# Project 2 README

## About the Project/Project Title

This project is to create a client facing website that interacts with a database using user credentials. The website would have a data chart that has all the information of the database. It will also have a pie chart and geolocation that gives some insight into the data being presented.

## Motivation

This project exists for a project in class. On another note, this can be used to interact with a database. It is also modular so it can be used in any code or with any database. This makes it valuable in multiple ways.

## Getting Started

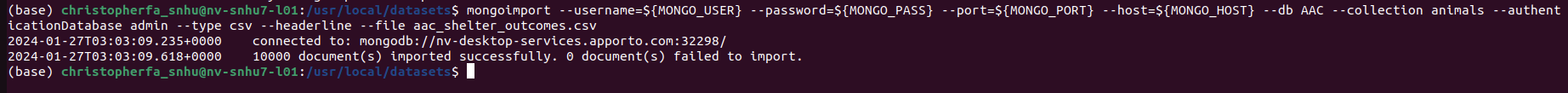
To begin you would need to create or have a database to work with within Mongo DB. From there, creating users within the system to access the database would be next. After these steps have been completed, you would develop a python code to interact with Mongo DB and your database that holds a class for your CRUD functions. From there you would develop the website client in Jupyter. This would entail creating the layout of the website and various functions/callbacks needed to create the data table and graphs. That should be everything you need to get this running locally.

## Installation

You would need Mongo DB to house the database and manipulate the data, a database(s) in Mongo DB to have objects to work with, code editor and Python to create the code to interact with the data. To test your code, you would need Jupyter Notebooks to create an ipybn file to create the website and design the data table and graphs.

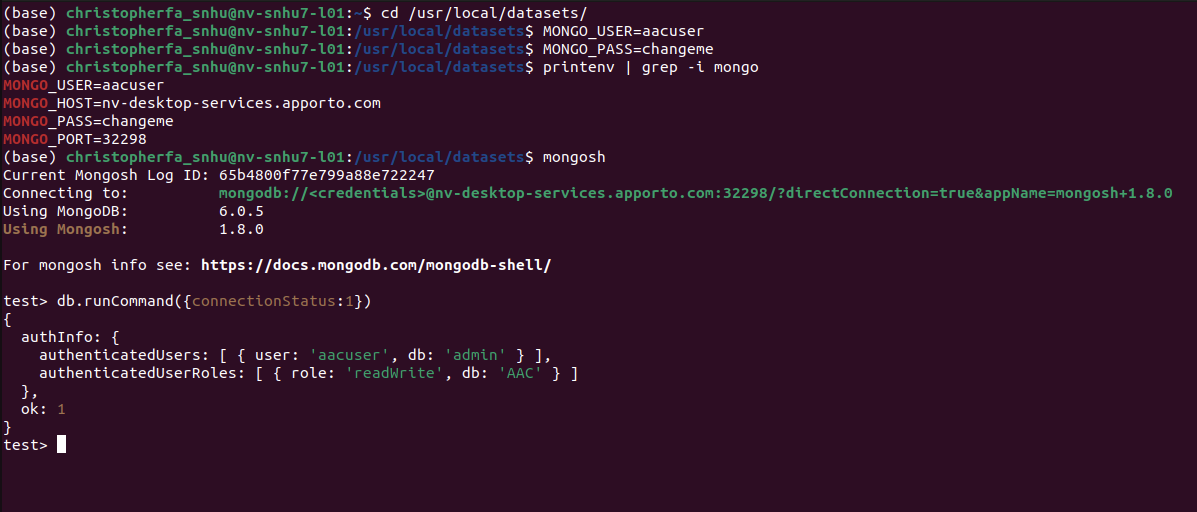
## Usage

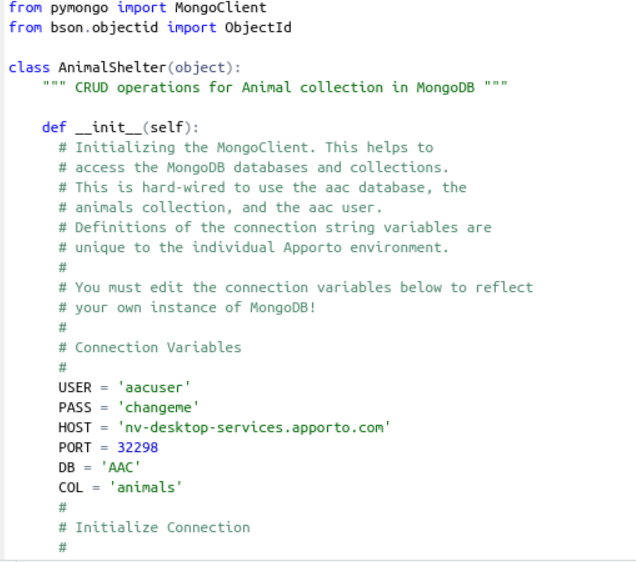
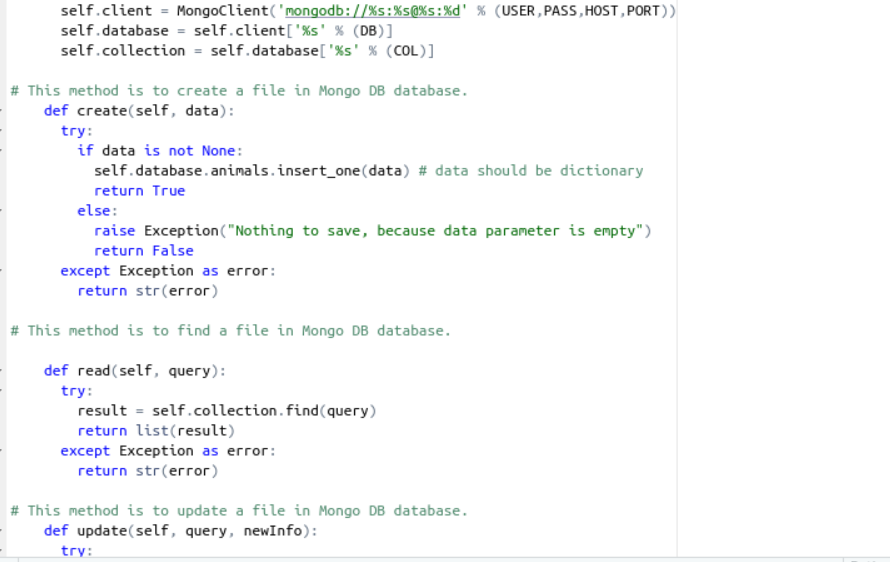
Import of Database

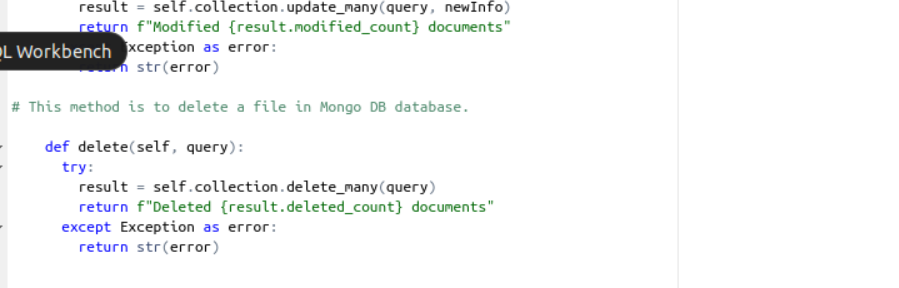


Create User



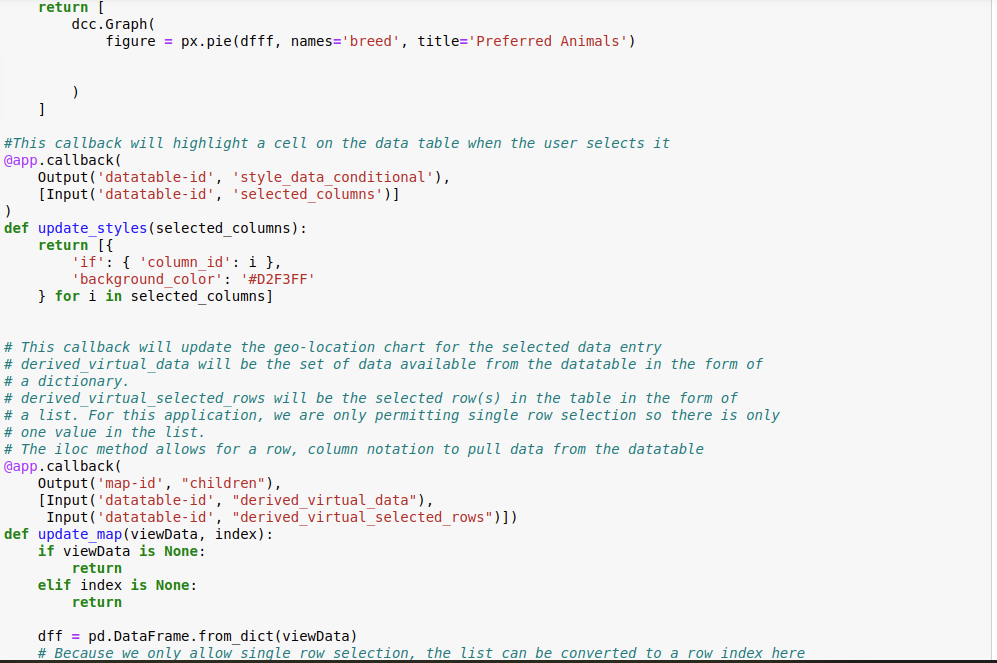
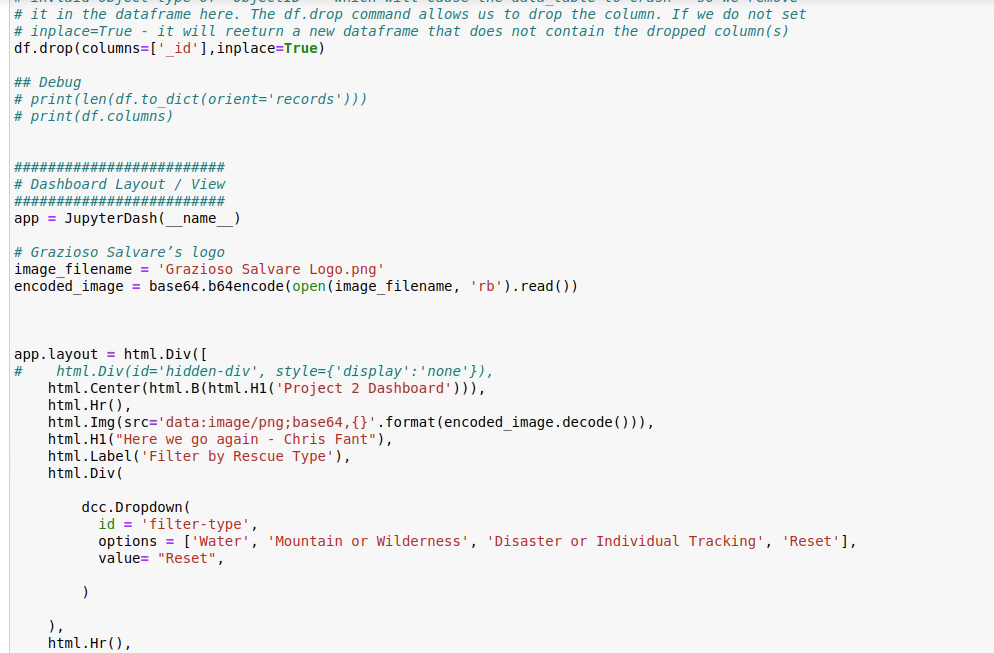
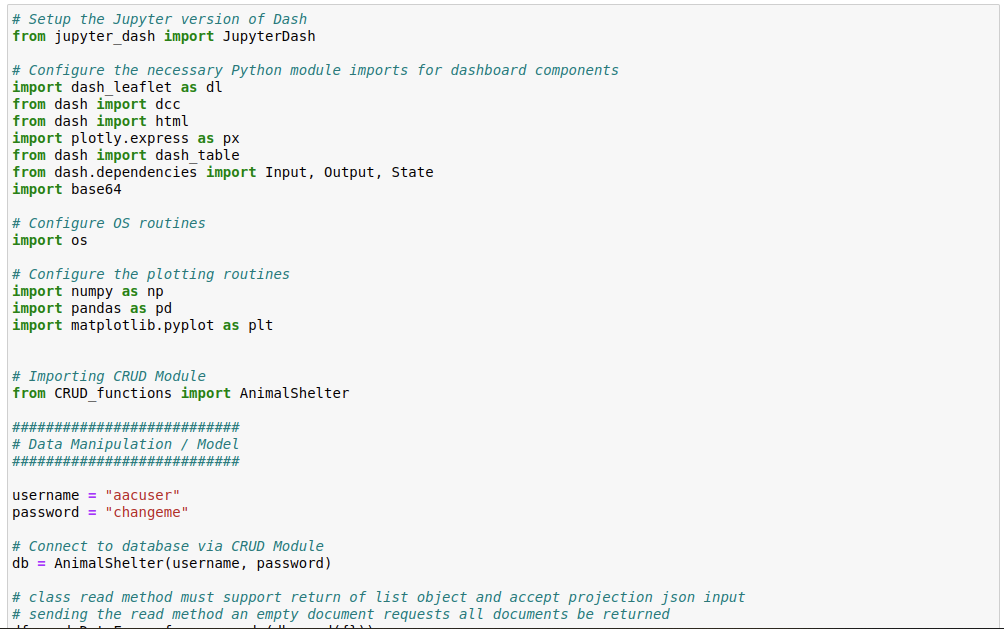
Authenticate User

**Code Example****



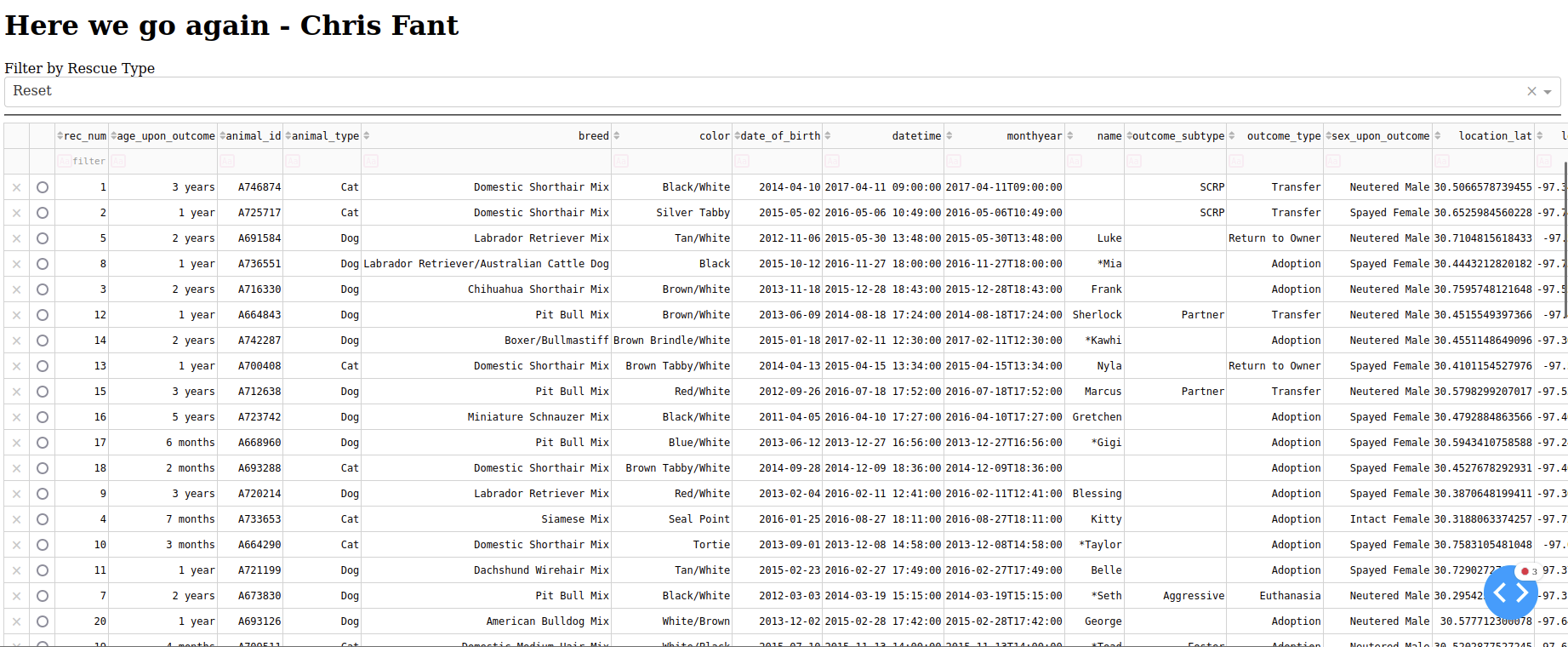
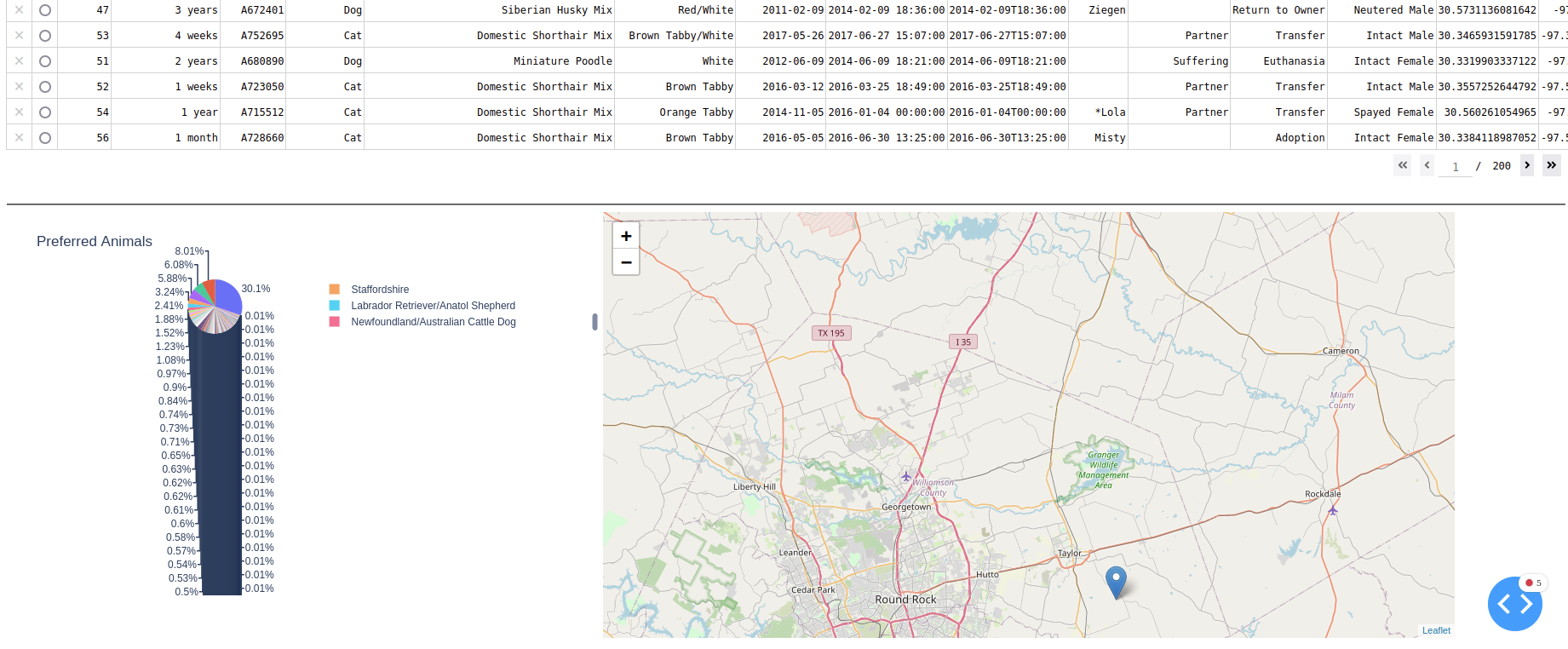
### Website Creation

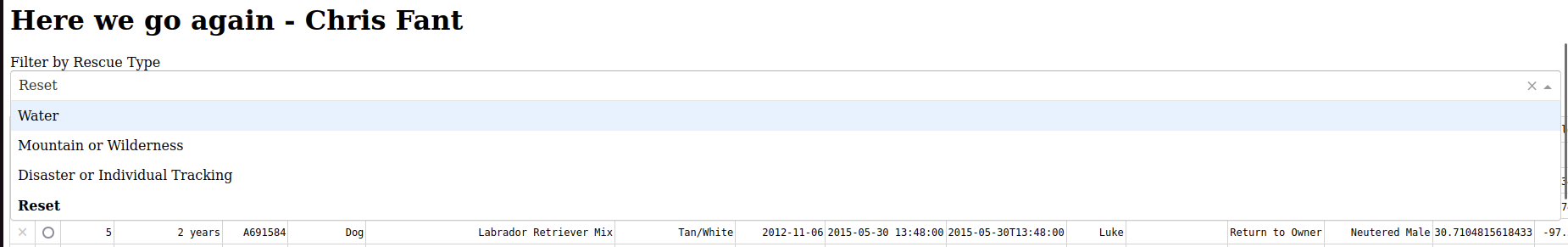
To create the website, you can use Jupyter Notebook code that interacts with your python code. In this, you can bring in Dash and Plotly to help create the features that are wanted. This is the easier way to do it as it integrates really well.

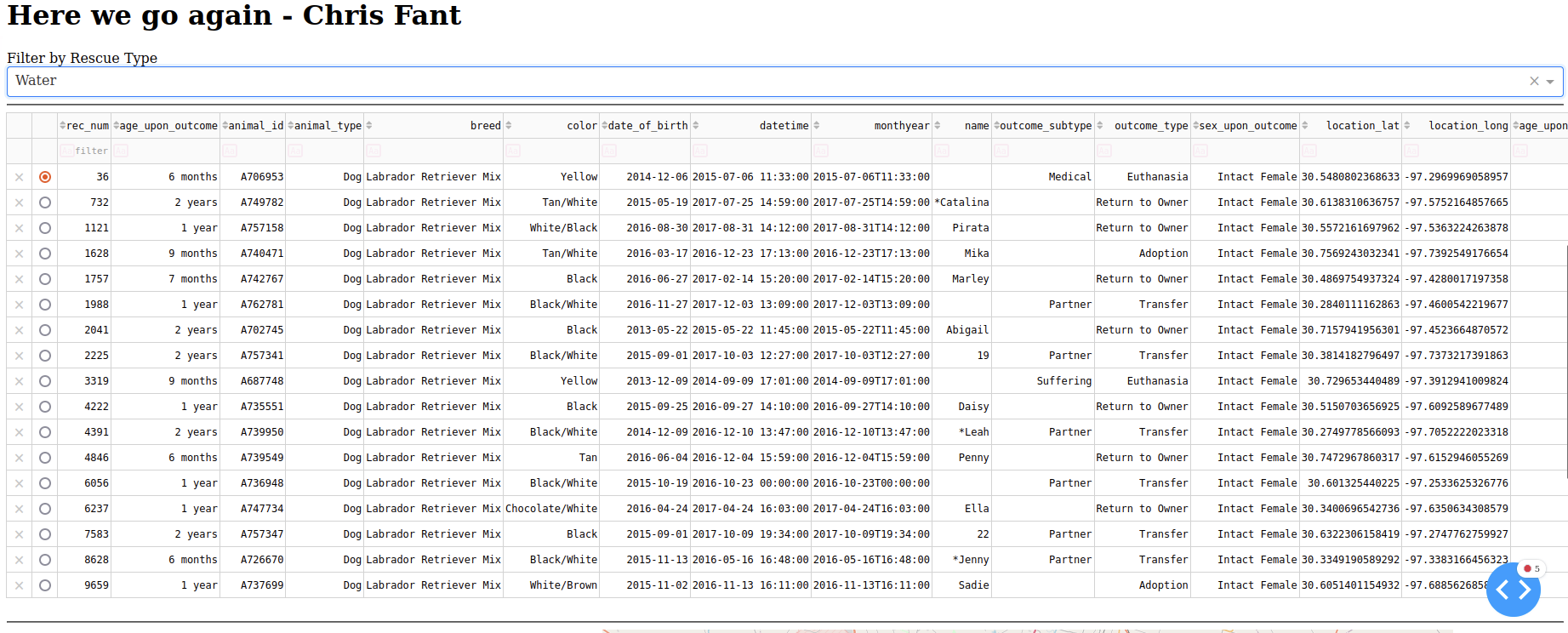
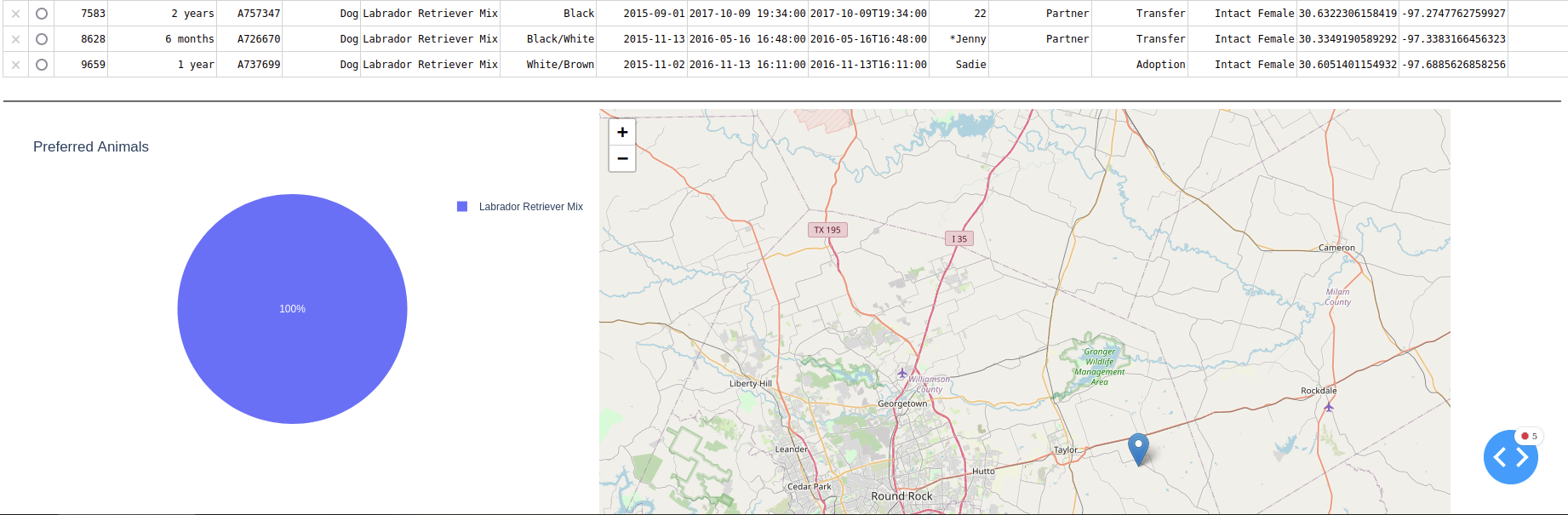


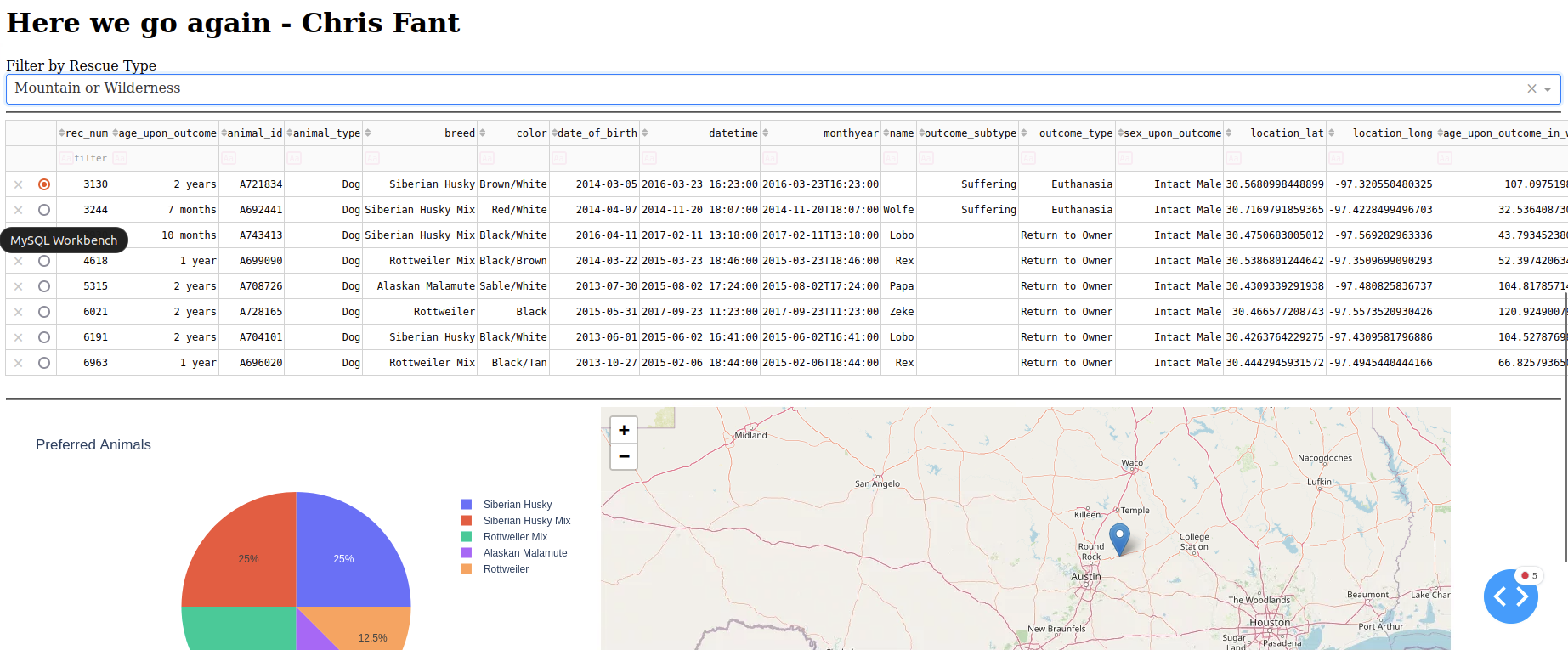
**Website Product**

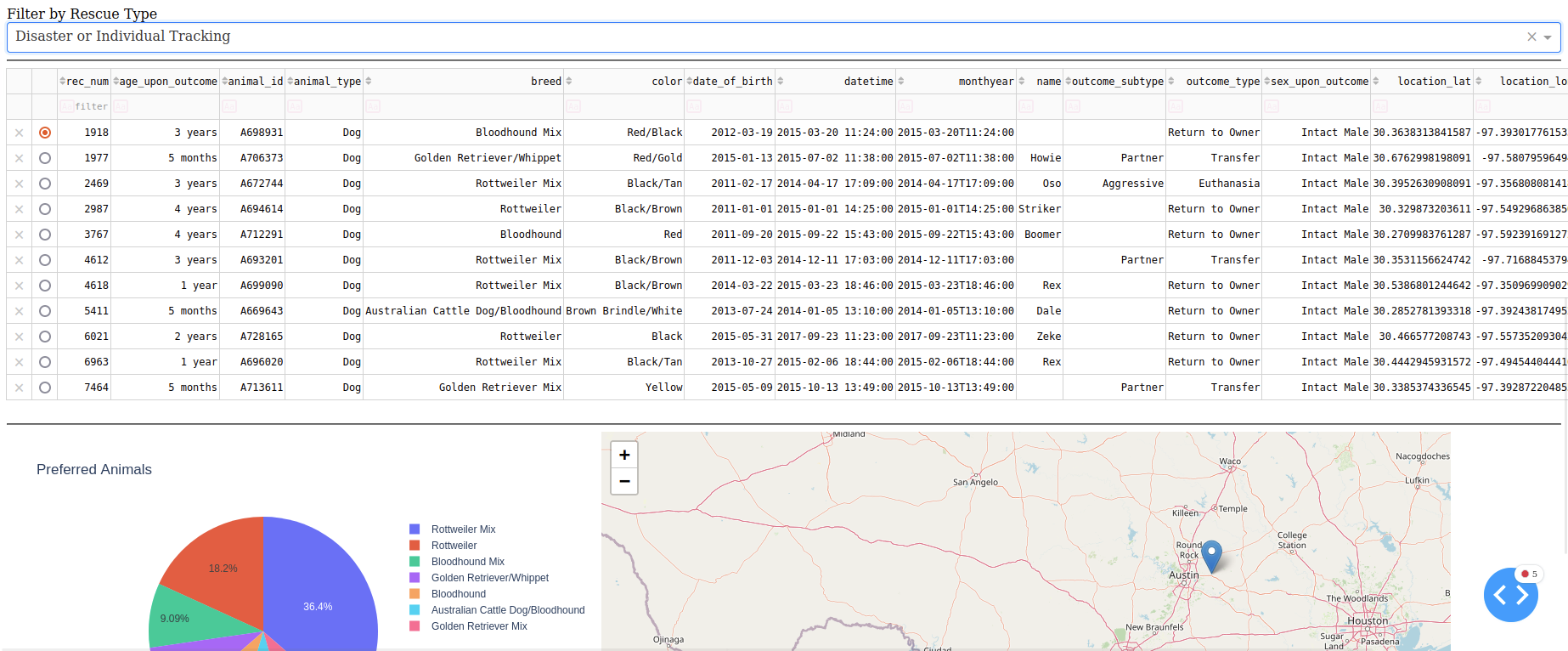
Logo and Header****

Unique Identifier and Data Table****Pie Chart and Geo Location

Dropdown Menu****

Water Filter – Data Table****

Mountain/Wilderness Filter

Disaster/Tracking Filter

## Roadmap/Features

With the completed code we have a completed product. There are some cosmetic things that could be added in the future to make the website more appealing. For example, in the initial pie chart it is a little small and hard to read. I would remedy this by having the data not take up so much room and try to be encased in the chart itself or hide some of the smaller percentages. Other future options could be to add more filters for those with special needs or mental illness.

**Resources**

Here is a list of resources used to create the code.

* Dash Core Components
  + <https://dash.plotly.com/dash-core-components>
* Dash Plotly
  + <https://dash.plotly.com/datatable>
* Plotly Pie Charts
  + <https://plotly.com/python/pie-charts/>

## Mastering MonogDB Testbook

## <https://web-p-ebscohost-com.ezproxy.snhu.edu/ehost/ebookviewer/ebook?sid=7e6bf675-2f45-4344-9b3d-387083aa5ddf%40redis&ppid=Page-__-87&vid=0&format=EK>

## Contact

Your name:

Chris Fant