**Project Two**

Ikechukwu Onuoha

Southern New Hampshire University

CS-340-J7030 Client/Server Development 23EW2, CS-340-J7030

Satish Penmatsa, Ph.D.

December 11, 2023

# **Project Overview**

This dashboard project aims to facilitate animal shelter management for Grazioso Salvare by providing an intuitive web-based interface to interact with their animal database. The dashboard allows users to filter, visualize, and display pertinent information about animals in the shelter based on specific criteria such as rescue type, breed, age, and sex upon outcome.

# **Required Functionality**

The dashboard enables users to filter animals based on three rescue types: Water Rescue, Mountain or Wilderness Rescue, and Disaster Rescue or Individual Tracking. It provides a data table showcasing animal records, a pie chart illustrating breed distribution, and an interactive map displaying geographical locations of selected animals.

# **Tools Used**

* **JupyterDash:** Employed as the framework for developing the interactive web application directly in Jupyter Notebook, providing ease of integration with Python code.
* **Dash:** Utilized for creating the dashboard's view and controller components, offering a reactive and interactive user interface.
* **Dash Leaflet:** Integrated for generating interactive maps within the dashboard.
* **Plotly Express:** Used for data visualization, particularly for creating the breed distribution pie chart.
* **Pandas, NumPy, Matplotlib:** Employed for data manipulation, handling, and visualization.
* **MongoDB:** Chosen as the model component for its flexibility, scalability, and compatibility with Python. MongoDB facilitates seamless interfacing with Python through the animals\_shelter module, enabling data retrieval and manipulation for the dashboard.

# **MongoDB Usage**

MongoDB was selected due to its schema-less nature, which accommodates flexible data structures common in animal shelter databases. Its ability to store JSON-like documents aligns well with the varied animal records. The animals\_shelter module aids in database interaction, enabling querying and retrieval of relevant data for display in the dashboard.

# **Dash Framework**

Dash, built on top of Flask, React, and Plotly.js, provides a reactive and component-based framework for creating web applications using Python. It allows for easy integration of interactive elements like graphs, tables, and maps, offering a seamless user experience.

# **Steps Taken for Completion**

* **Data Retrieval:** Connected to the MongoDB database using the AnimalShelter module, retrieved animal data, and structured it into a Pandas DataFrame.
* **Dashboard Layout:** Designed the dashboard's layout using HTML and Dash components, integrating a logo, filter options, data table, charts, and maps.
* **Callbacks Implementation:** Defined callback functions to update the dashboard components based on user interactions, such as filtering, selecting rows/columns, and updating visualizations.
* **Testing and Deployment:** Conducted testing to ensure functionalities work as expected and deployed the dashboard within the Jupyter Notebook environment using app.run\_server().

# **Challenges**

* **Data Querying Complexity:** Dealing with complex queries for various rescue types required careful structuring of MongoDB queries. This was resolved by constructing conditional queries based on filter types.
* **Mapping Geolocation Data:** Handling geolocation data and ensuring accurate mapping required verifying column existence and handling potential empty or missing values, resolved by thorough data validation checks within callback functions.

# **Instructions**

* Ensure Python environment setup with necessary libraries (Dash, Pandas, etc.).
* Connect to the MongoDB database using the credentials in the animals\_shelter module.
* Run the code within a Jupyter Notebook environment to start the dashboard.
* Access the dashboard via the localhost URL to interact with the functionalities.

# **Testing**

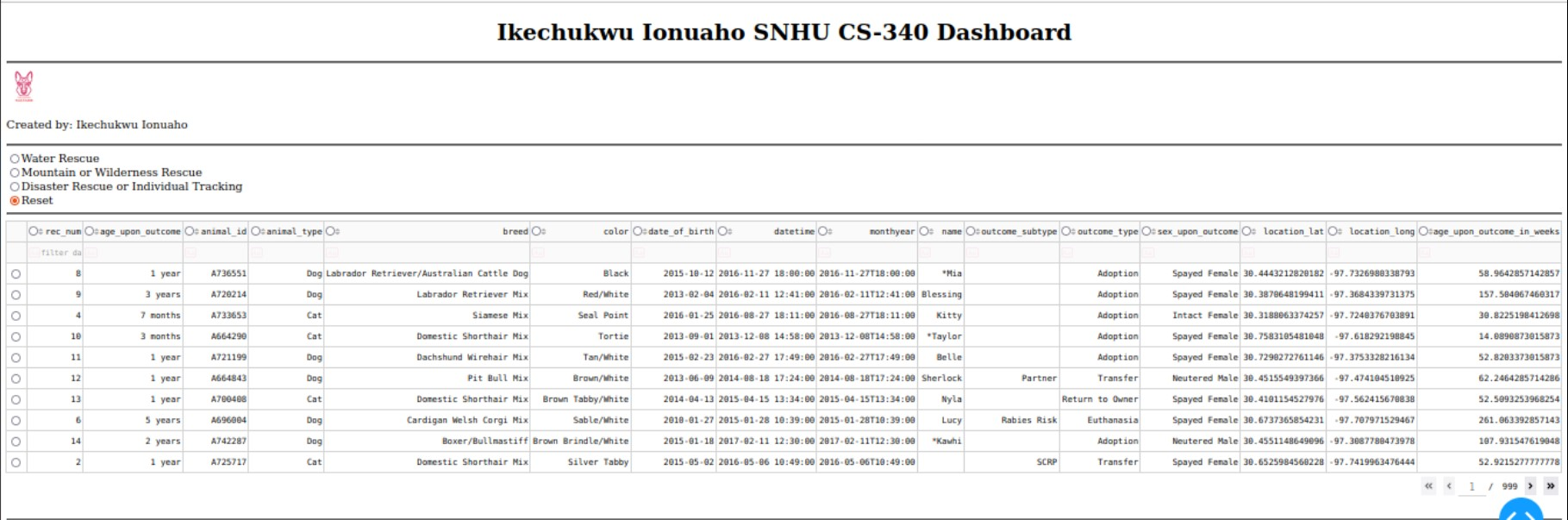


Figure : Dashboard

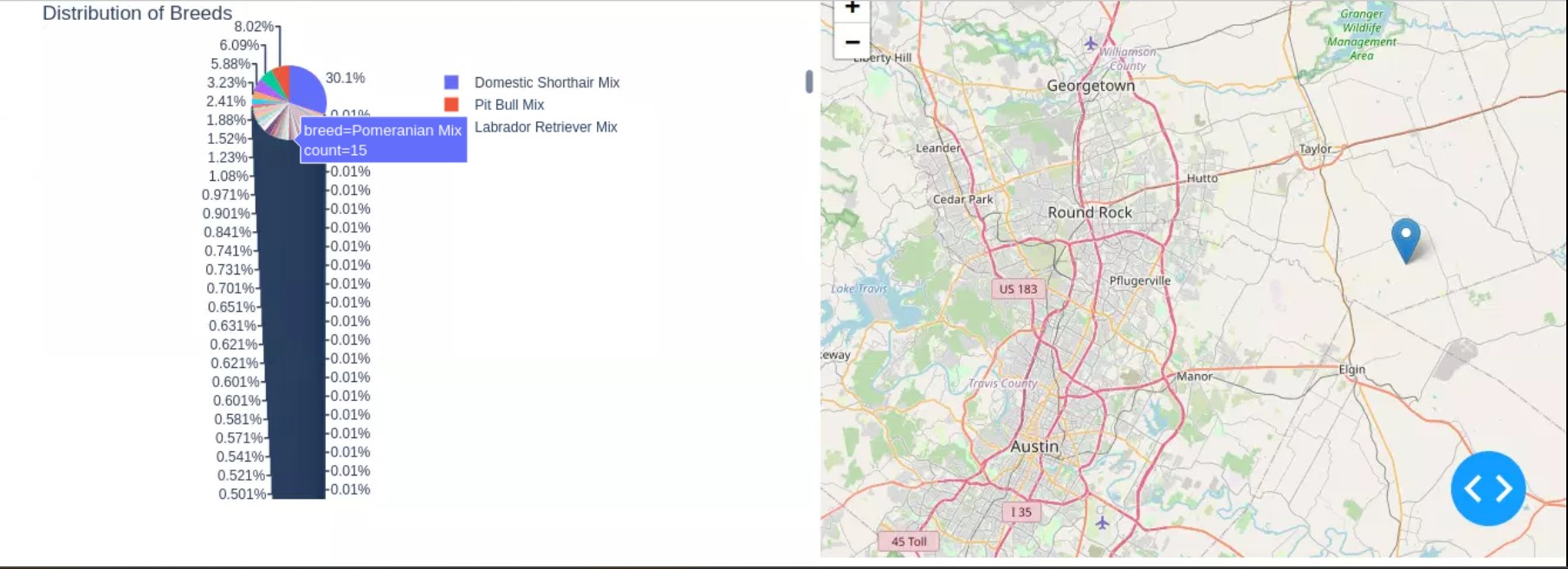


Figure : Reset Chart and Map

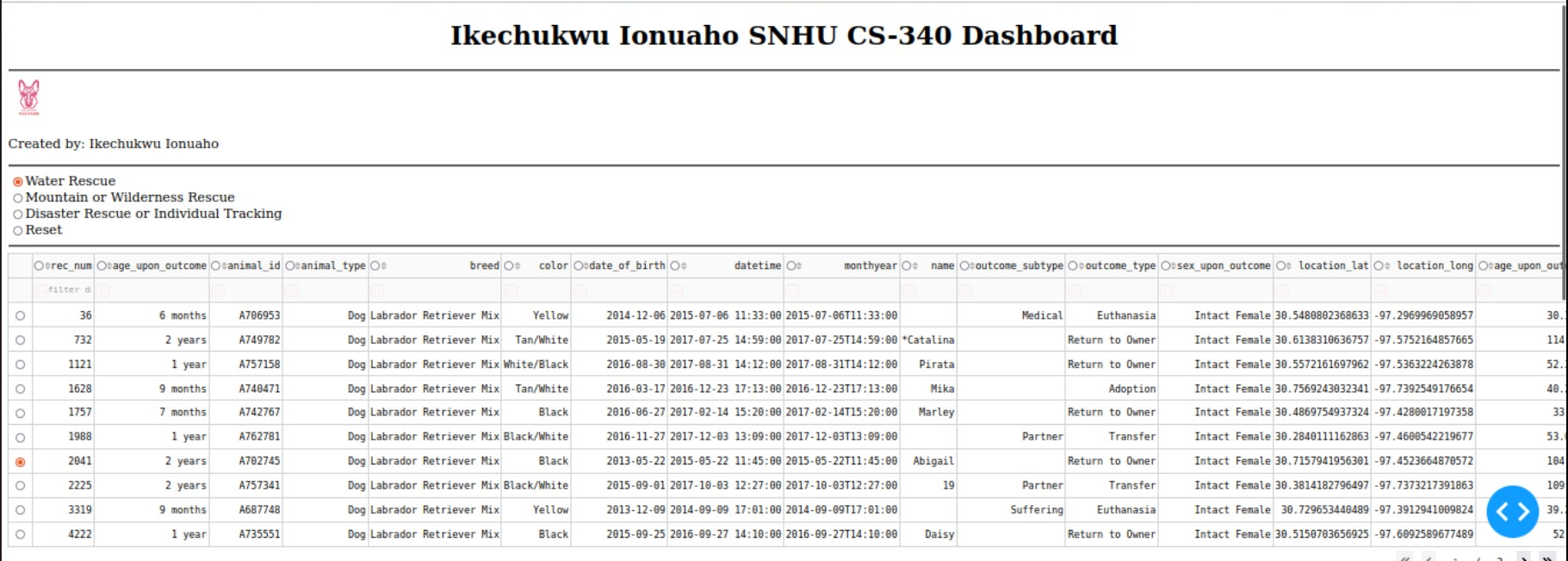


Figure : Water Rescue Dashboard



Figure : Water Rescue Chart and Map

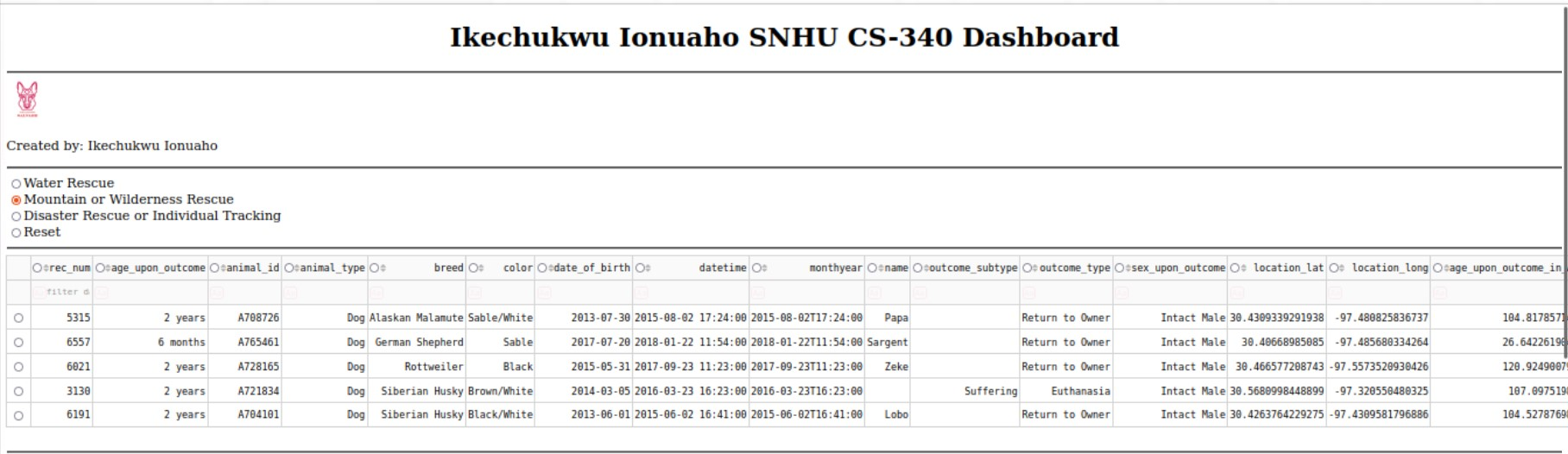


Figure : Mountain/Wilderness Rescue Dashboard

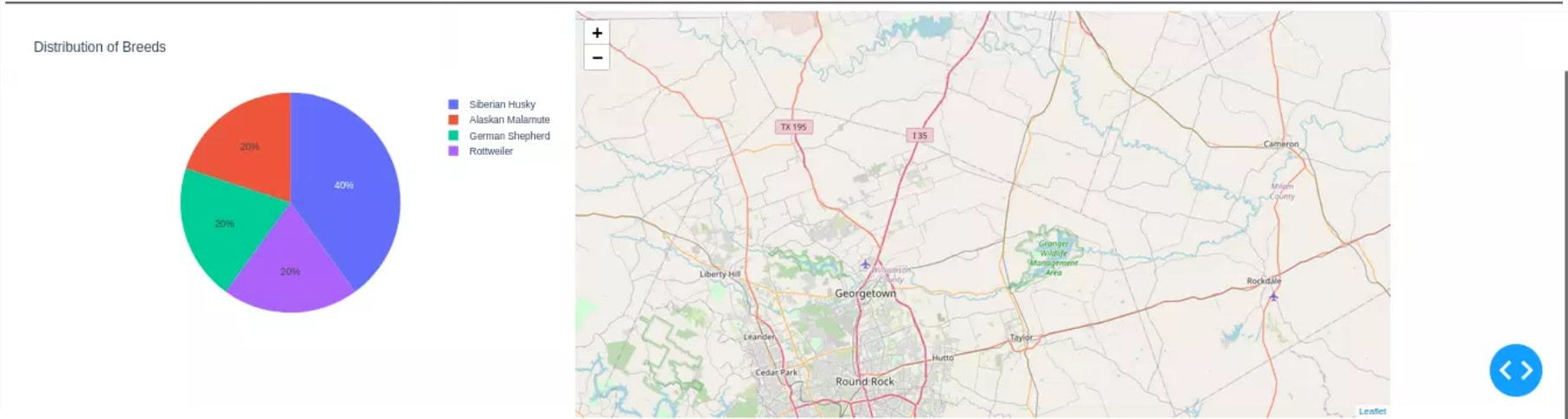


Figure : Mountain/Wilderness Chart and Map

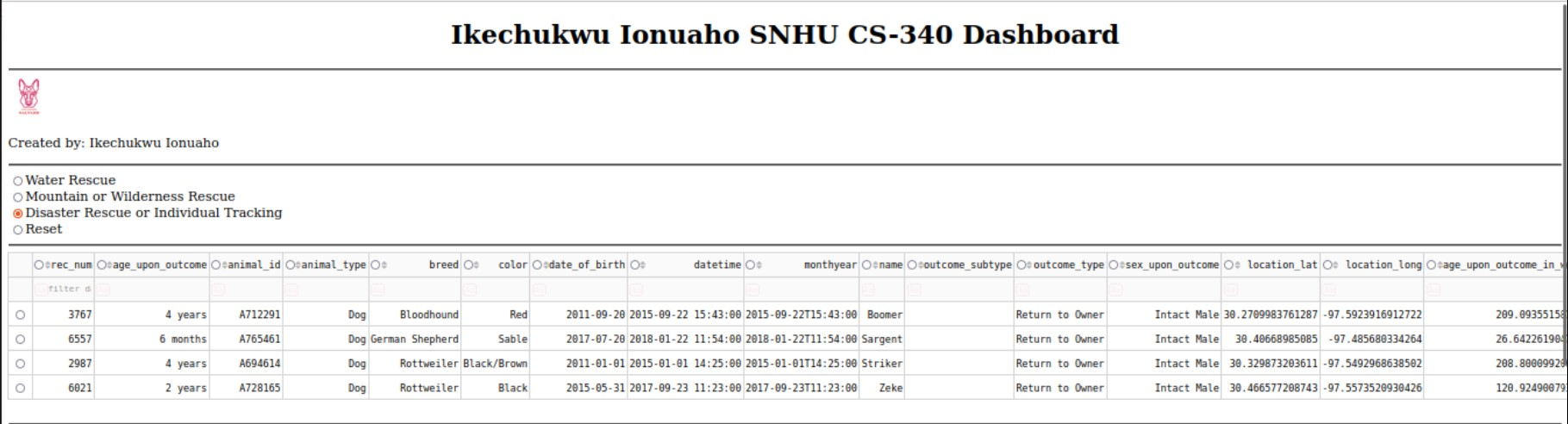


Figure : Disaster Rescue/Individual Tracking Dashboard

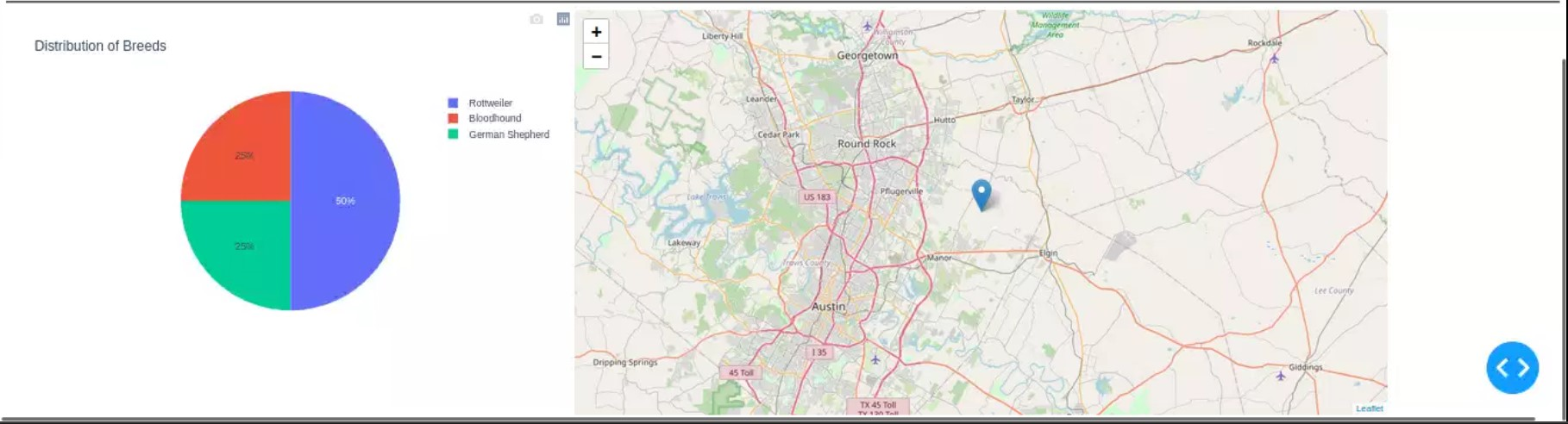


Figure : Disaster Rescue/Individual Tracking Chart and Map