# CS 340 Project Two README

## About the Project

This full-stack application allows users to identify and categorize available dogs from animal shelters around Austin, Texas.

## Motivation

The international rescue-animal training company, Grazioso Salvare, wanted to identify dogs that are good candidates for search-and-rescue training. They also wanted the project to be open-source and accessible on GitHub so that other organizations could use and adapt it.

## Getting Started

To use this application, download the ZIP file and extract the files onto your server. Make sure the files are placed in the same folder.

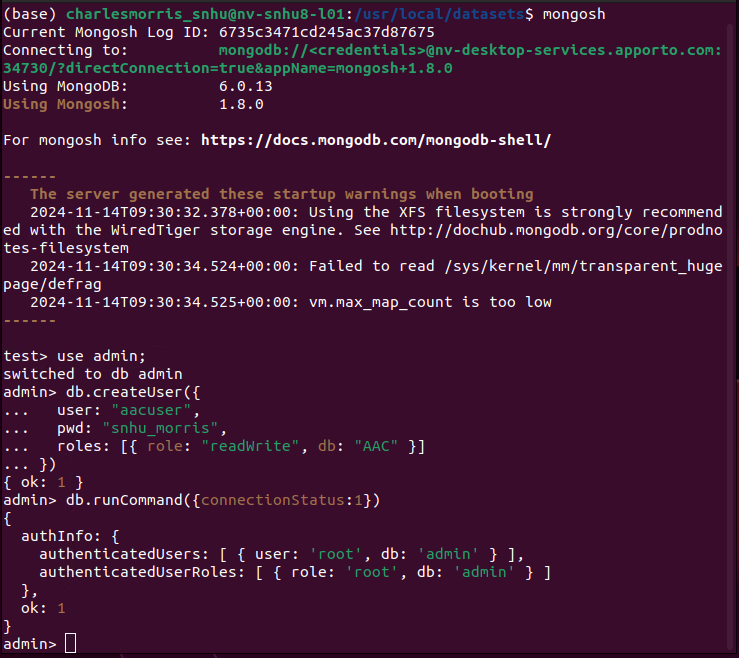
Before using this application, you should examine the connection variables in the **\_\_init\_\_** function of the AnimalShelter class in the **Mongo\_CRUD.py** file and modify them for your instance of MongoDB. **USER** and **PASS** should be the username and password you set up in MongoDB. **PORT** should be the port your MongoDB instance will use to connect. **DB** should be the name of the database containing the animal shelter collection. **COL** should be the name of the animal shelter collection.

Initially, I had trouble getting my module to connect to the database because my **DB** variable was “aac”, but I had named my database “AAC”. Improper capitalization caused the problem. I also had trouble connecting because I forgot to set the port. I found the proper port number by running the **mongosh** command at the terminal and examining the connection string.

Before using this module, you must first import the file **aac\_shelter\_outcomes.csv** into your MongoDB database:



## You must also create a separate user account aacuser with a unique password that you will use later to log in:





After this is complete, open the file **ProjectTwoDashboard.ipynb** and set the username and password variables to the username and password you set up in your database:



Now, run the file **ProjectTwoDashboard.ipynb** in Jupyter Notebooks. Users navigating to your website will be shown the application dashboard.

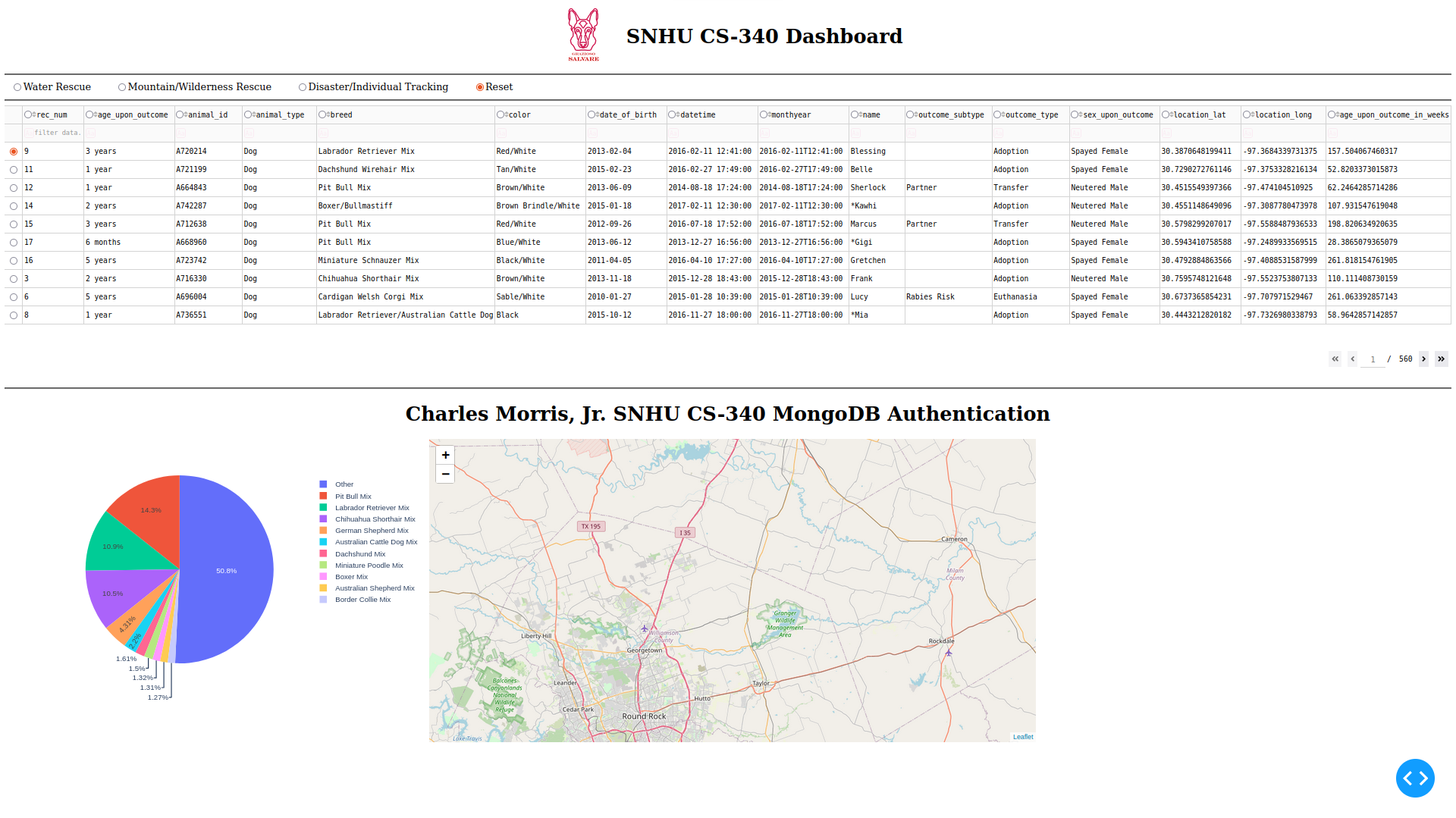
## Installation

To use this module, you will need the following software installed:

* **Python:** Mongo CRUD is written in Python and needs the Python compiler to run
* **MongoDB:** For the MongoDB database.MongoDB was chosen for this project because its NoSQL structure makes storing data about dogs and animal shelters easy without requiring a predefined schema.
* **PyMongo:** This library is necessary to allow Python to work with MongoDB. We can easily make queries to retrieve new data from the database based on the selected filters.
* **Plotly Dash:** This framework provides all the essentials to create a dashboard easily. This was used to make the data table, map display, and pie chart.
* **Jupyter Notebooks:** This is required to run the file **ProjectTwoDashboard.ipynb**

## Usage

When you first open the dashboard, you will see this:



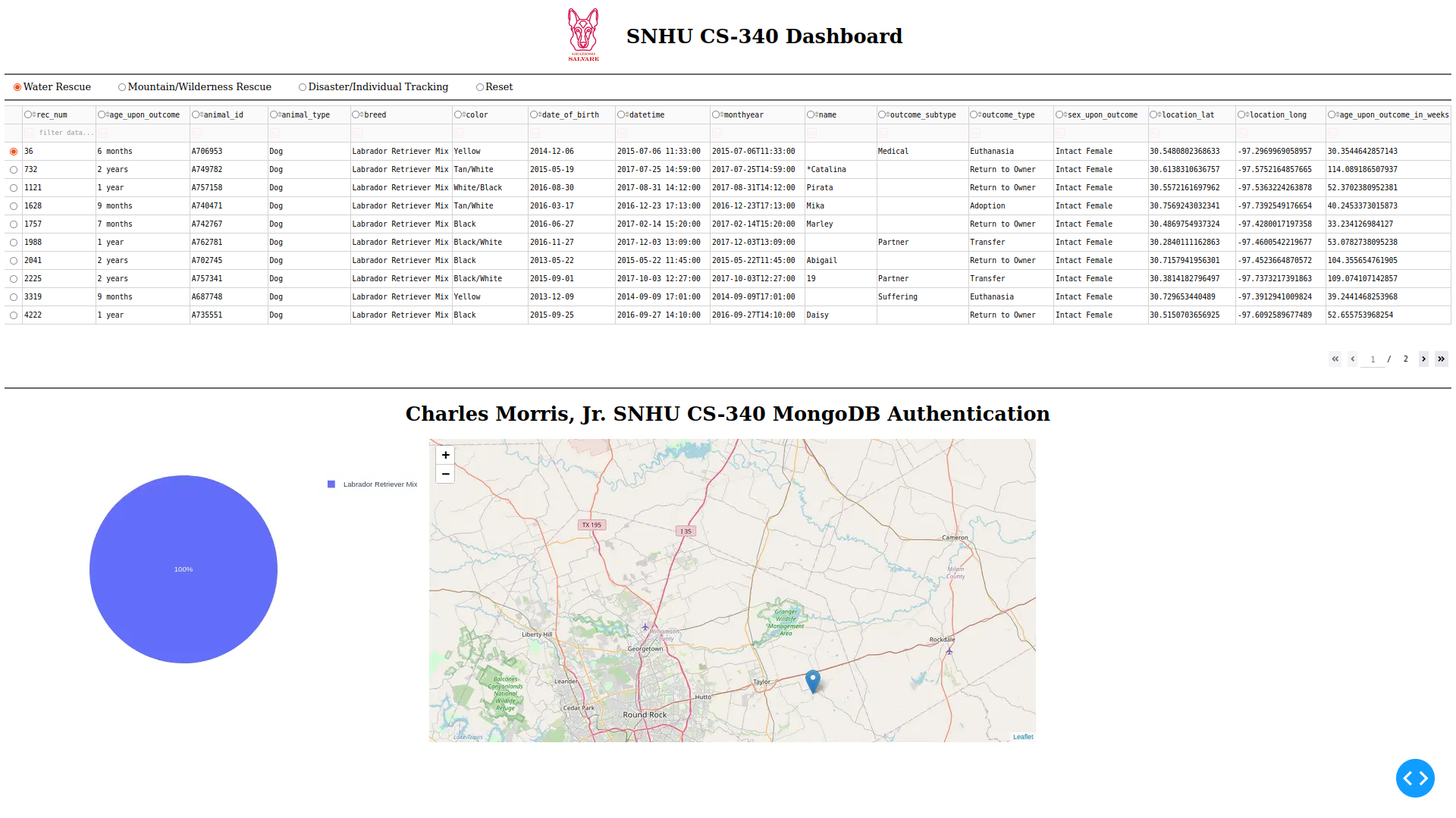
**The dashboard has the following features:**

* Radio buttons to filter dogs based on the type of rescue or tracking they will be performing
* A data table with detailed information on the dogs
* A pie chart graph that shows the different dog breeds
* A map that shows the location of the currently selected dog

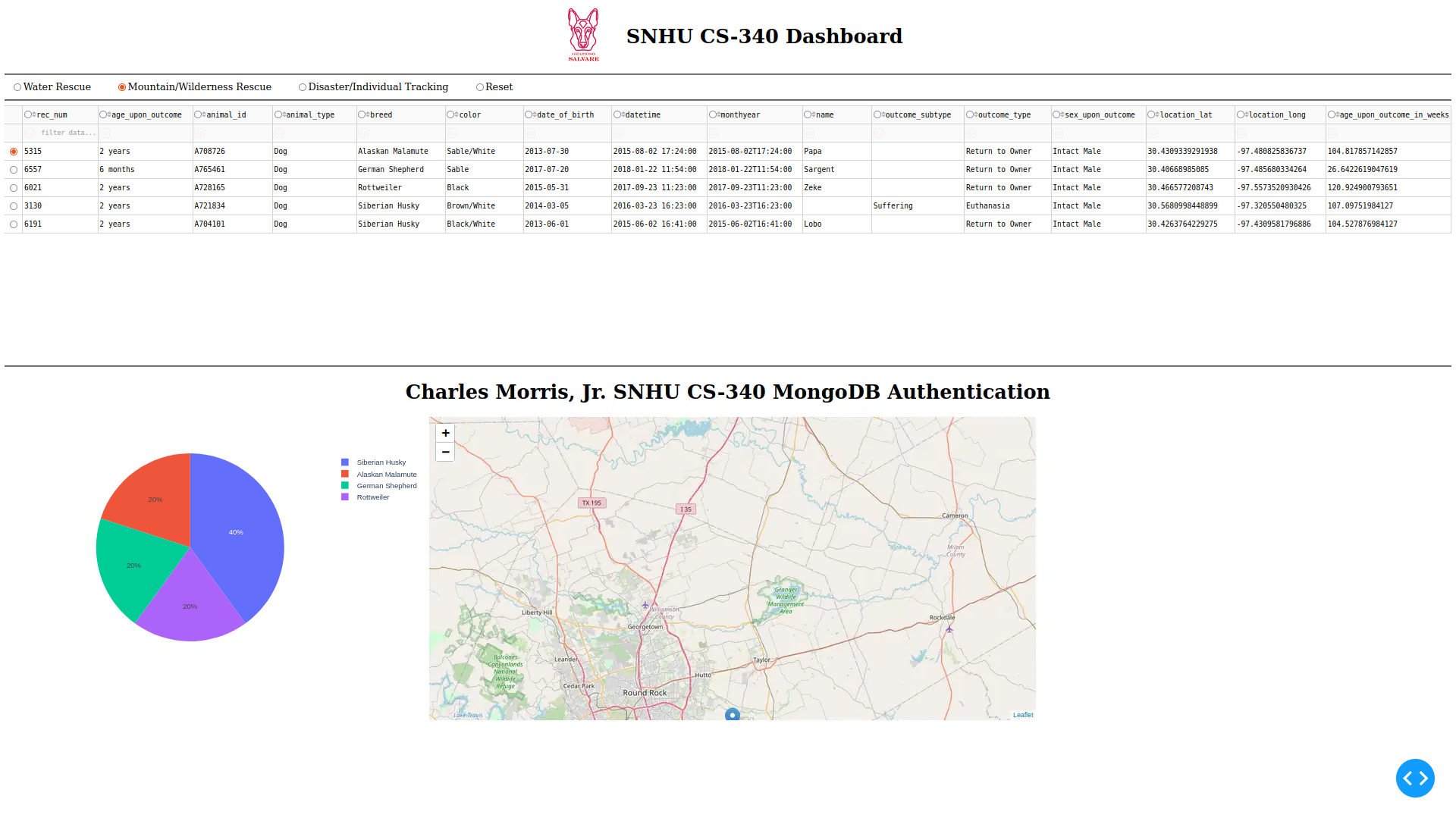
The row of radio buttons allows you to filter the data by dogs capable of water rescue, mountain/wilderness rescue, or disaster/individual tracking. You can also view all the data by selecting the “Reset” option.

**Examples:**

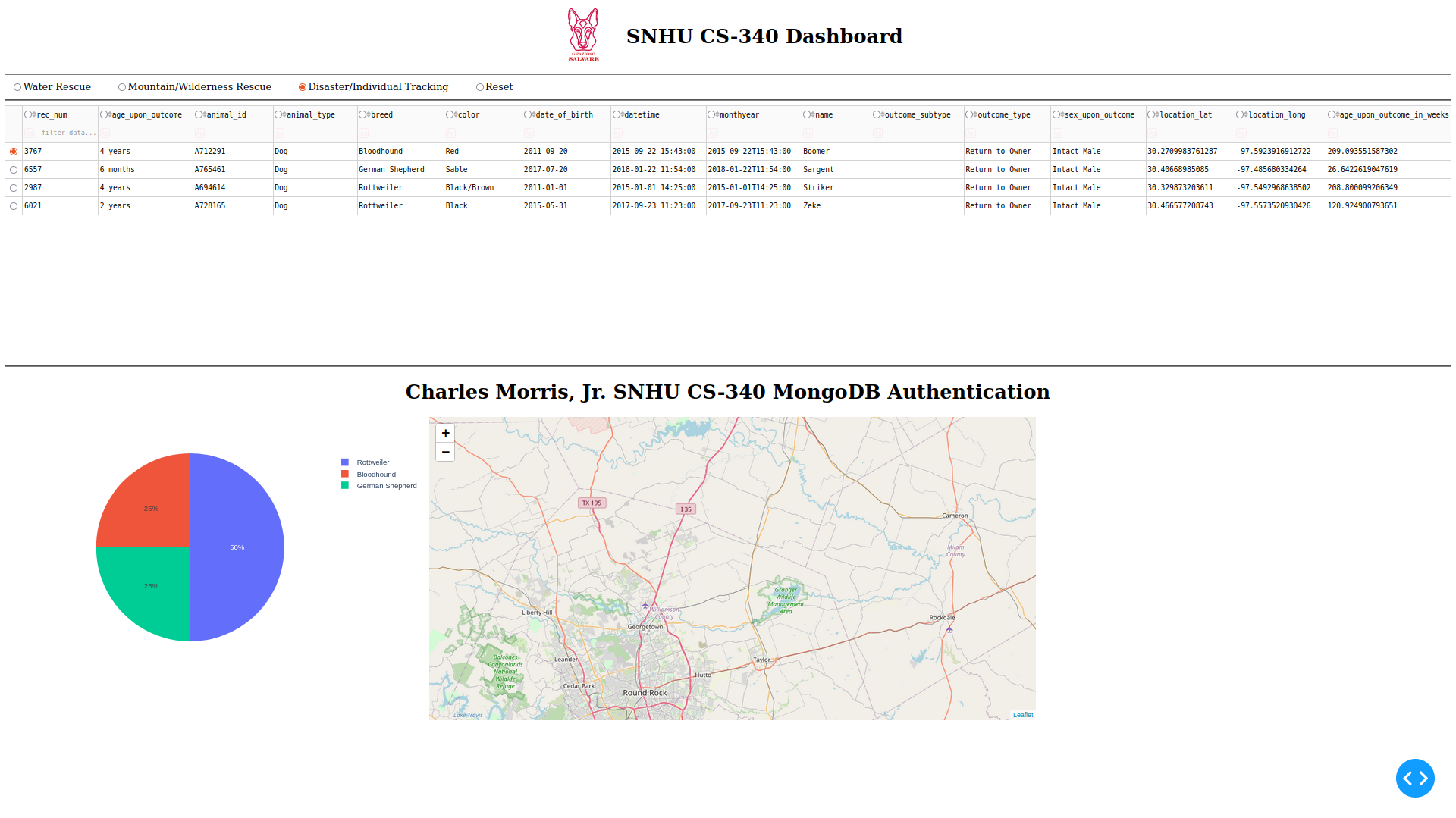
**Water Rescue**



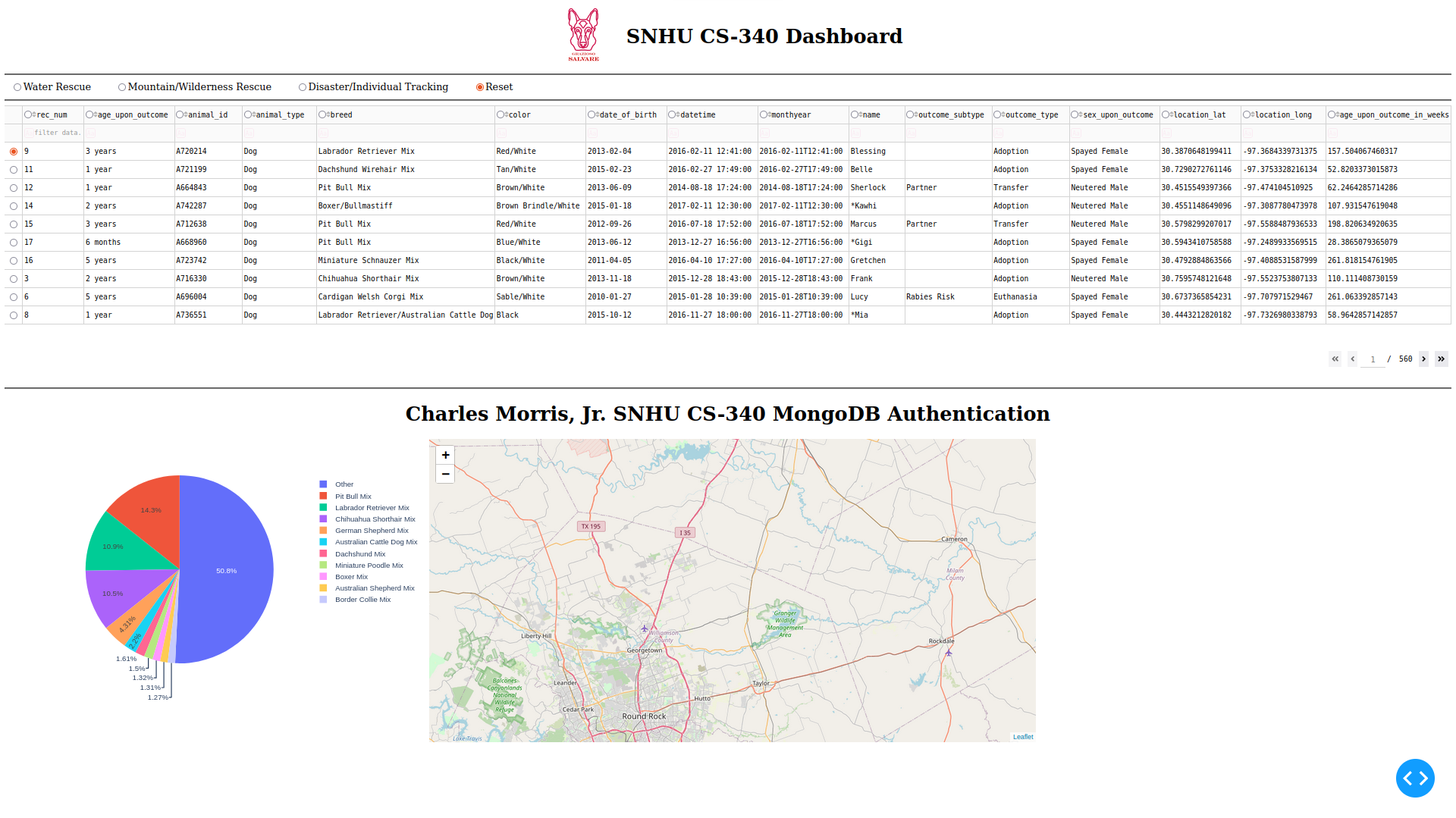
**Mountain/Wilderness Rescue**



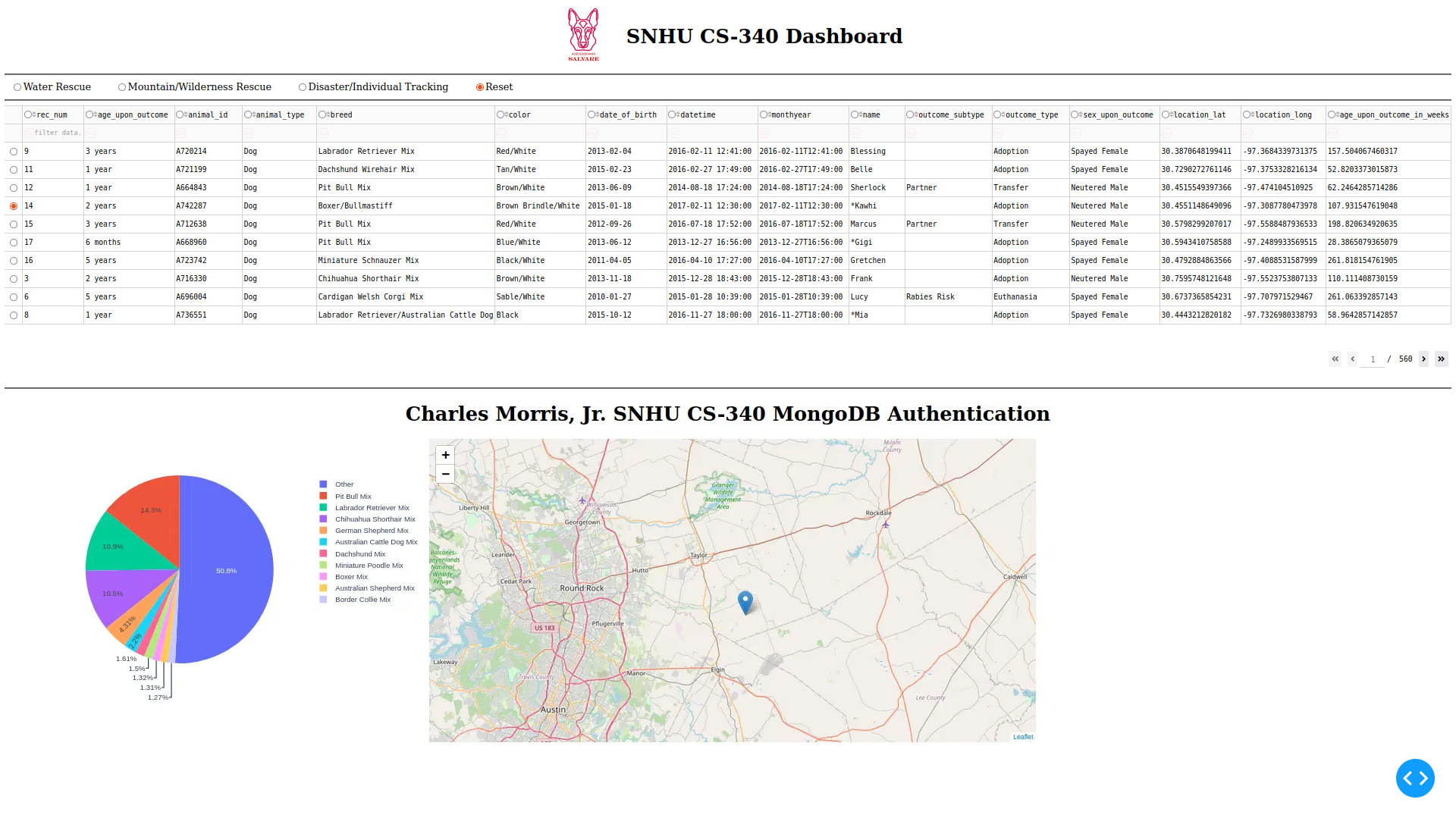
**Disaster/Individual Tracking**



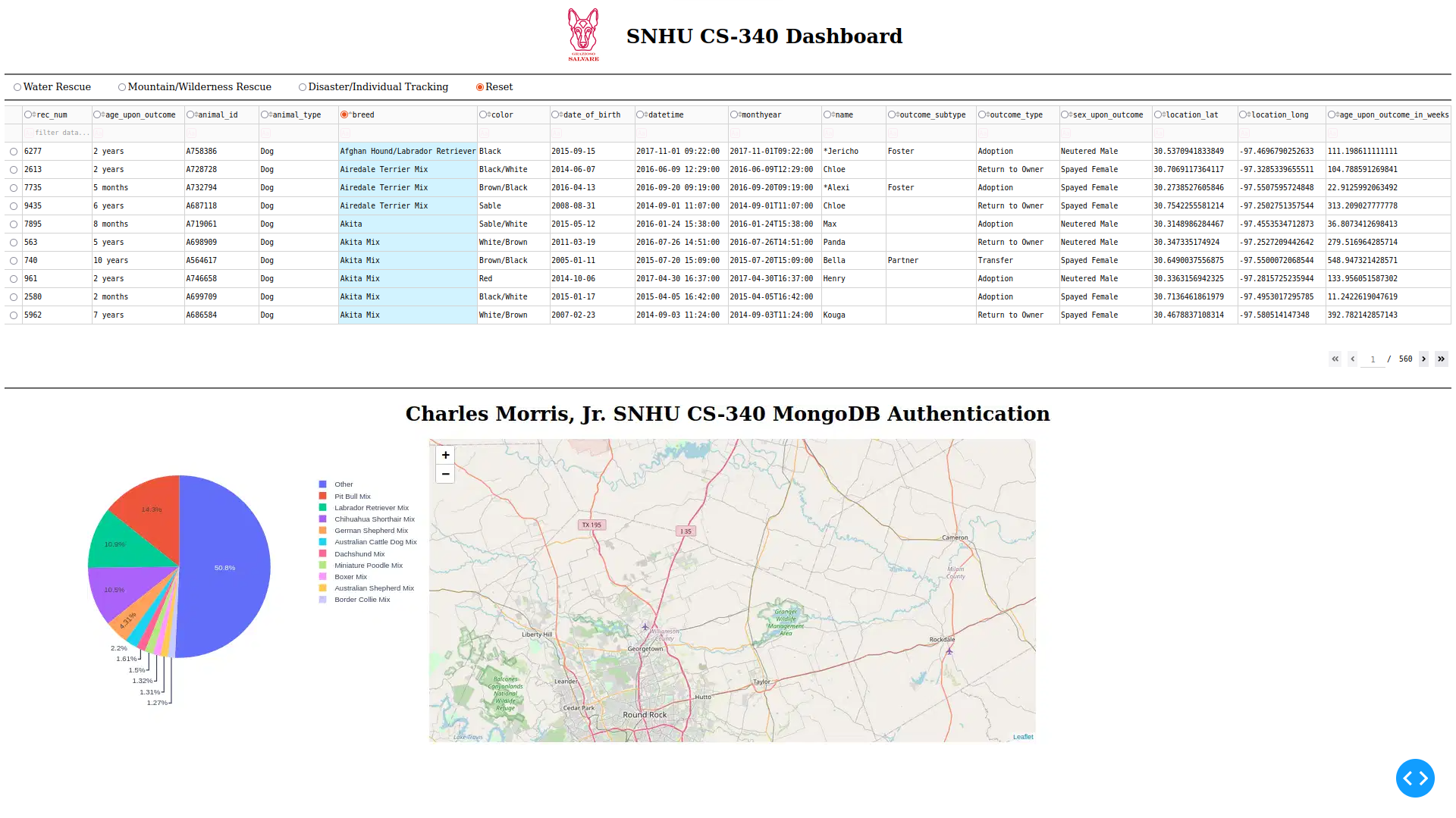
**Reset (removes all filters)**



Below the radio buttons is the data table. This displays all the dogs currently stored in your database. The leftmost column allows you to select a dog and see its location on the map:



To the left of the name of each column are arrows that allow you to sort the data in ascending or descending order by that column:



The pie chart visually represents all the data currently shown in the data table. It changes based on the filters you select.

One of the challenges I faced when drawing the pie chart was that it would be improperly drawn when attempting to visualize the entire dataset because there were too many breeds. To fix this, I only showed the top 10 dog breeds in the current data. All remaining breeds are placed into a category called “Other.”

## Contact

Your name: Charles Morris, Jr.