# README Grazioso Salvare Dog Identification

## About the Project/Project Title

The project is an open source software application that accesses an animal shelter database to search for specific animals. This software application is a full stack which includes a database and a client-facing web application dashboard for user access. The software will display a data table, a pie graph, and a geolocation map of each selected animal.

## Motivation

This project is requested from the Grazioso Salvre company to search for specific dogs at five animal shelters in Austin, TX. The dogs will meet the criteria for training in specialized search-and-rescue tasks.

## Getting Started

Setting up the project locally, a user will need access to the ‘AAC’ database in MongoDB which administration can give authorization and the CRUD.py file needs to be downloaded to local machine. The Jupyter Notebook program is used to execute the Python file and the methods created to interact with the MongoDB database also the Dash Framework will need to be imported to Jupyter Notebook.

## Installation

MongoDB can be downloaded from <https://www.mongodb.com/try/download/community>

Python Language IDE can be downloaded from <https://www.python.org/downloads/>

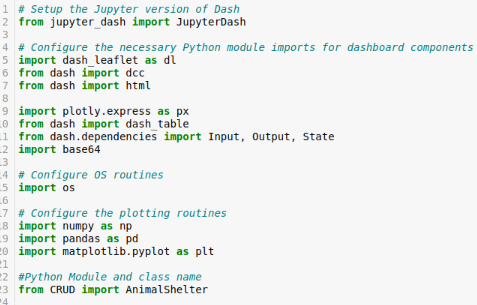
Jupyter Notebook directions can be found at<https://jupyter.org/install>

Dash Framework directions to install can be found at <https://dash.plotly.com/installation>

## Usage*:*

### Code Example

Imports necessary to compile the data table, the graph, the map, manipulate the data, and to display the company logo.

**

### Tests

Running tests are done in the Jupiter Notebook by importing the AnimalShelter class from the CRUD.py file along with imports from Dash, Ploty, and Pandas (shown in image above). Query filters are set to radio item buttons, the data table is set up for selection and display

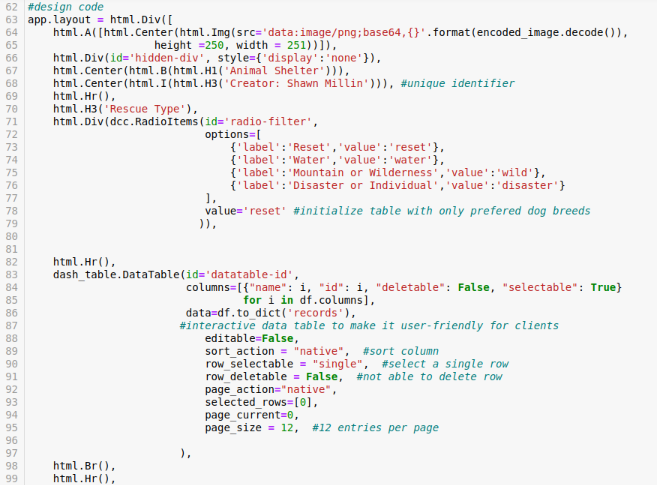
### Screenshots

Setting up the Grazioso Salvre company logo



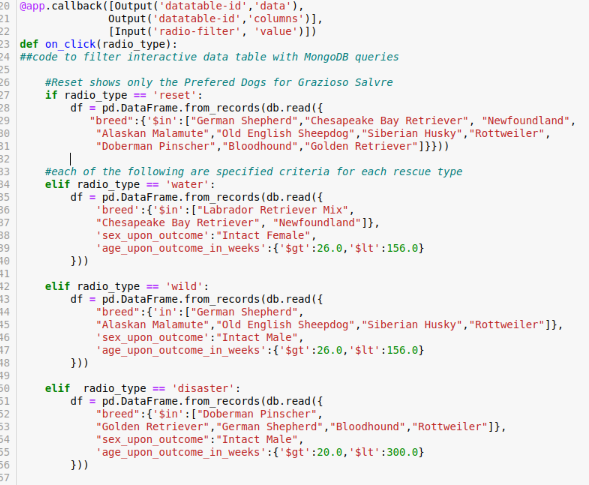
Setting up the layout with company logo at the top center, next is the title “Animal Shelter, then the creator's name.

Radio buttons are set up with “RadioItems” and the data table is set up with “DataTable”

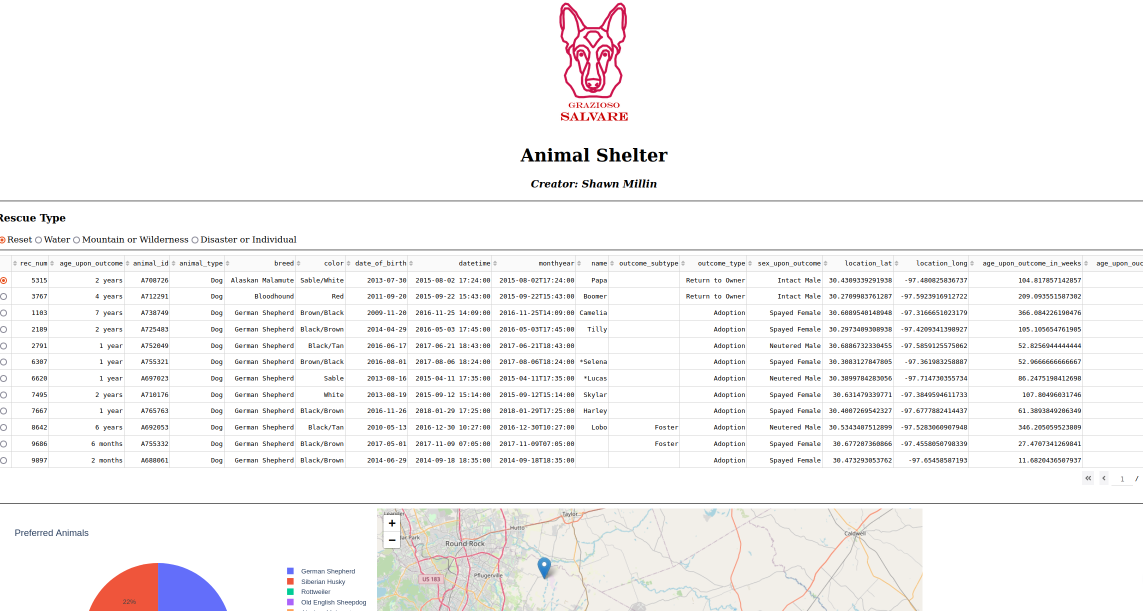


The callback is set for the radio buttons. Each radio button queries from CRUD.py file which accesses the MongoDB database of AAC in the animals collection.

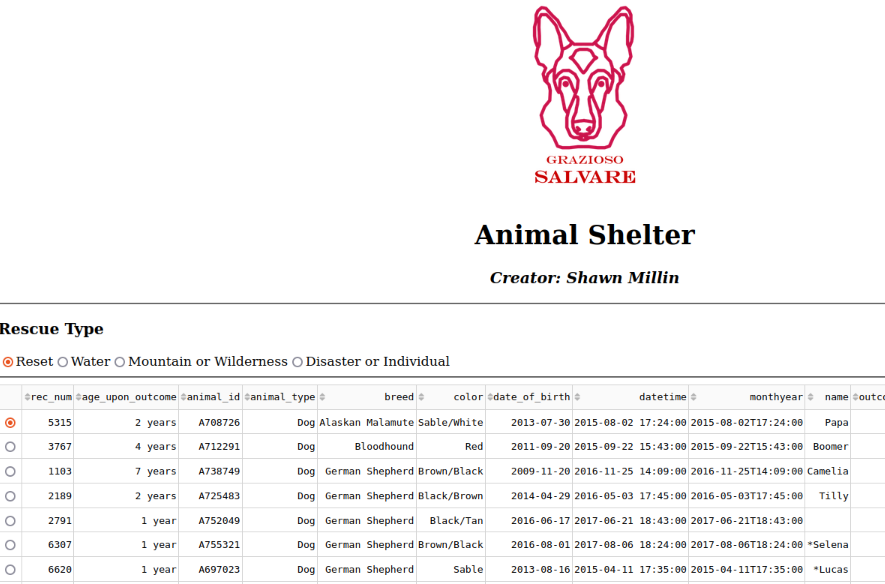
Each radio button has specific queries for dogs that meet the specified rescue type



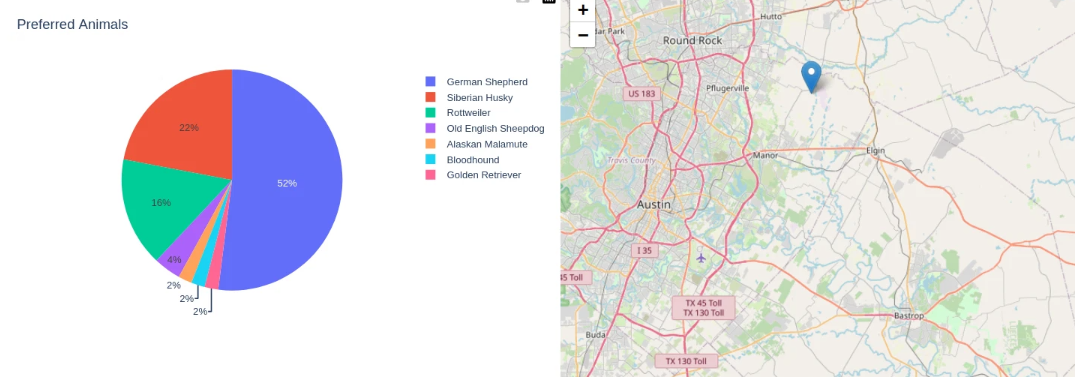
Initial screen when IPNY file is run.



Close up of initial data table, initializes with reset and preferred dog breeds and the first row selected

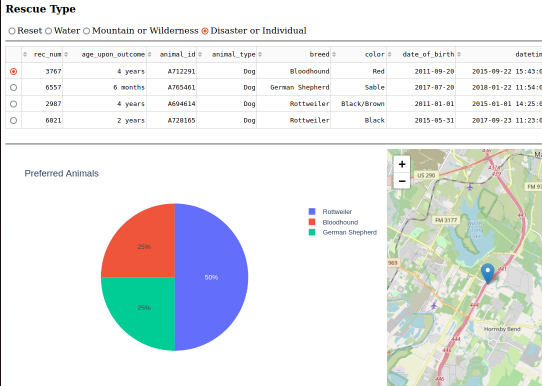


close up of graph and map initial screen

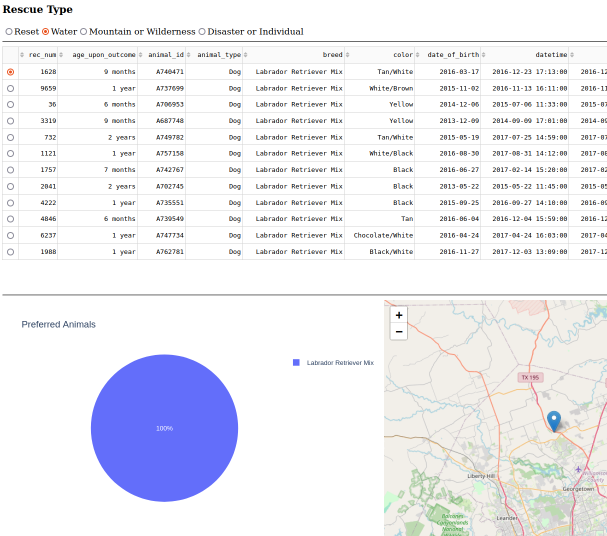


Selection of radio buttons changes the graph to only those breeds found matching the specific criteria in the animal shelter.

Disaster or Individual Rescue



Water rescue



Mountain or wilderness (did not have any dogs that matched queries)



## Contact:

## Shawn Millin