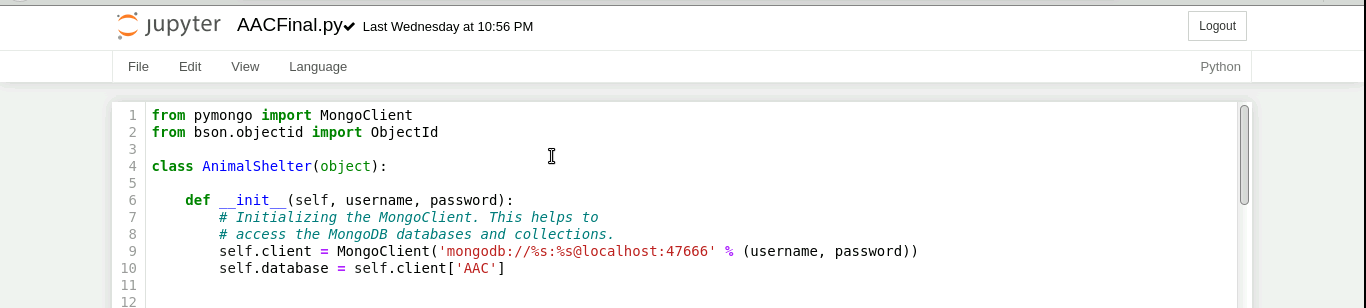
# Gravioso Salvare Rescue Animal ReadMe

## About the Project/Project Title

This project contains two files. The first file creates a class Animalshelter. This class will use user authentication to access the animal collection of the AAC database. Once the authentication is verified, the Create method is used to create a new set of data to add to the collection. The Read method will be used to read the animal collection to search for criteria based on a specific argument to find an entry.



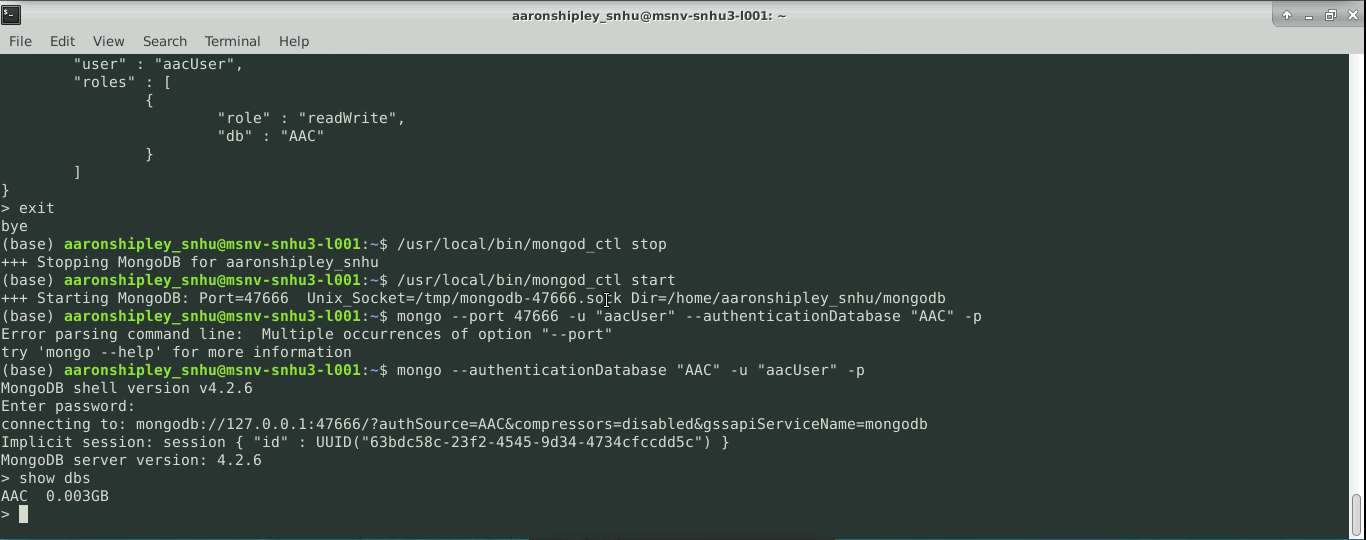
The second file is an interactive dashboard that displays an unfiltered data table retrieved from the AAC database, of animals collection of the Austin Animal Center outcomes CSV file. The dashboard also displays a geo-location map displaying the location of the first dog in the list. There is also a pie chart breaking down the breeds of dogs within it. The dashboard will also include an interactive Dropdown menu to allow the user to filter the type of rescue dog the table will display. Along with the filter for the table, the filter will apply to the geo-location map and pie chart.

## Motivation

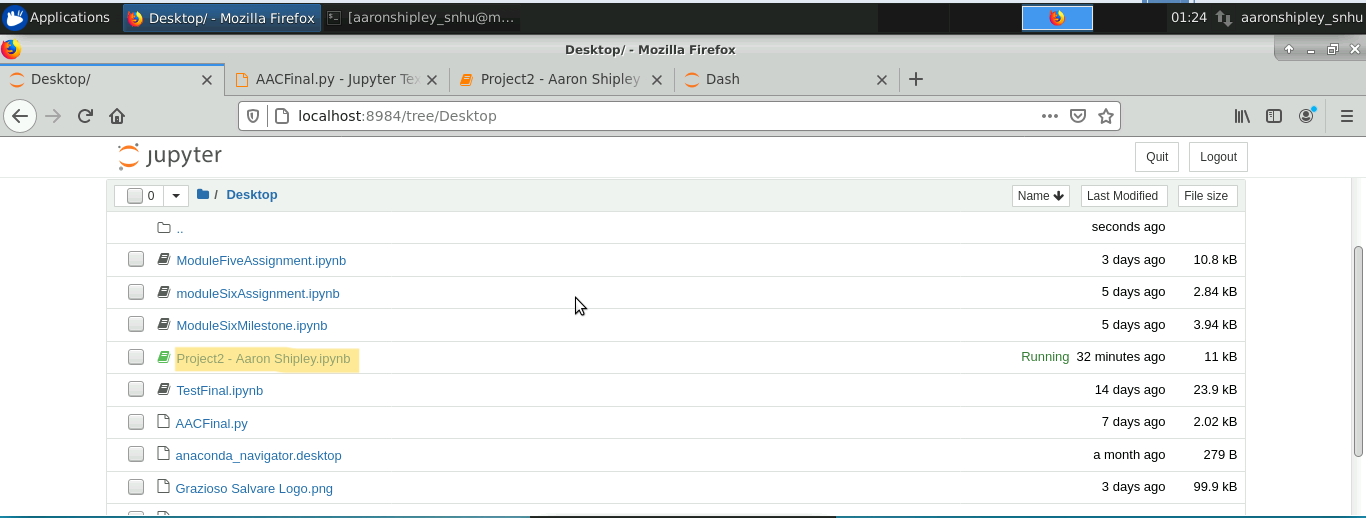
*The project exists for the purpose of the user to successfully look at the provided data table and then filter the table, chart and map accordingly to their specific need.*

## Getting Started

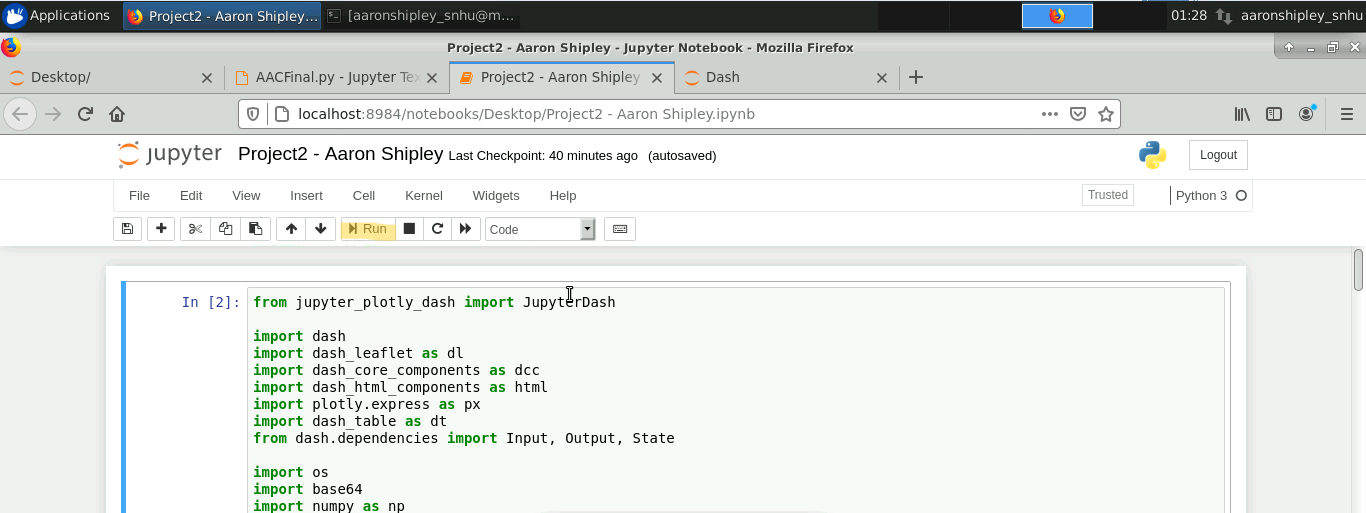
1. *Start the Linux shell and then open mongo via authentication*

**

1. *Upload IPYNB file to Jupyter Notebook and open.*

**

1. *Run IPYNB file in Jupyter*

**

## Installation

*pymongo must successfully be installed on the machine.*

*MongoDB must be installed on the machine.*

*Using Jupyter Notebook will allow users to test the script line by line or the entire script at once.*

*Local datasets will need to be uploaded and imported to MongoDB as well.*

*AAC access and admin access to MongoDB will be required in order to create, read, update, or delete data from the collection as well.*

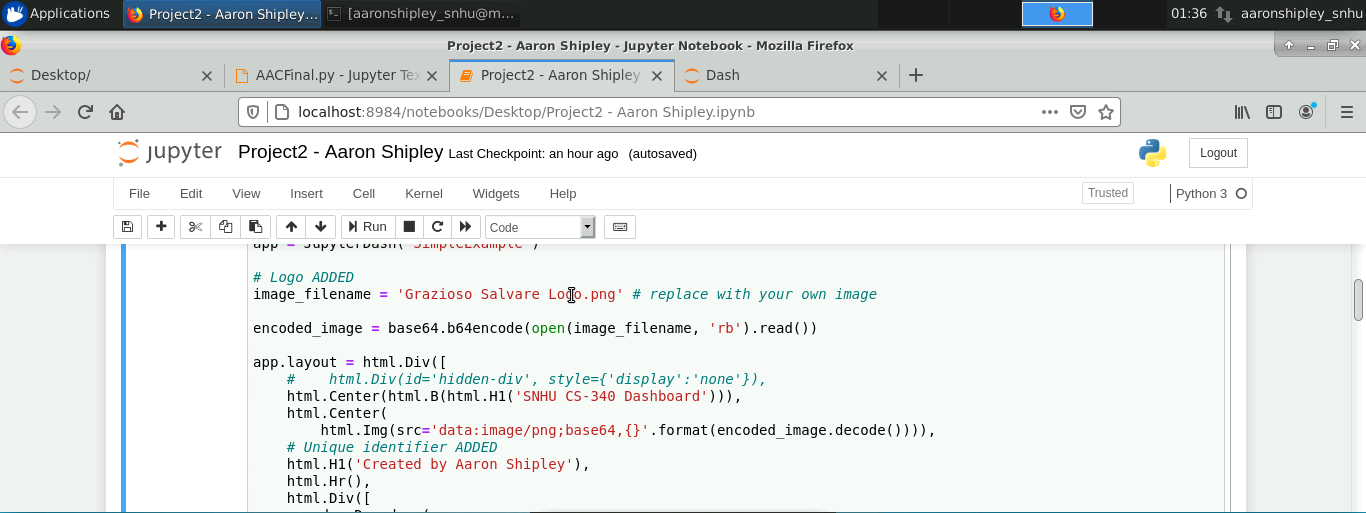
*Proper imports of Dash, NumPy, Pandas, and Plotly need to be done*

## Usage

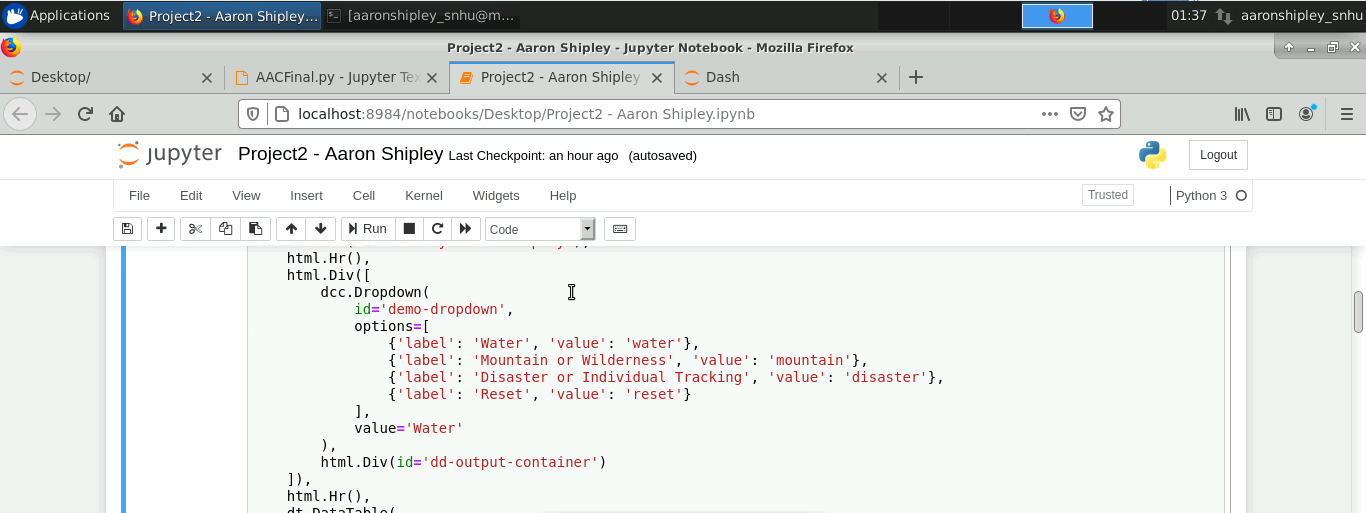
*The Dashboard will allow the user to look through the Austin Animal Shelter data table and then filter the table, pie chart, and map accordingly. The user will be able to select the proper dog for training based on the criteria set forth in the data table. The user will be able to locate the dog via the geo-location map to see its proximity to the user. The user can use the pie chart to see the relation of breeds within each filter of the animal shelter CSV file.*

### Code Example

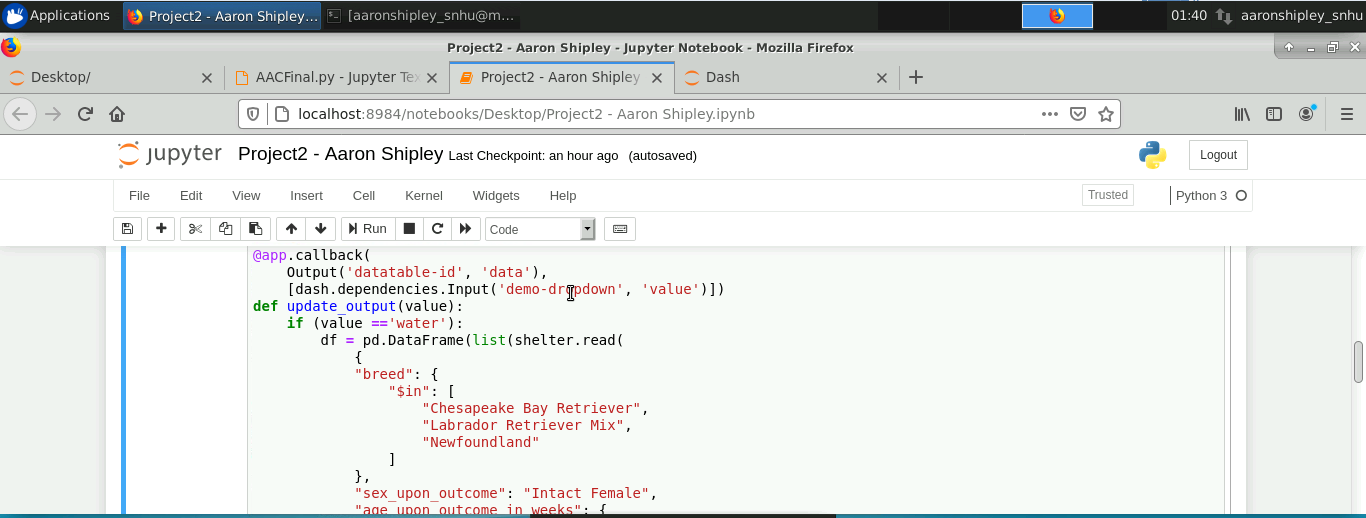
* Designate File name for logo to upload
* Load image into application dashboard
* Begin app layout to display Title menu, Logo, and unique Header



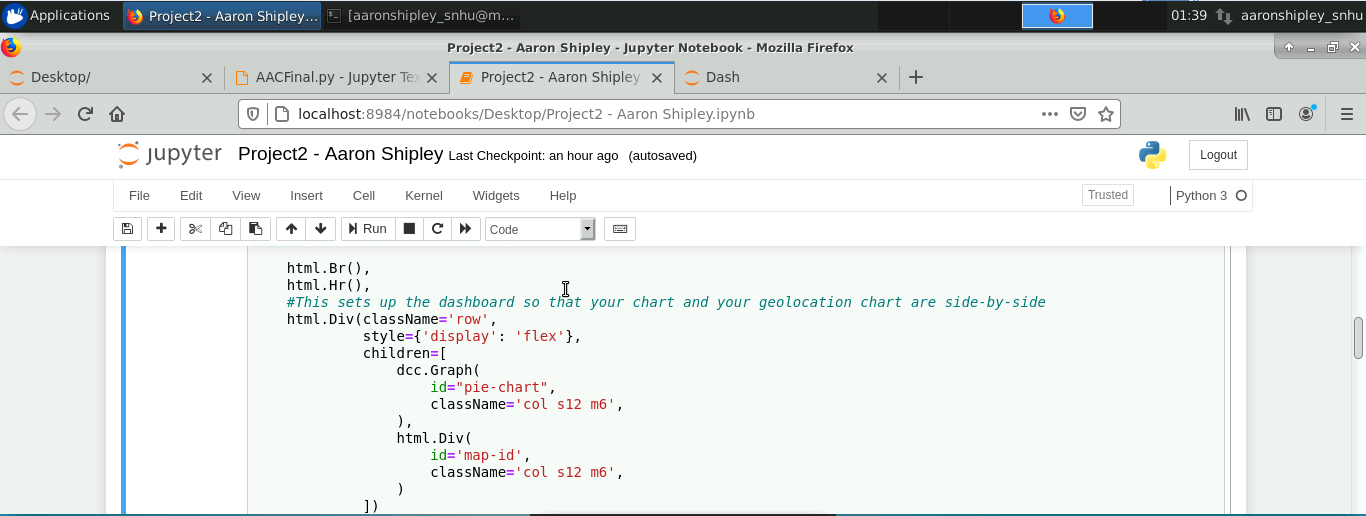
* Within app log create interactive option for drop down menu
* Set values for dropdown to display
* Call upon output to create within dashboard



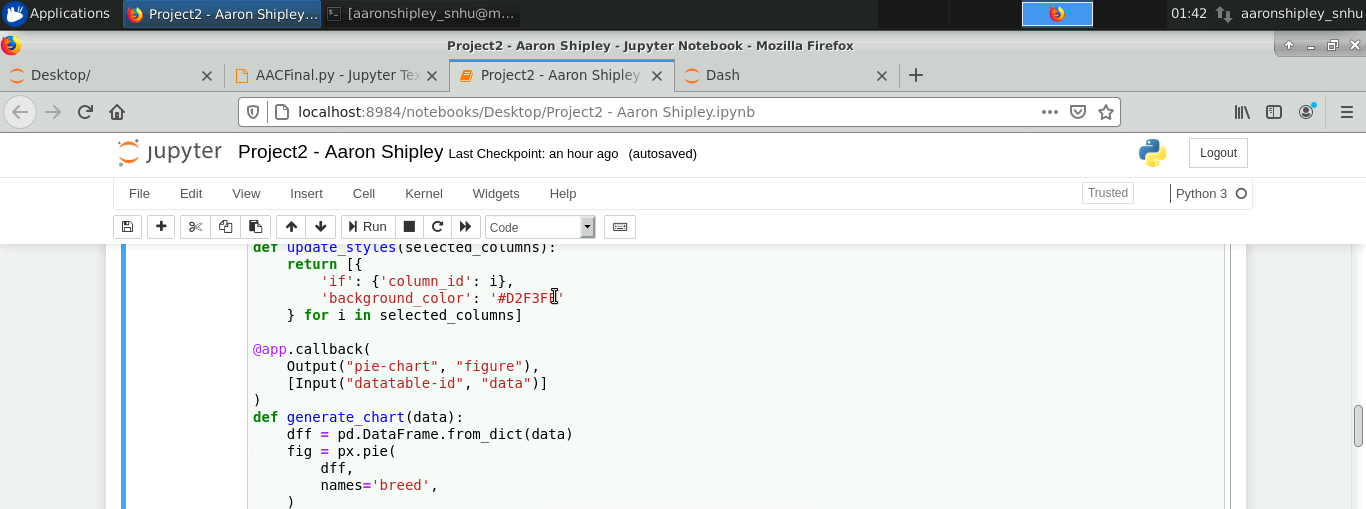
* Callback to create datatable
* Call CSV data to display in datatable
* Filter results to call from “water” dropdown value



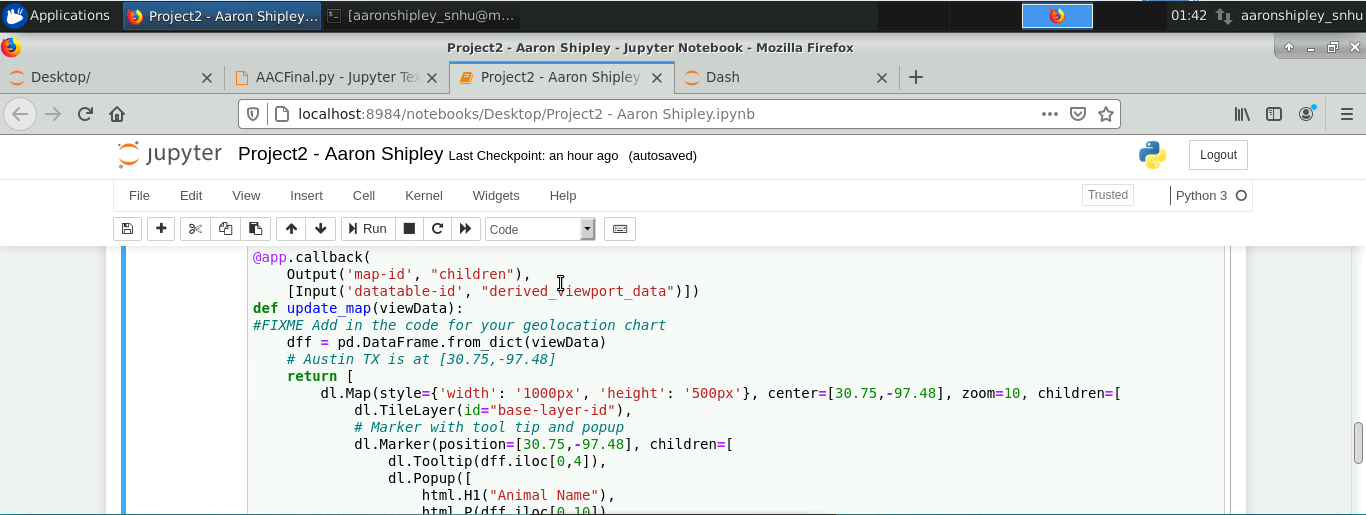
* Map and graph layout
* Map and graph will be side by side in dashboard output



* Call upon pie chart
* Call data from CSV to generate in chart
* Set data by “breed” value for chart



* Call map
* Geolocation code for real-time GPS data
* Set map container with style and specifications
* Call upon location via Lat and Lon calls
* Display data via “Animal Name” value in dataset

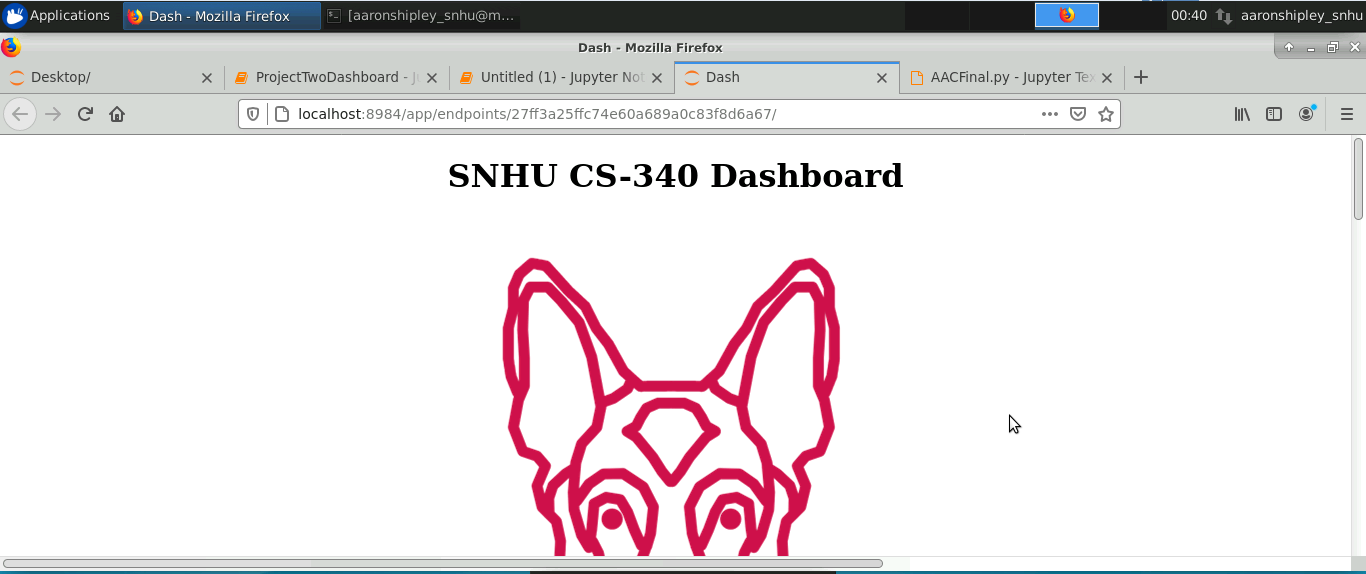


### Tests

Testing of the dashboard required the running of the code. When no syntactical errors were present, the run is tested for proper output of the dashboard. When the dashboard loaded, ensuring all header, display and logo outputs were present. The dropdown menu was tested for functionality, along with the filters contained within. The filters were tested to ensure expected output within the data table, ma, and chart were achieved.

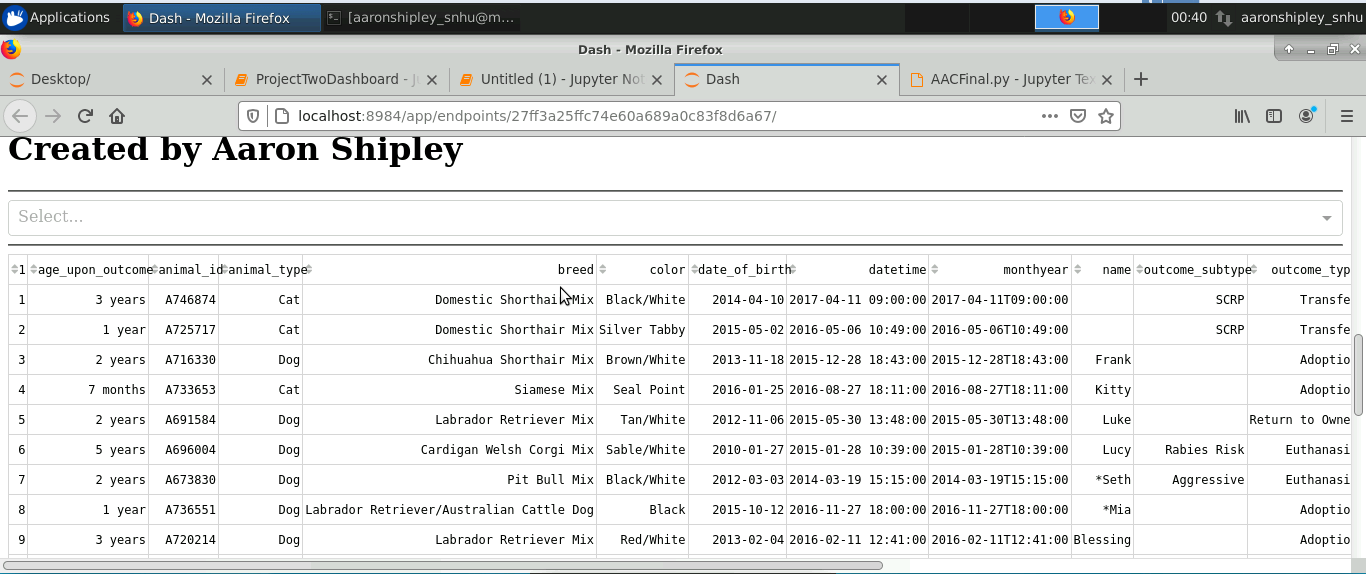
### Screenshots

Gravioso Logo w/ Unique Header

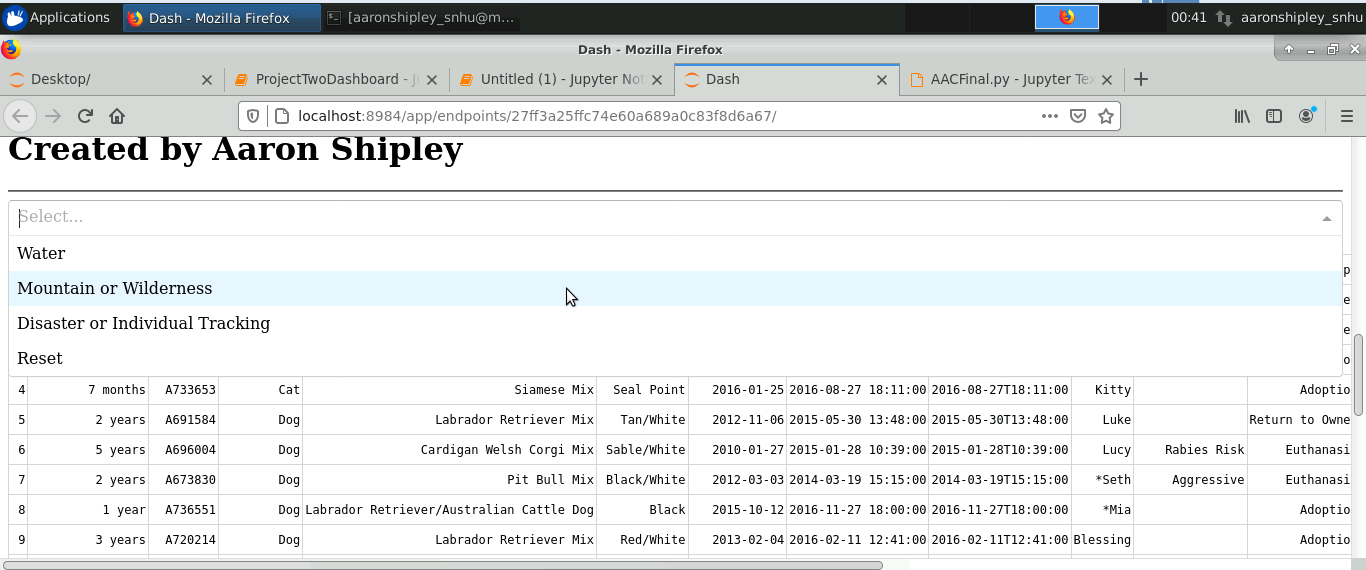


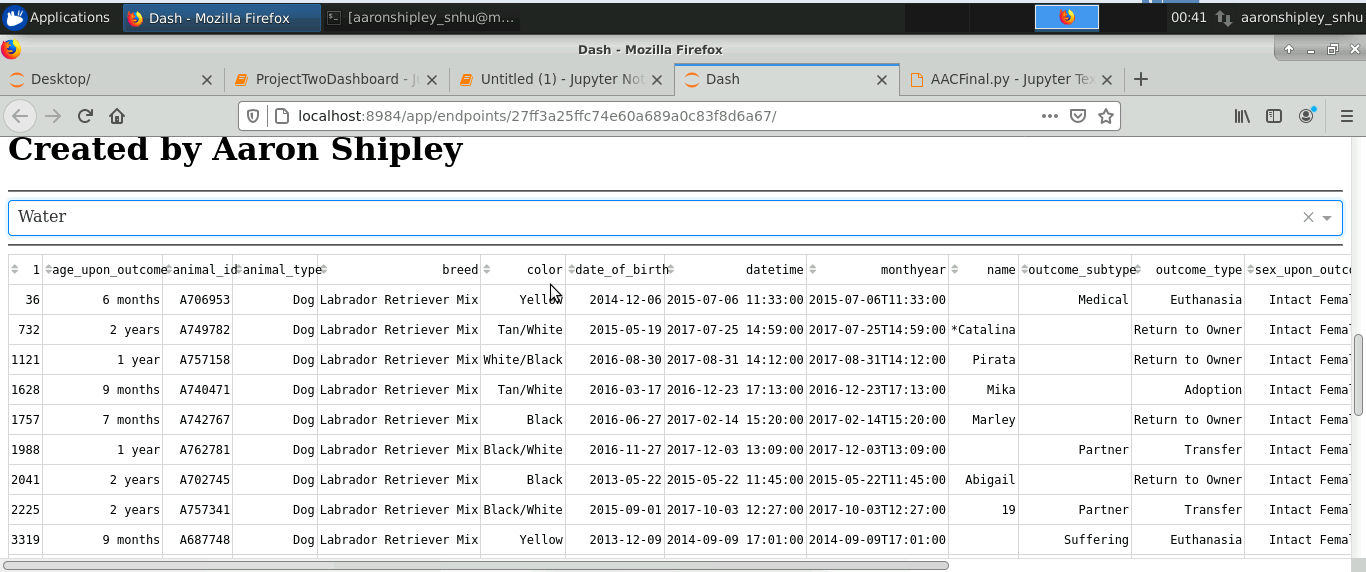


Data Table

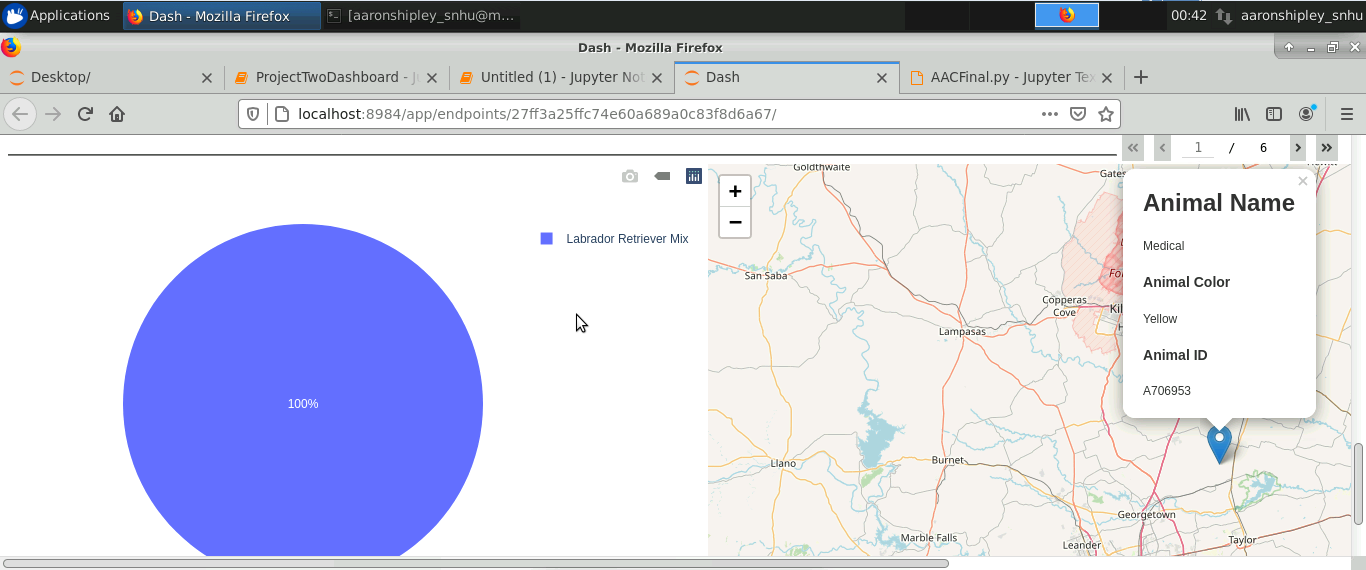


Dropdown Menu

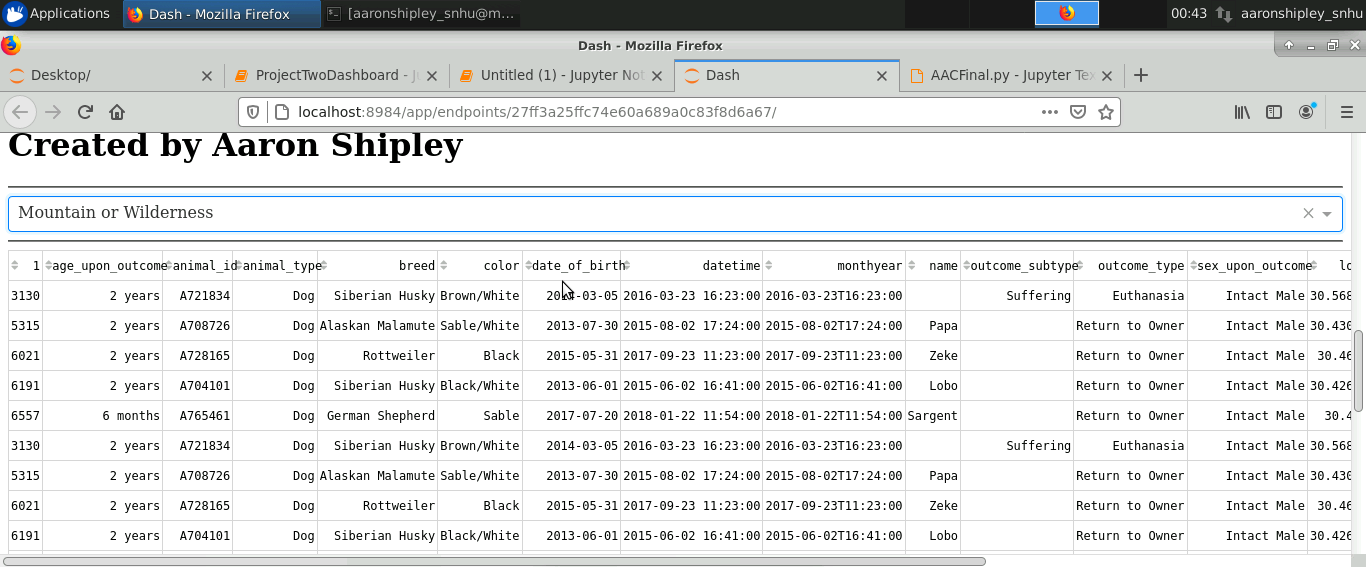


Water Filter

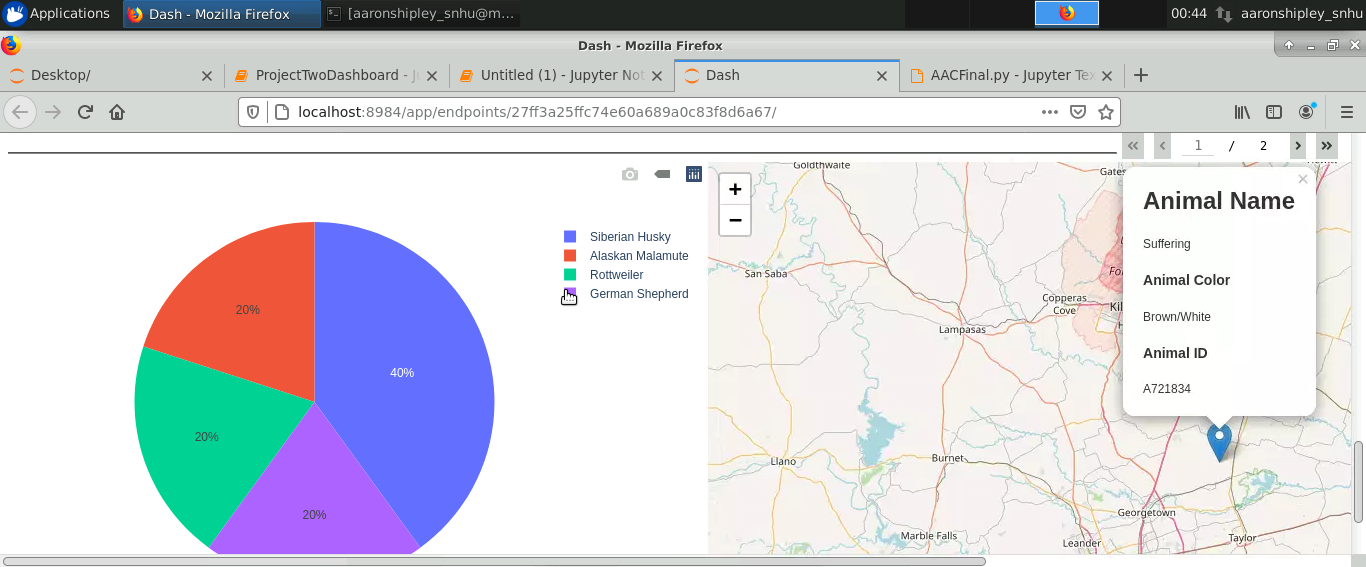
Water Map and Chart



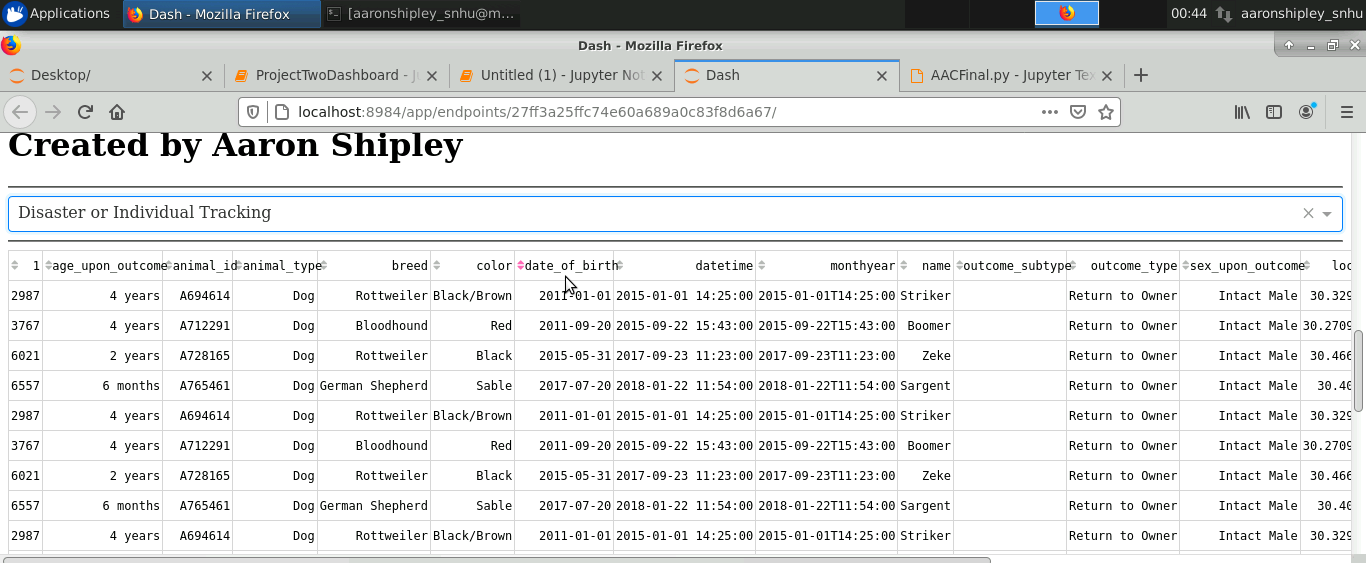
Mountain Data Table



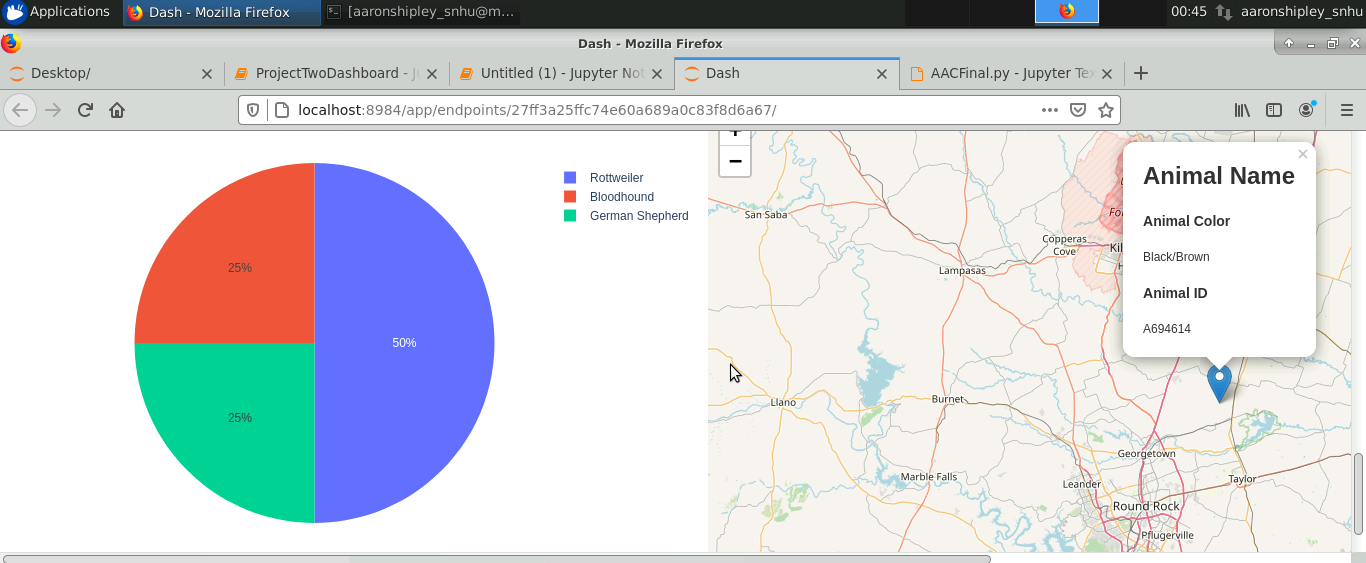
Mountain Map and Chart



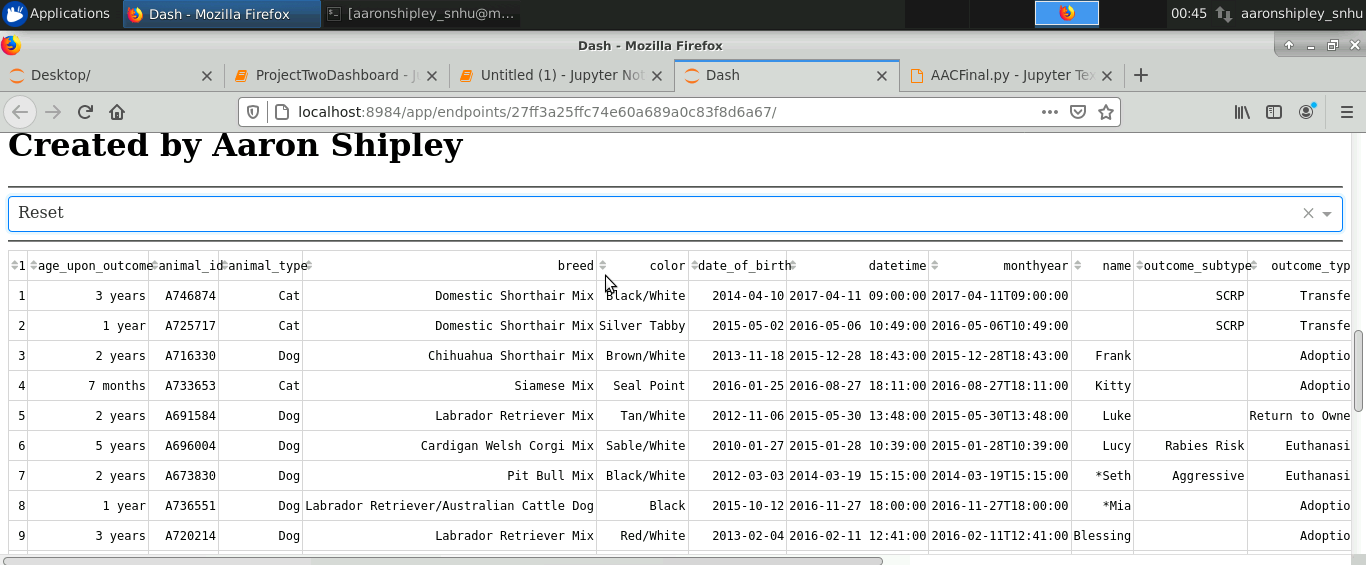
Disaster Table



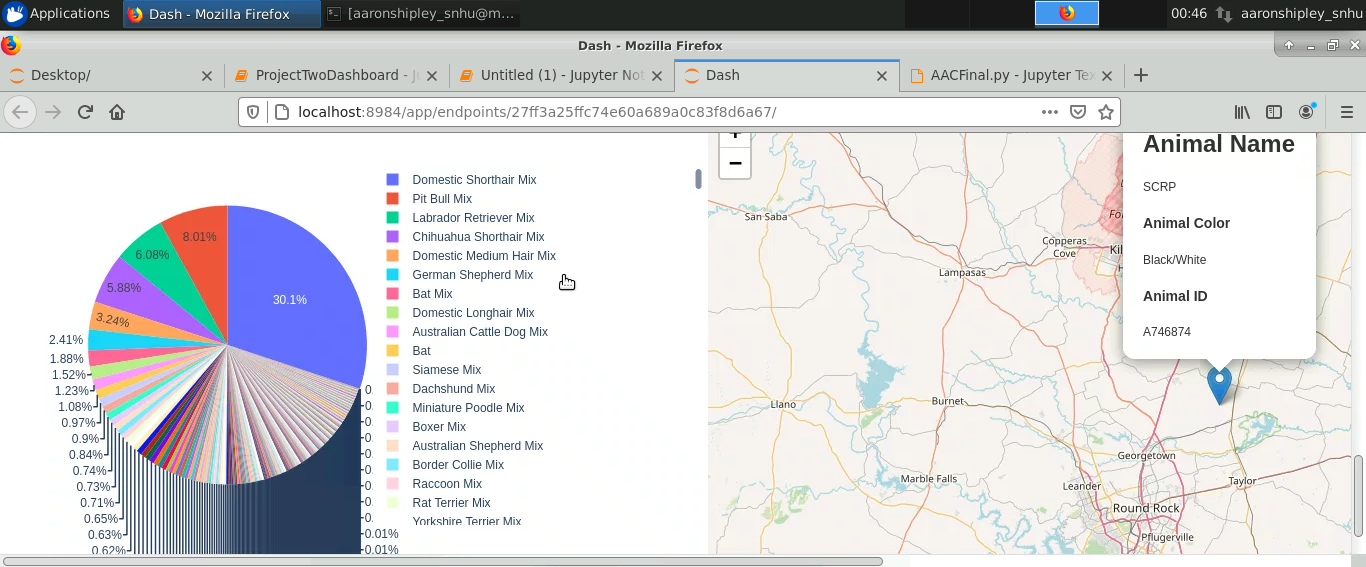
Disaster Map and Chart



Reset Table



Reset Map and Chart



## Contact

Your name: Aaron Shipley