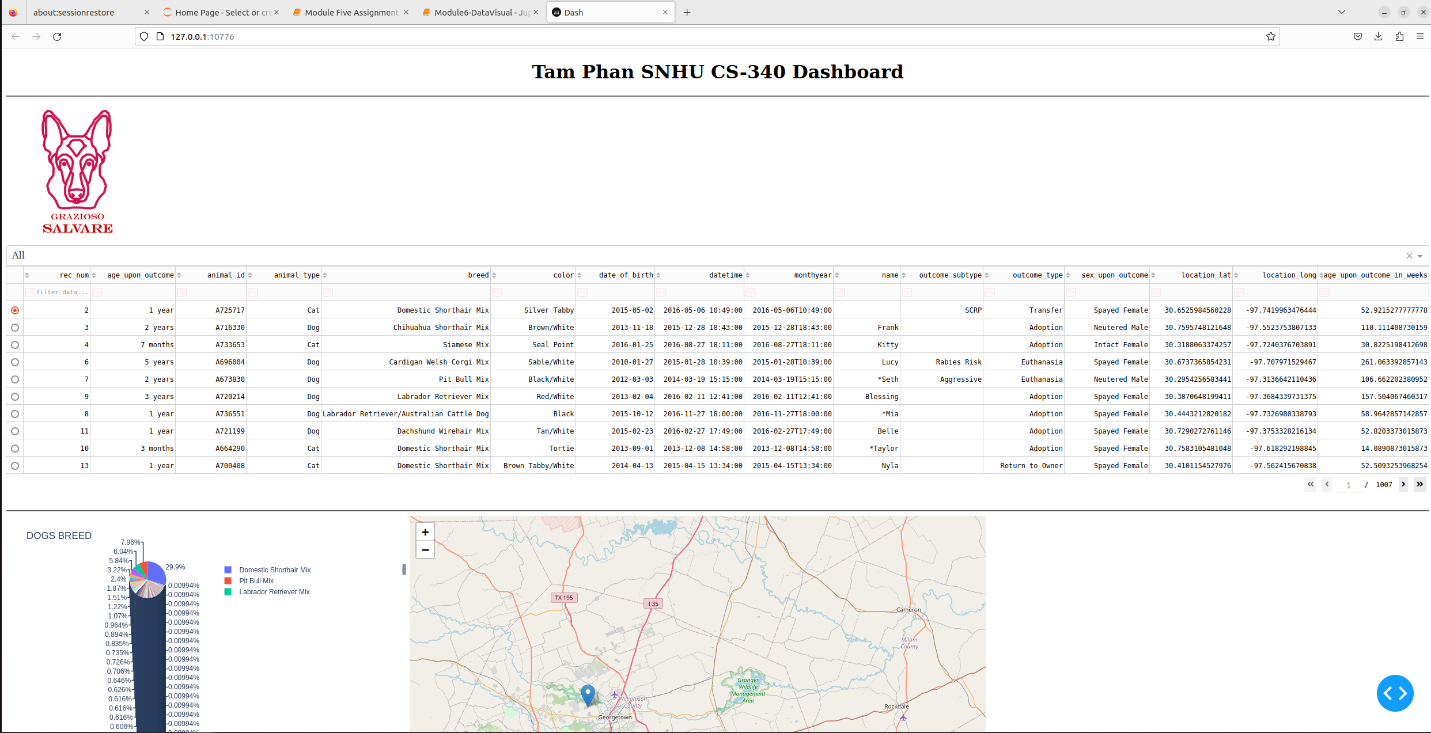
*Dropdown Menu with Mountain Option*

****

*Dropdown Menu with Disater Option*

****

*Dropdown Menu with All Option for dog breed (since there are too many dog breeds, the pie chart is splitting up too many slices, hence, the unattractive display)*

****

**Challenge:**

Since we are using VM, the constantly lagging and delaying is unavoidable; therefore, it’s giving me a bunch of errors when running the code, mostly typo, but it’s very time-consuming to google the error and to backtrack every single line since they are all interconnected from the python file, to the Jupyter dash board; even though, I learn a lot from video instruction and material. Also, if you are messed up on week 4 or week 5, then, you can’t move on to Project One and Two.

**Contact:**

Tam