**CS340 README Document**

**About the Project**

This Grazioso Salvare Dashboard is an interactive web application that is designed to filter and display data from a database documenting animal information from Austin Animal Center in Texas. The dashboard includes functionality allowing the user to filter the data with specific requirements, interact with the data table on screen, interact with a geolocation chart, and view a pie chart containing data from their filter selections.

**Tools Used**

**Python Coding Language:** Used to script and perform data manipulation

**Dash Framework:** Used for building web applications

**MongoDB:** Database used for the storage and retrieval of data

**Plotly:** Used for creation and usage of interactive charts and graphs

**MongoDB Explanation**

MongoDB was selected for this project because of its ability to handle large amounts of unstructured data, and its ability to easily interface with Python via the pymongo library.

**Dash Framework Explanation**

The Dash Framework is a great resource for building effective web applications in Python. It has interactive capabilities and has real time data visualization while integrating seamlessly with Plotly.

**Steps Taken**

1) **Setup MongoDB and Python Environments**

2) **Develop CRUD code:** In this case, see animal\_shelter.py

3) **Build Dashboard Layout and Style:** Create interactive filters, data table and charts, as seen in ProjectTwoDashboard.ipynb

4) **Implement Callbacks:** Callbacks add interactivity to update the charts and graphs in real time based on filter selections

5) **Testing and Debugging:** Validation of all parts, ensuring they all operate as expected.

**Challenges and Solutions**

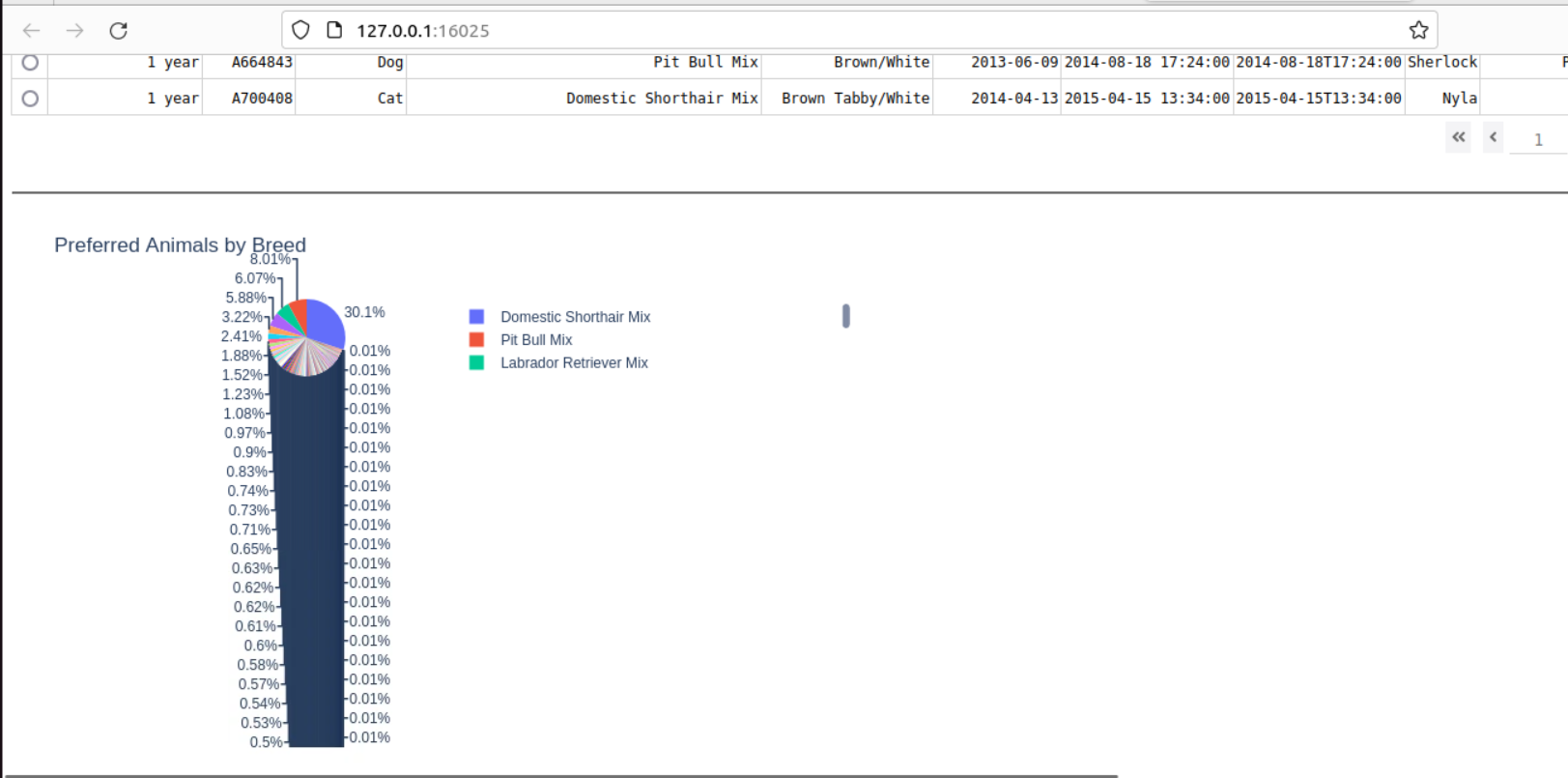
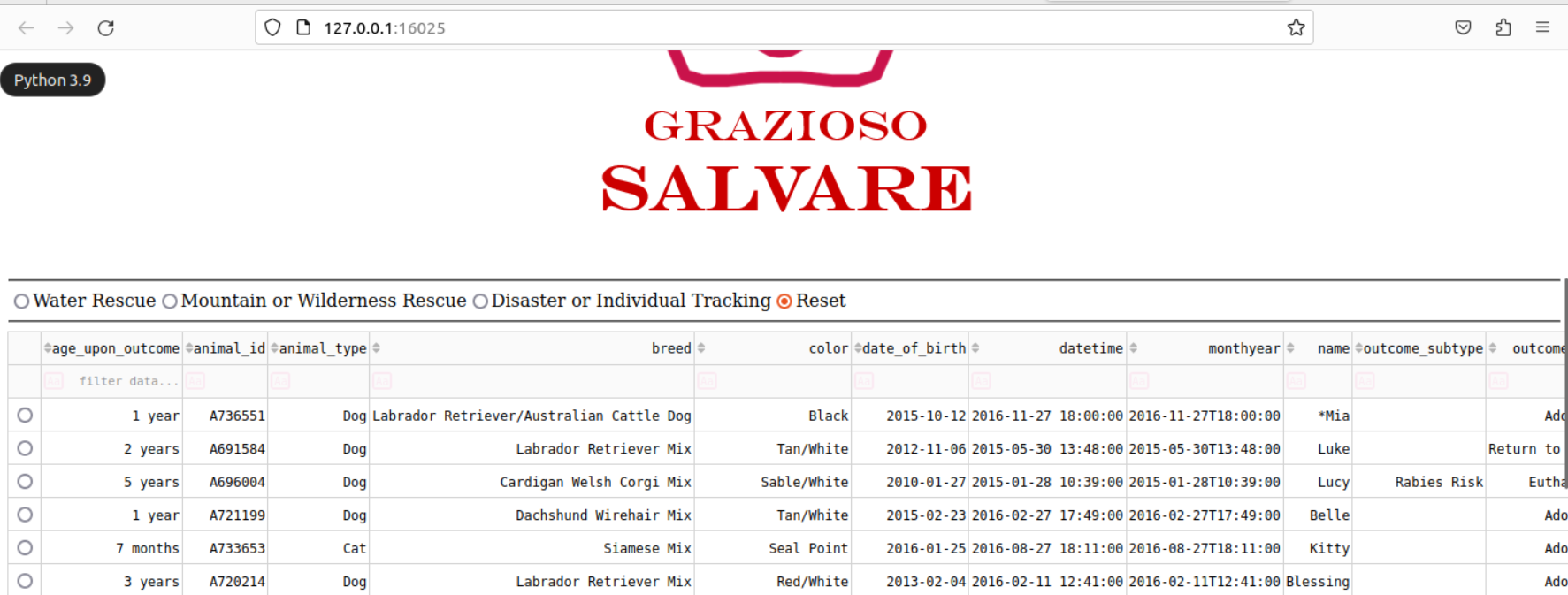
**Authentication Errors:** Made changes to the CRUD module to resolve error calls for authentication parameters.

**Interactive Filtering:** Fixed issues with filter options not applying to the data table by altering the code for the table and refining queries.

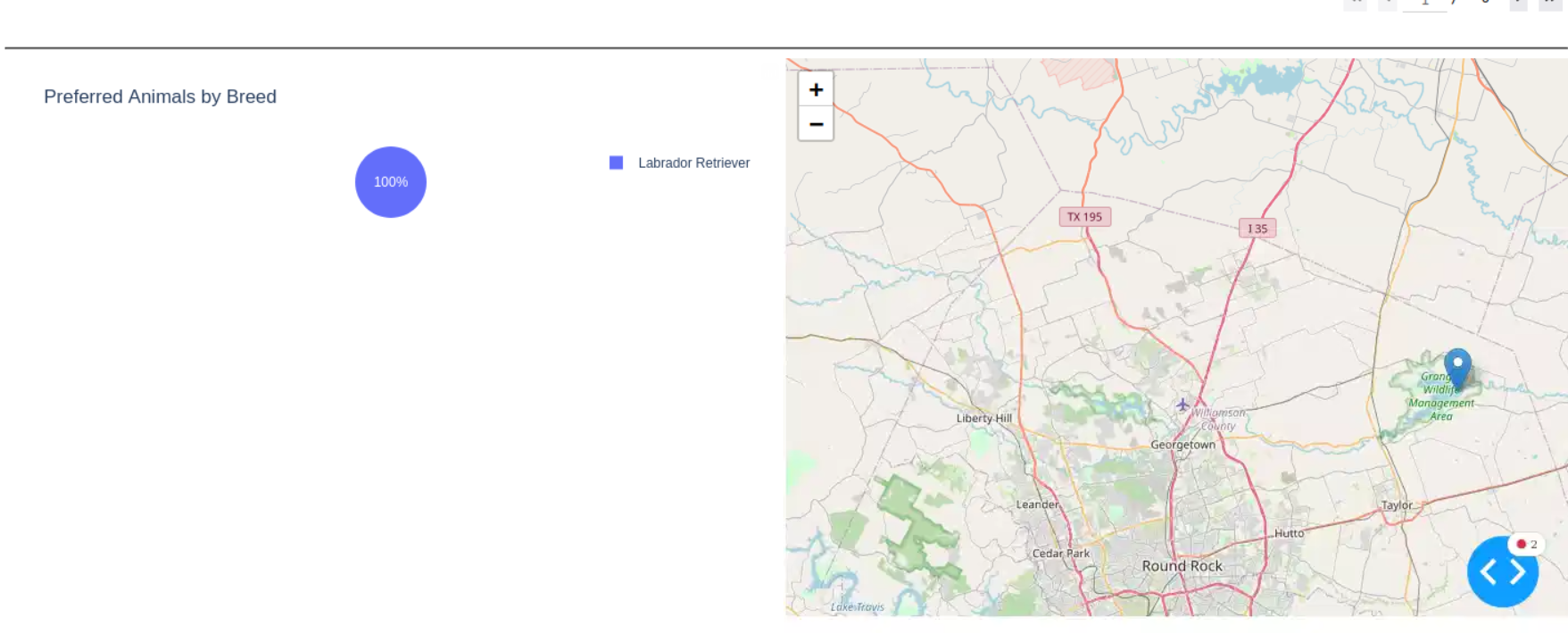
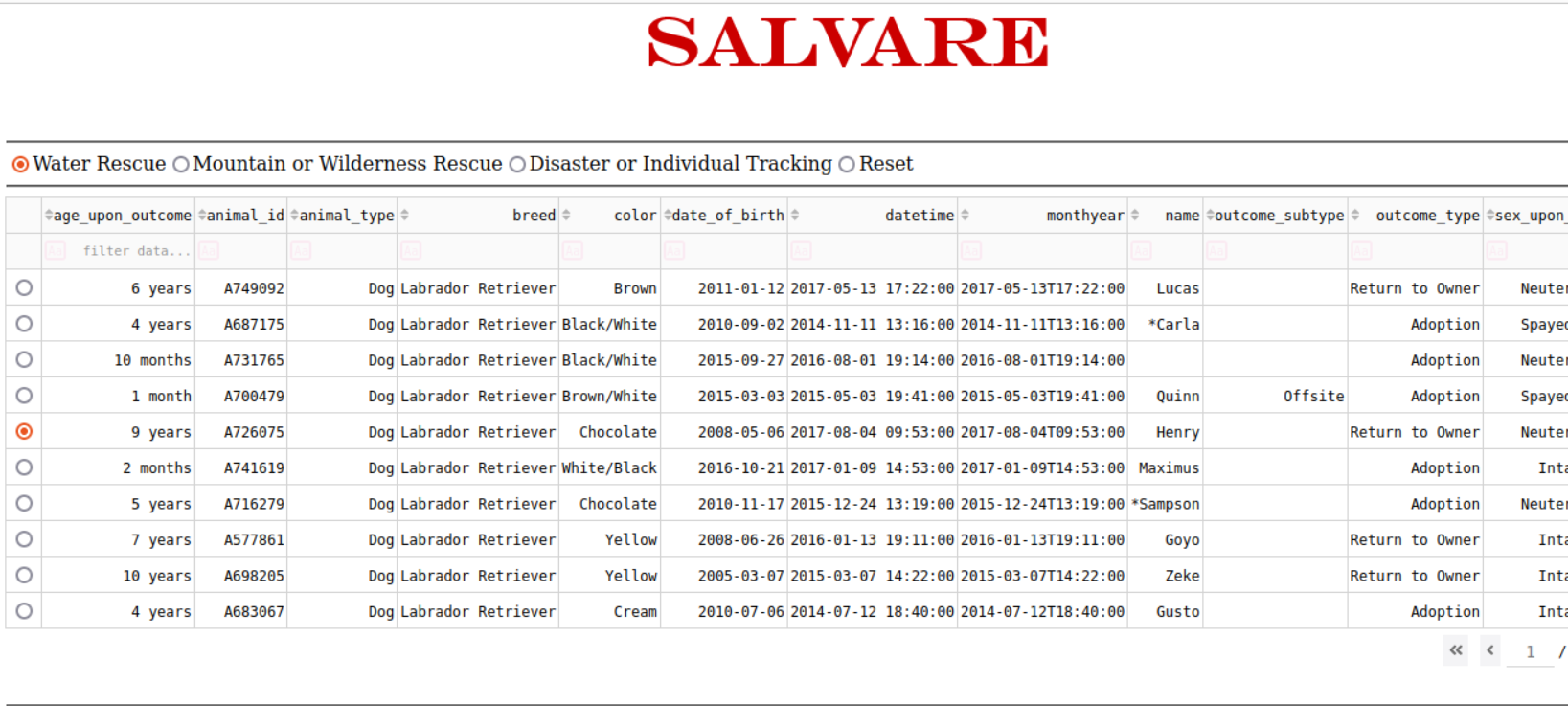
**Geolocation Chart Functionality:** Issues arose where the geolocation chart was not displaying correctly, and then where it was not interactive. Fixed by checking code structure and changing keywords

**Screenshots**

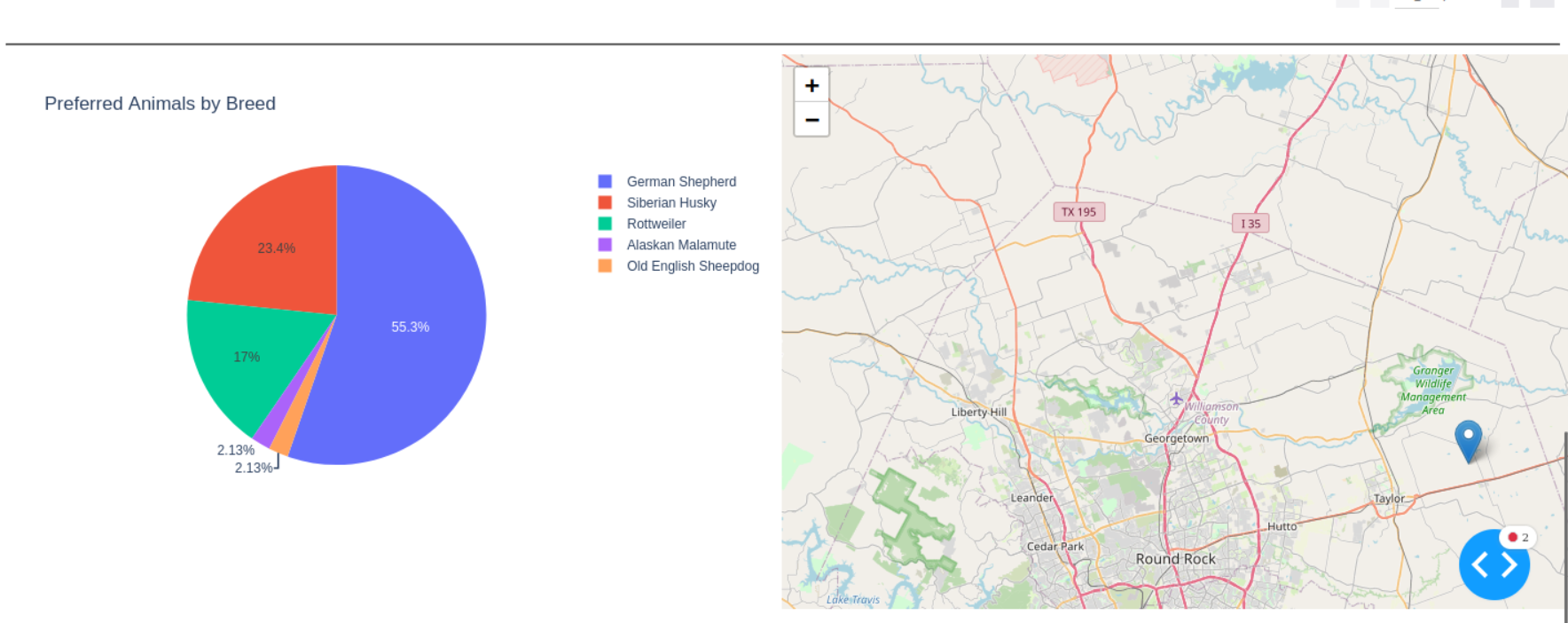
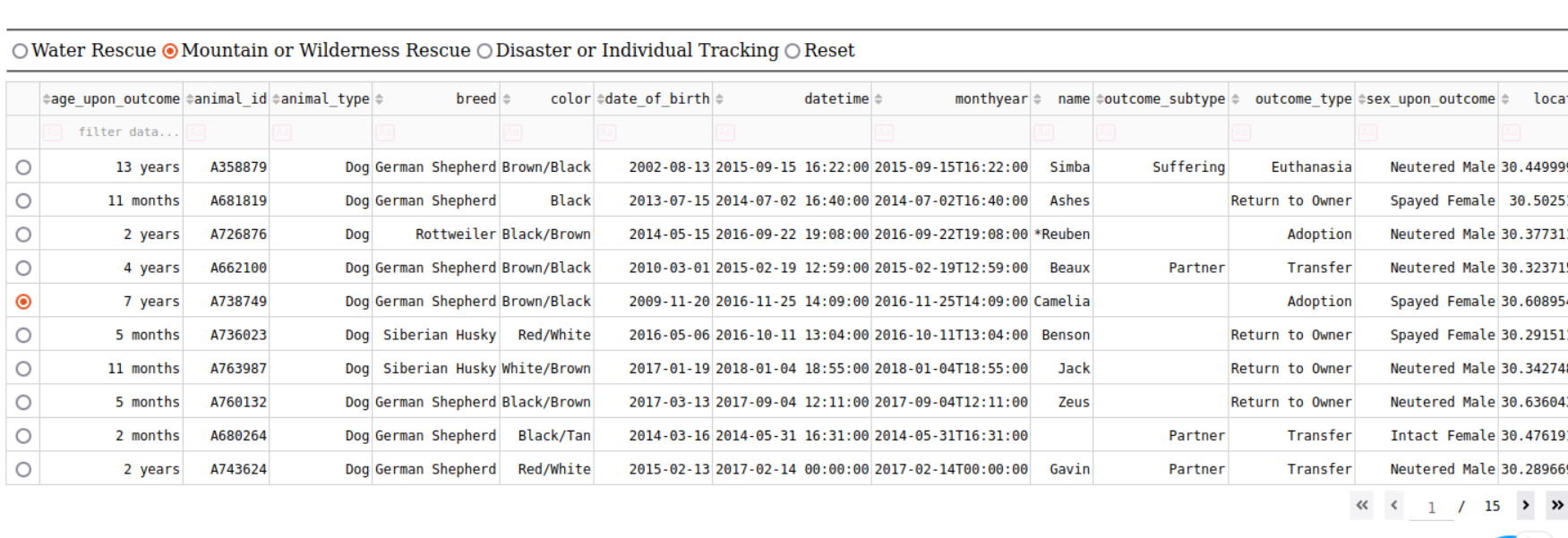
Starting State on Launch:

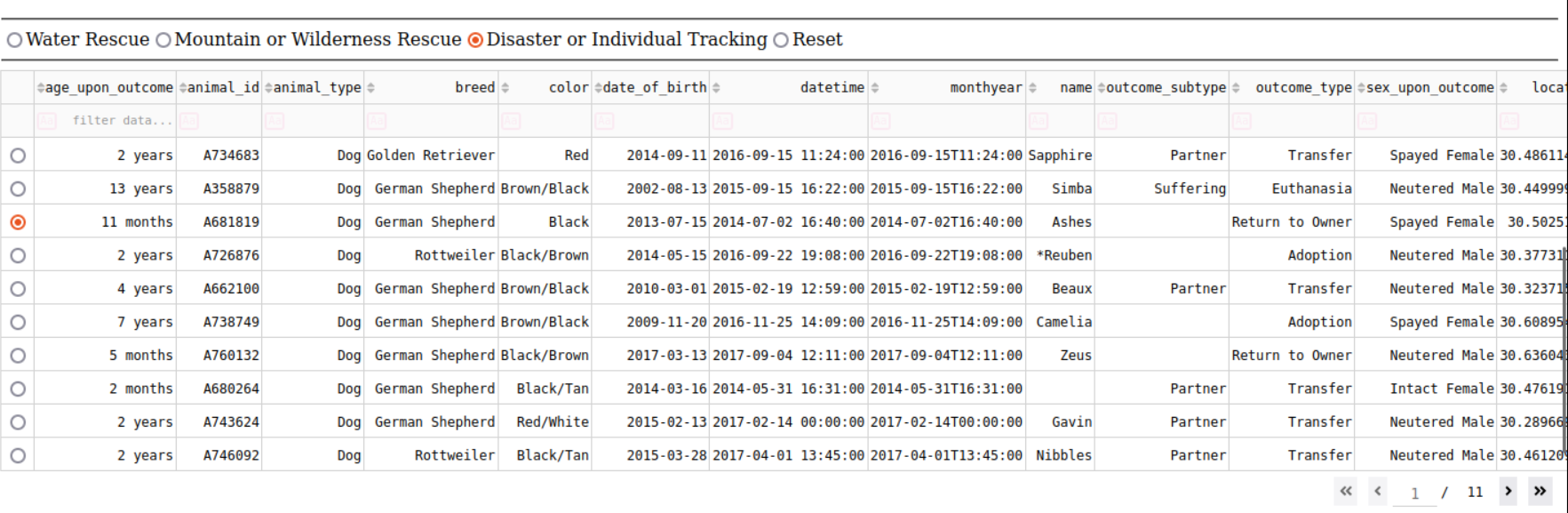


Water Rescue Selection:



Mountain or Wilderness Rescue Selection:

Disaster or Individual Tracking Selection:





Reset:

