# CS 340 README Project 2

**Michelle Hitchcock**

**About the Project/Project Title**

The purpose of this project is to develop a dashboard for Grazioso Salvare, an innovative rescue-animal training company. This dashboard will allow for viewing, sorting, and filtering data on shelter animals Included in this is a series of radio buttons that will filter for, and display, good candidates for search-and-rescue training based on Grazioso Salvare criteria.

**Motivation**

The motivation behind this project is to increase and grow the company’s ability to find dogs that are available for adoption and identify dogs that are good candidates for search-and-rescue training.

**Getting Started**

**Installation**

For reproducing the project, steps include installing MongoDB, installing/importing the (csv) data, user authentication setup, and installing the CRUD and Query python module.

Software that will be needed to run this program:

MongoDB, Jupyter Notebook locally installed and running correctly along with the ACC database uploaded to MongoDB as well as the AnimalShelter CRUD/Query module and the ProjectTwoDashboard.ipynb

## Usage

This dashboard functions using **Three** main sections. this dashboard is Utilizing MongoDB for database handling, Python/Pymongo for filtering and CRUD functionality, and Dash for dashboard displays.

Data Manipulation / Model

This section establishes the username, password, and functions which the data will be retrieved from.

Dashboard Layout / View

This section establishes the layout of the dashboard, calls the logo image, implements the native filters/multi-sorting/ and page size

Interaction Between Components / Controller

This section establishes the functions of the dashboard elements.

### Code Example

Show what the library does as concisely as possible. Developers should be able to figure out how your project solves their problem by looking at the code example. Make sure that your code is short and concise.

In order to identify the correct candidates for search and rescue training, functions were created in the AnimalShelter.py file for each of the filter types.

This allows for a cleaner main dashboard file. as the functions can then be called in the data manipulation/model section as:

dfwr = pd.DataFrame.from\_records(shelter.RescueTypeWater())

dfmwr = pd.DataFrame.from\_records(shelter.RescueTypeMountainWilderness())

dfdit = pd.DataFrame.from\_records(shelter.RescueTypeDisasterTracking())

For the displayed graph the value that the chart is displaying is the number of the breed. this is achieved using the function assignment.

new\_df = dff.breed.value\_counts()

An important note is that the map will not automatically display, an animal must first be selected. This is because of the check

odff = df if viewData is None else pd.DataFrame.from\_dict(viewData)

The selection of an animal prompts the geolocation chart to appear.

### Tests

testing using the radio button:

-Disaster or Individual Tracking

Expected outcome:

Doberman Pinscher, German Shepherd, Golden Retriever, Bloodhound, Rottweiler

Intact Male

20 weeks to 300 weeks

updated Pi chart

-Mountain and Wilderness

Expected outcome:

German Shepherd, Alaskan Malamute, Old English Sheepdog, Siberian Husky, Rottweiler

Intact Male

26 weeks to 156 weeks

updated Pi chart

-Water

Expected outcome:

Labrador Retriever Mix, Chesapeake Bay Retriever, Newfoundland

Intact Female

26 weeks to 156 weeks

updated Pi chart

-Reset

Expected outcome:

Data to reset to no filters

Testing using individual selection:

Expected

updated geolocation

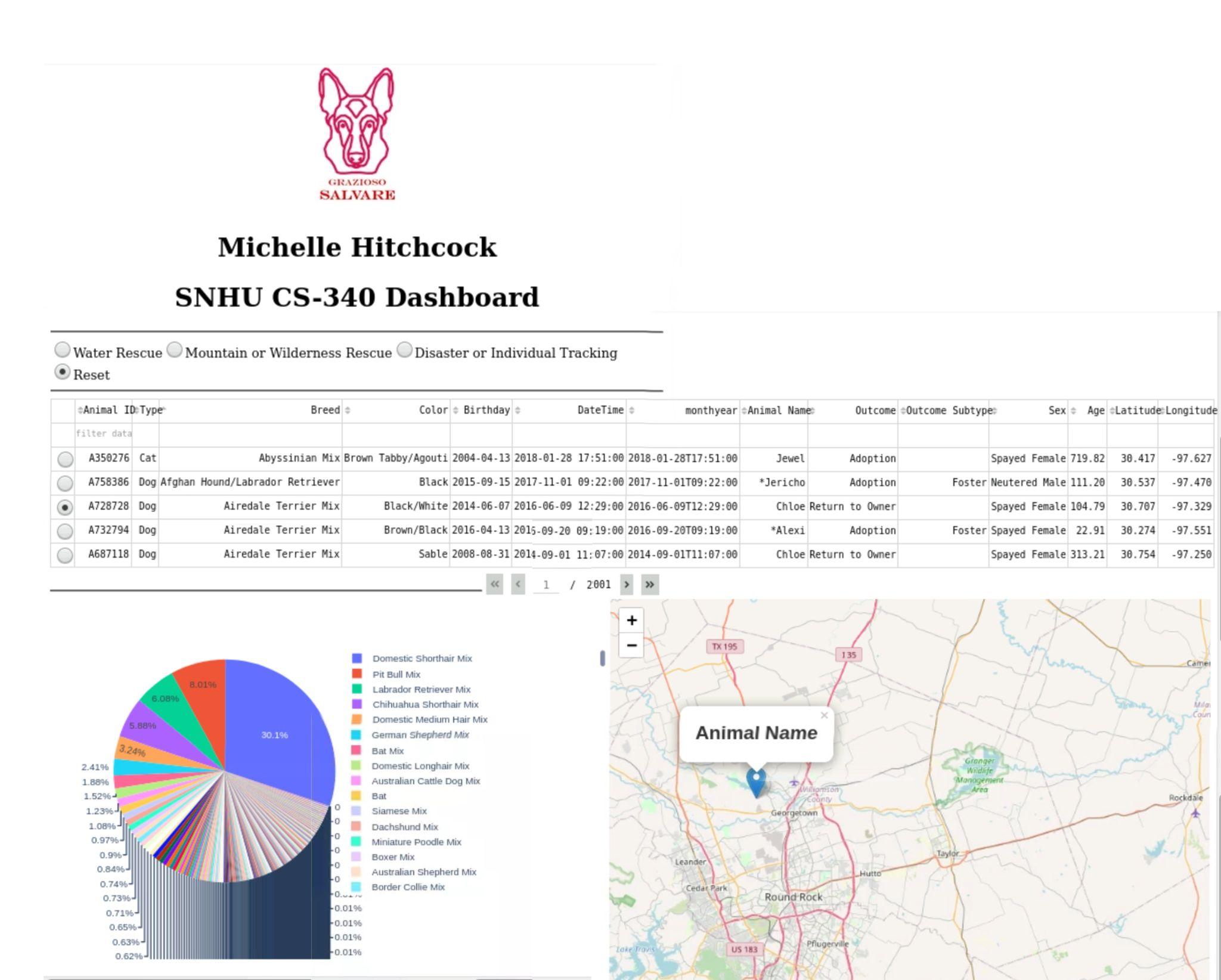
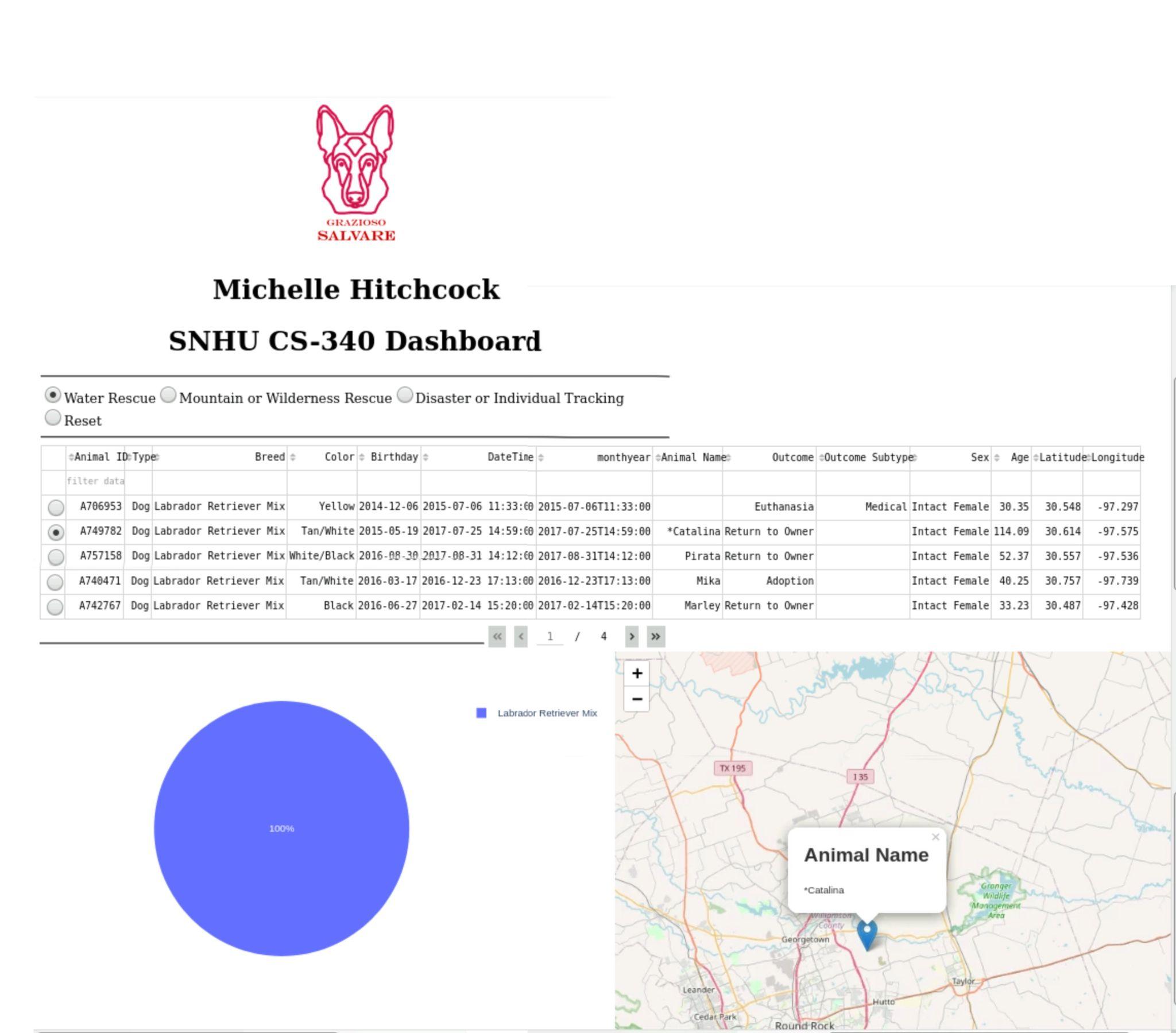
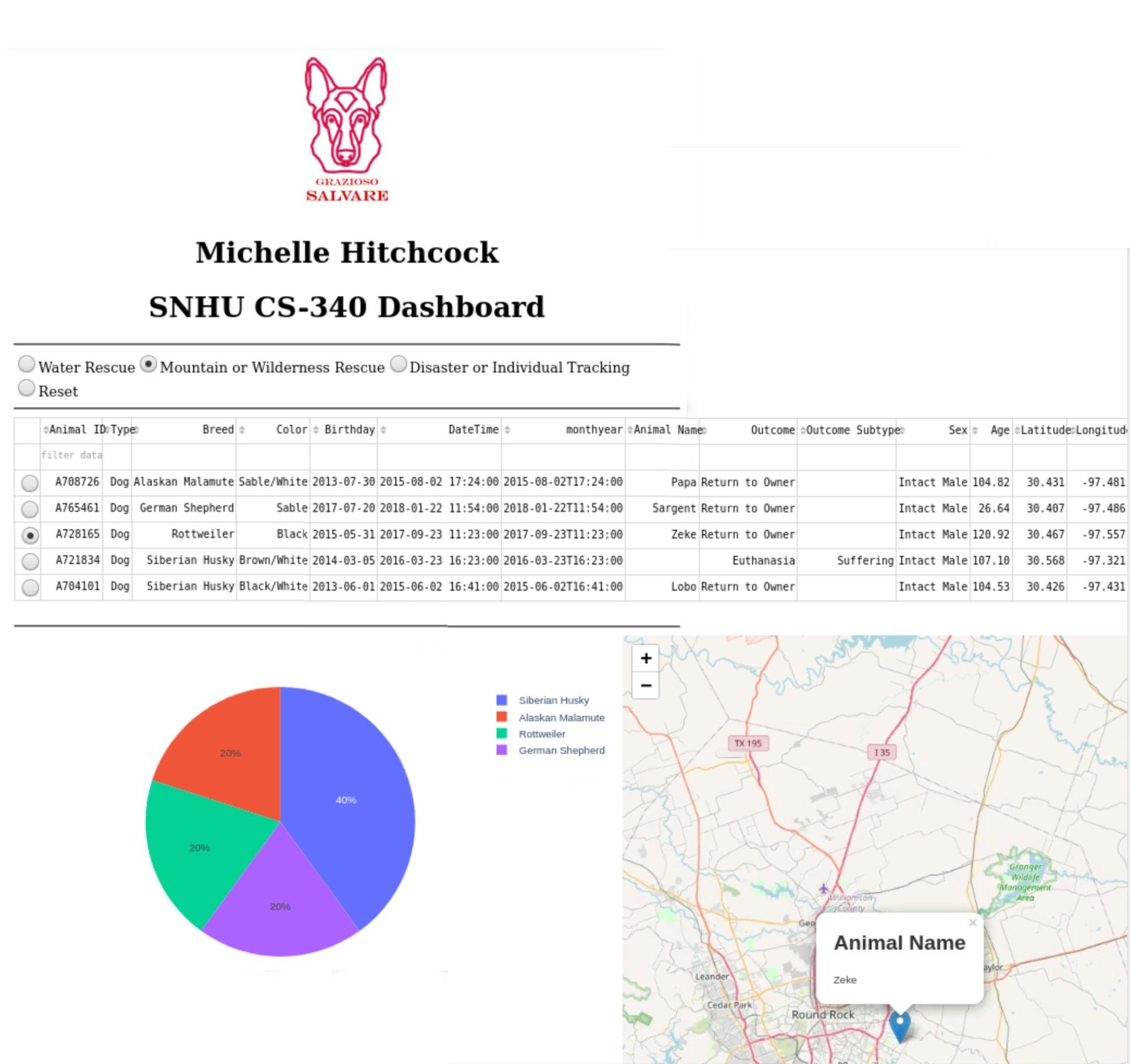
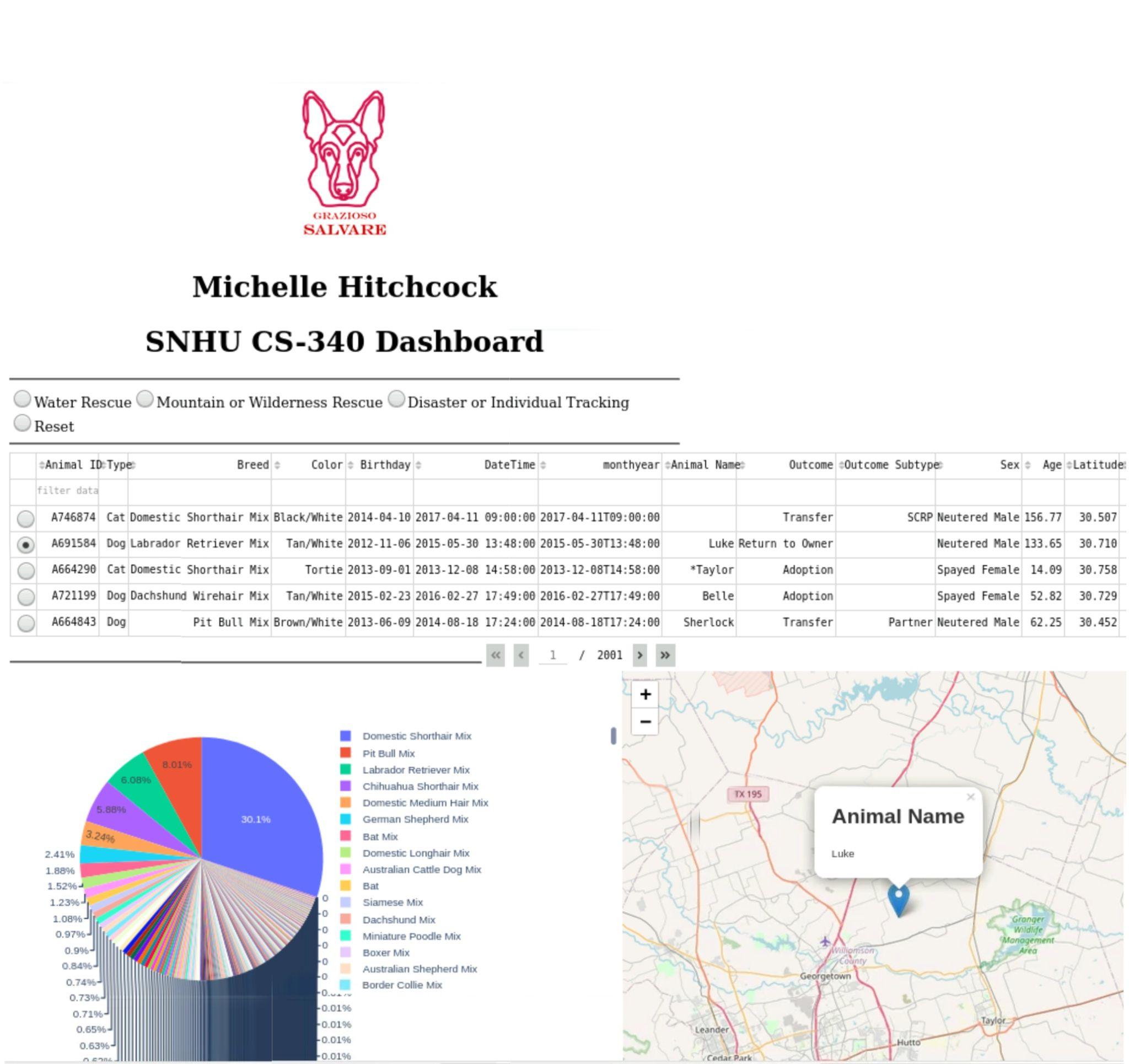
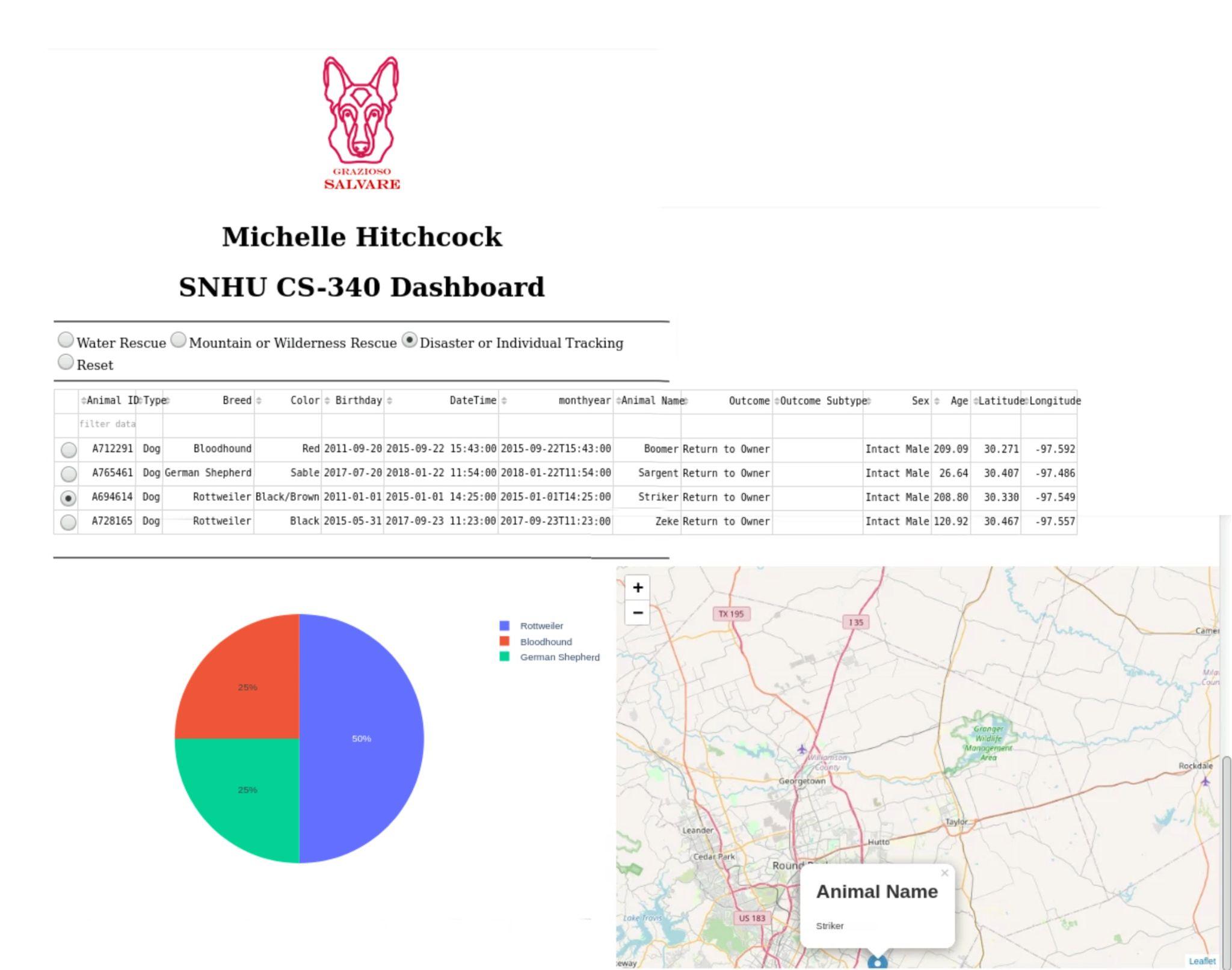
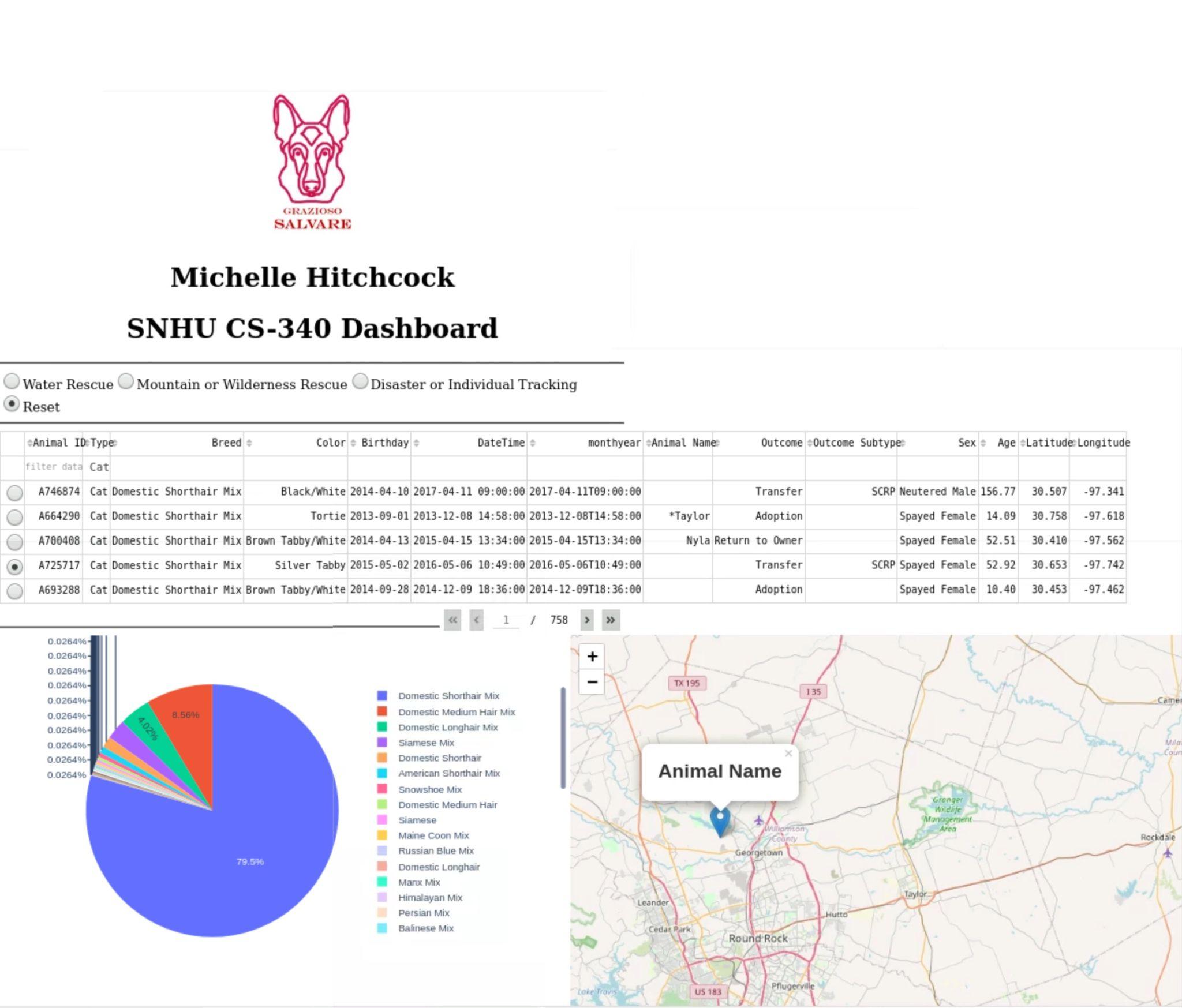
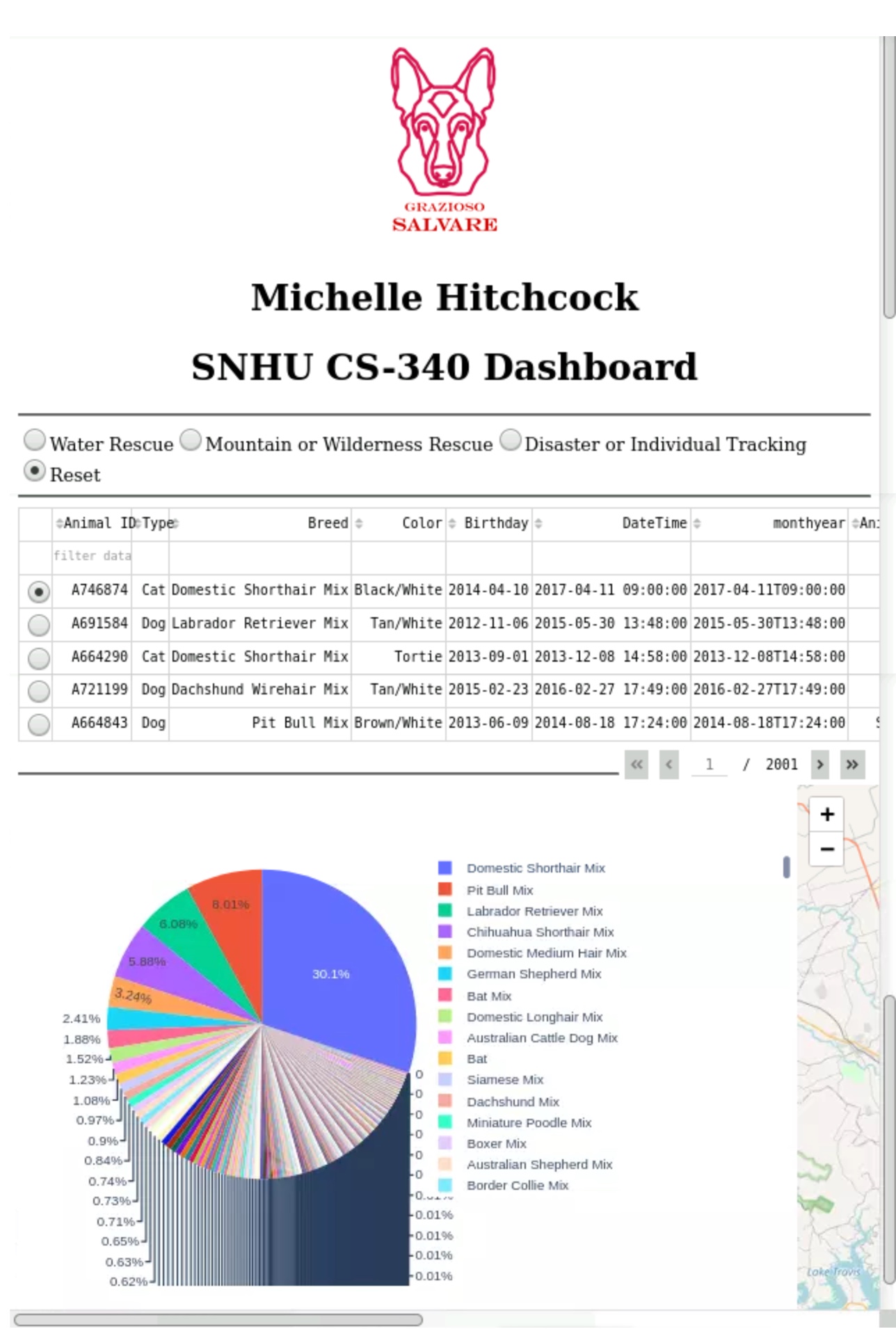
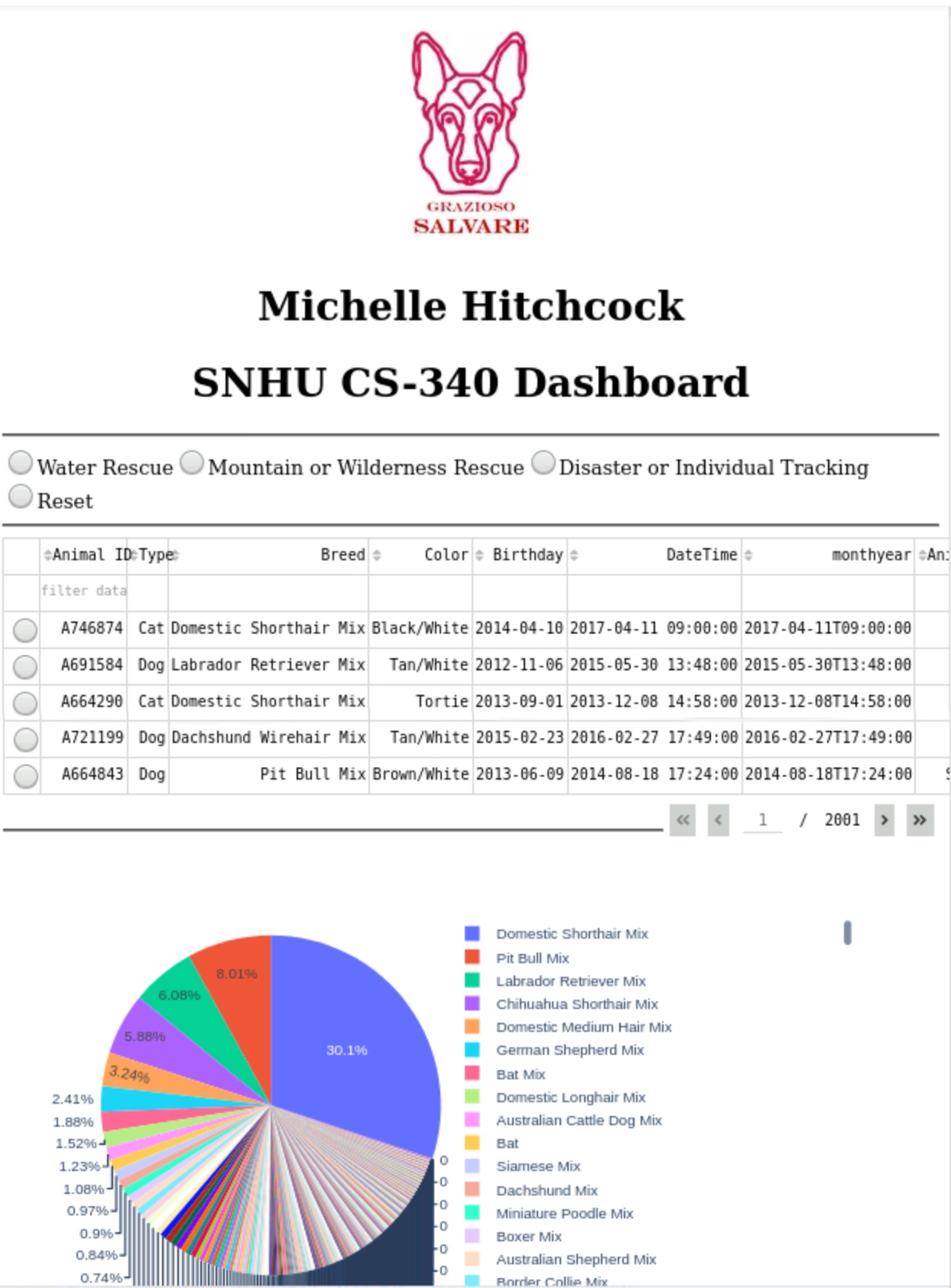
Testing using category search filter:

expected outcome:

updated Pi chart

### Screenshots

Provide screenshots that demonstrate your work.



These screenshots demonstrate testing the features of the radio buttons, filtering, searching using the categories, and updating displays of the map and pi-chart.

## Roadmap/Features

-Searching by category

-Filtering by category

-graph displays updated information based on the currently displayed filters

- A map will appear when an animal is selected and will update based on currently selected animals.

Radio Buttons:

Water

Mountain and Wilderness

Disaster or Individual Tracking

Reset

When selected these buttons will filter using their respective functions

**Challenges:**

some of the challenges that were faced in this project were updating the data for the charts and displays based on the user input.

solutions that were implemented included:

ensuring that the input included the data-id and the derived\_viewport\_data.

this ensured that the data was being sent from the correct location and being updated based on the current view.

## Contact

Michelle Hitchcock