Zachary Fizet

CS-340

Project Two README

6/22/2025

Grazioso Salvare Rescue Dog Dashboard

Overview

This project is a full-stack web dashboard application developed for Grazioso Salvare, an innovative international rescue-animal training organization. The dashboard allows users to explore and visualize data from animal shelters around Austin, Texas, to identify and categorize dogs suitable for search-and-rescue training.

The application connects to a MongoDB database containing dog data, provides interactive filtering options, and dynamically updates data tables and charts based on user selections.

Features

* Interactive Filters (Filters the dataset by rescue types)
  + Water Rescue
  + Mountain or Wilderness Rescue
  + Disaster or Individual Tracking
  + Reset (to view all data)
* Dynamic Data Table
  + Displays detailed dog information responding to the applied filters.
* Visual Charts
  + A geolocation map showing the locations of dogs in the Austin area
  + A pie chart showing the distribution of dog breeds in the filtered dataset.
* Branding
  + Includes the Grazio Salvare logo
  + Developer credits

Technologies Used

* MongoDB
  + MongoDB is a NoSQL document database used as the model component of this project. Its flexible schema and efficient querying capabilities make it ideal for storing and retrieving complex dog profile data. The project uses the PyMongo driver for Python to interact seamlessly with MongoDB.
* Dash by Plotly
  + Dash provides the framework for the web application’s view and controller. It allows rapid creation of interactive dashboards in Python with built-in support for UI components, callbacks, and data visualization, perfect for this MVC-style application.
* JupyterDash
  + A Jupyter-friendly version of Dash enabling the dashboard to be developed and run within Jupyter Notebook environments.
* Pandas & Plotly Express
  + Used for data manipulation and generating charts respectively.
* Dash Leaflet
  + Used to embed interactive geolocation maps displaying dog locations.

Project Structure

* animal\_shelter.py
  + Python module implementing CRUD functionality to connect to MongoDB and run queries.
* ProjectTwoDashboard.ipynb (rename as needed)
  + Jupyter Notebook containing the dashboard layout, interactive filters, callbacks, and visualizations.
* Grazioso\_logo.png
  + Grazioso image logo used on dashboard

Installation & Setup

1. Clone the repository.
   1. git clone https://github.com/zfizetSNHU/CS340/grazioso-salvare-dashboard.git
   2. cd grazioso-salvare-dashboard
2. Install dependencies
   1. Requirements include pymong, dash, jupyter-dash, pandas, plotly, dash-leaflet
3. Configure MongoDB credentials
   1. Update username, password, host, and port in the ipynb file
4. Run the dashboard
   1. Open the ipynb file in Jupyter Notebook and run all cells.

Usage

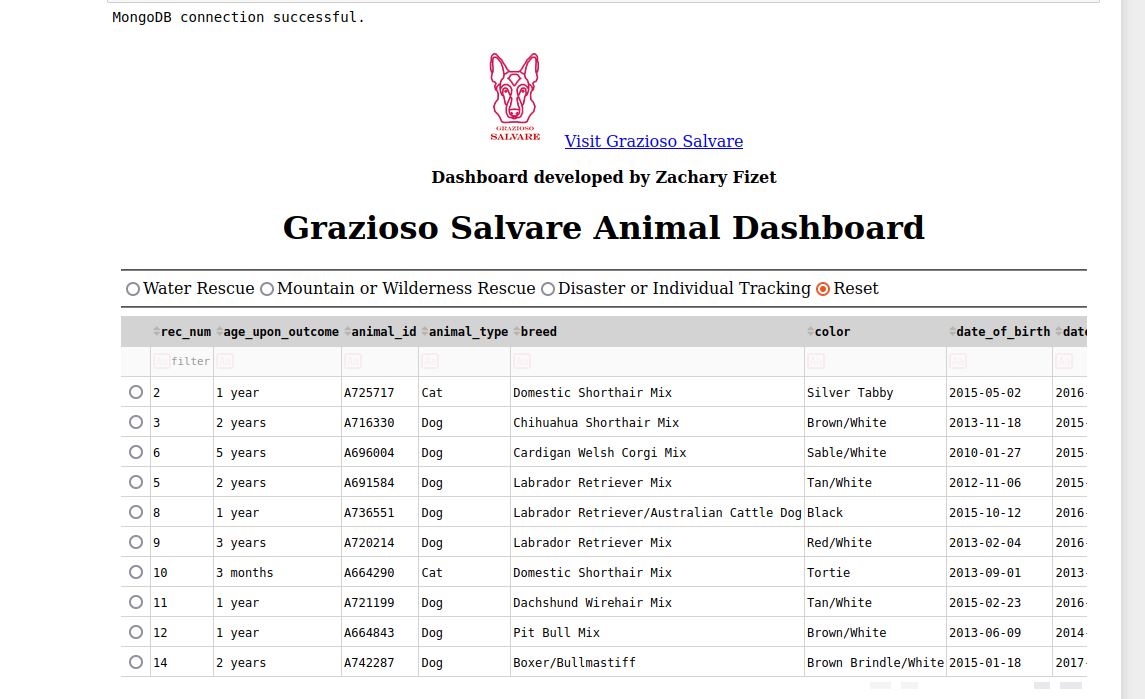
* Use the filter options to select a rescue type or reset all data
* The data table will update to display only dogs relevant to your selected rescue type
* The geolocation map shows markers of dog locations
* The pie chart visualizes breed distribution for the filtered data

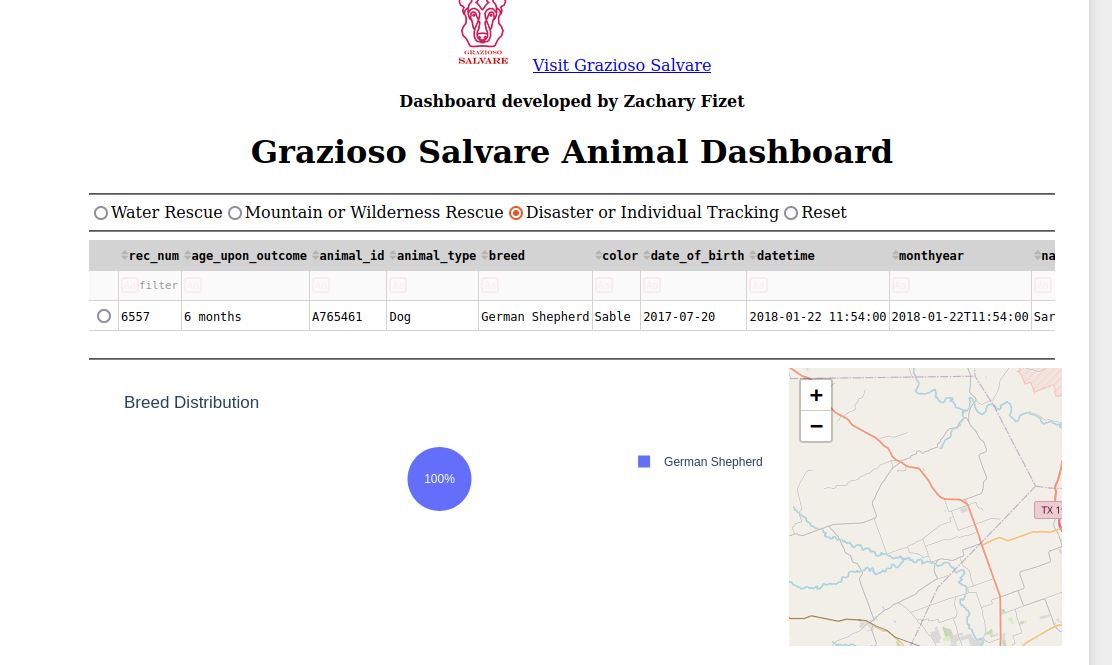
Challenges & Solutions

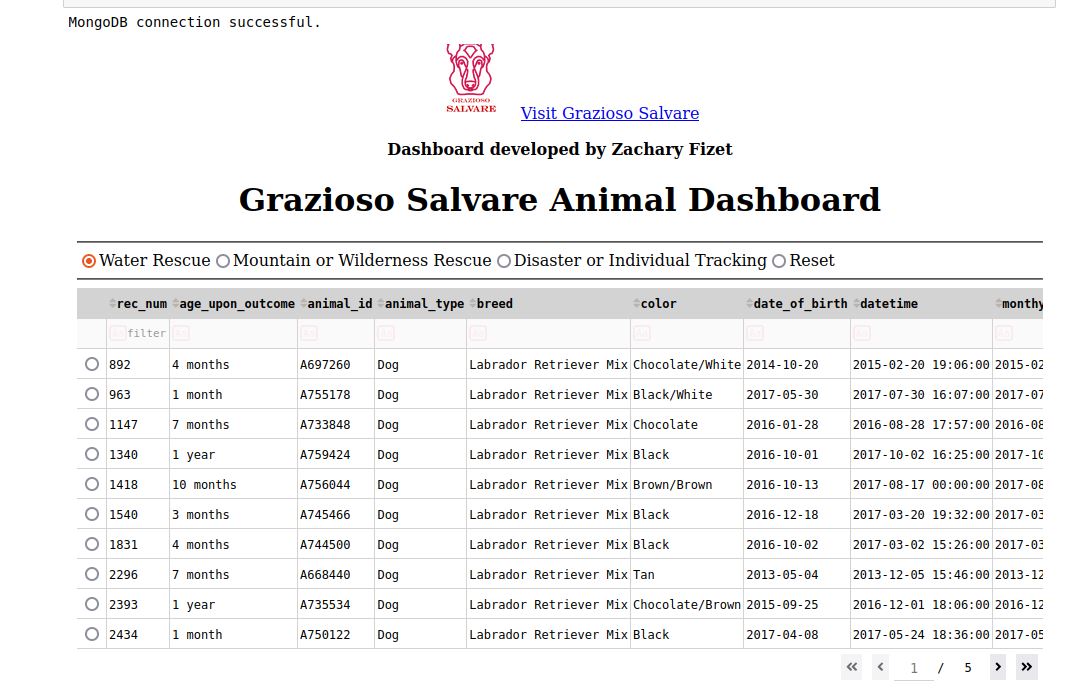
MongoDB connection issues initially proved to be an issue but were overcome by running a simple mongosh command (db.getMongo()) to provide host and port information.

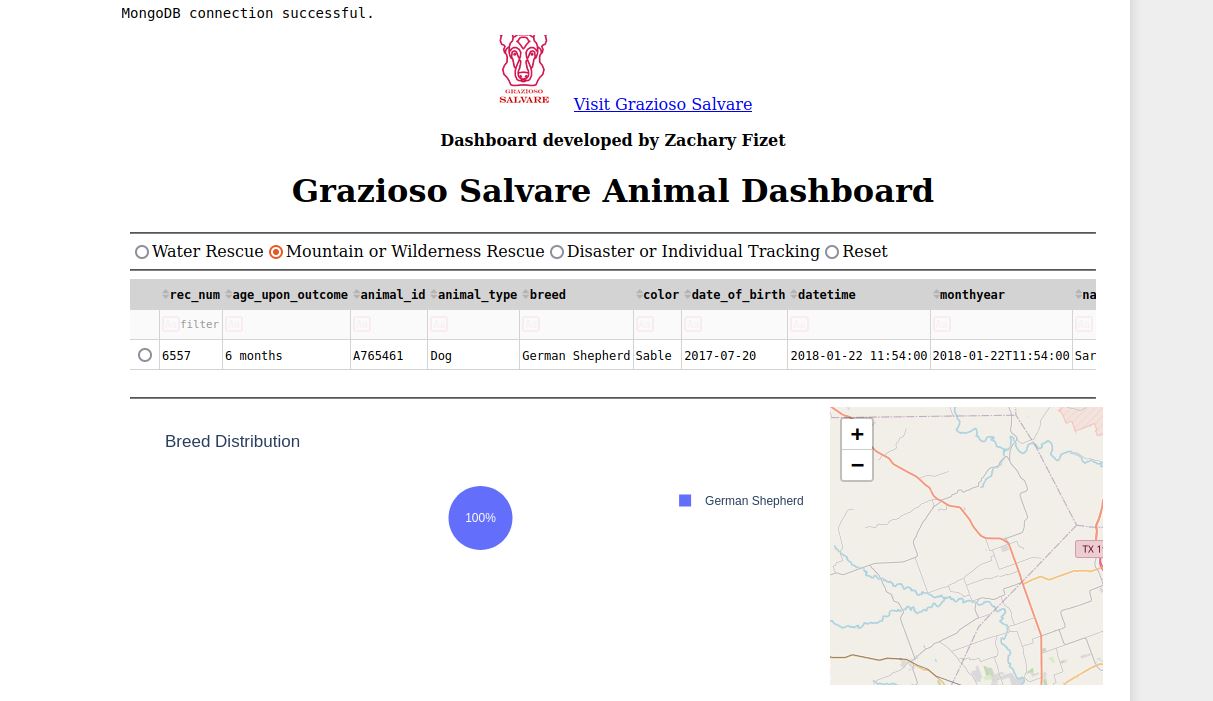
Handling NoneType errors in callbacks. Ensured all Dash callbacks returned valid, non-null values. Added additional error handling in callbacks to avoid problems/crashes.

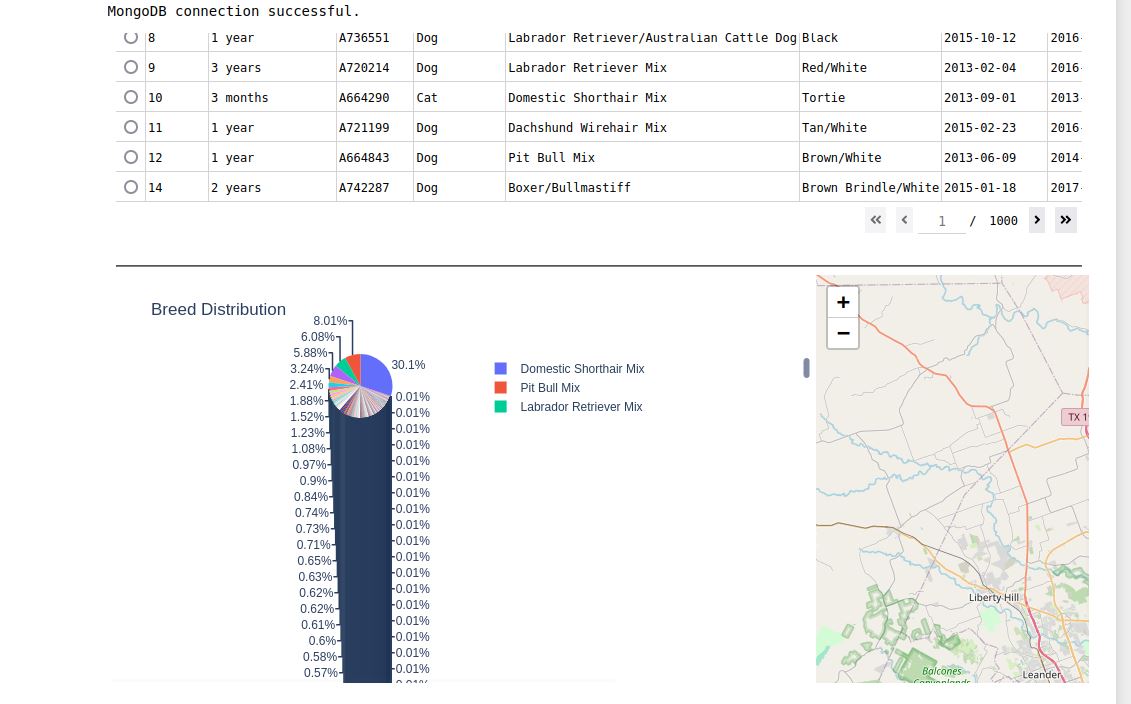
Screenshots

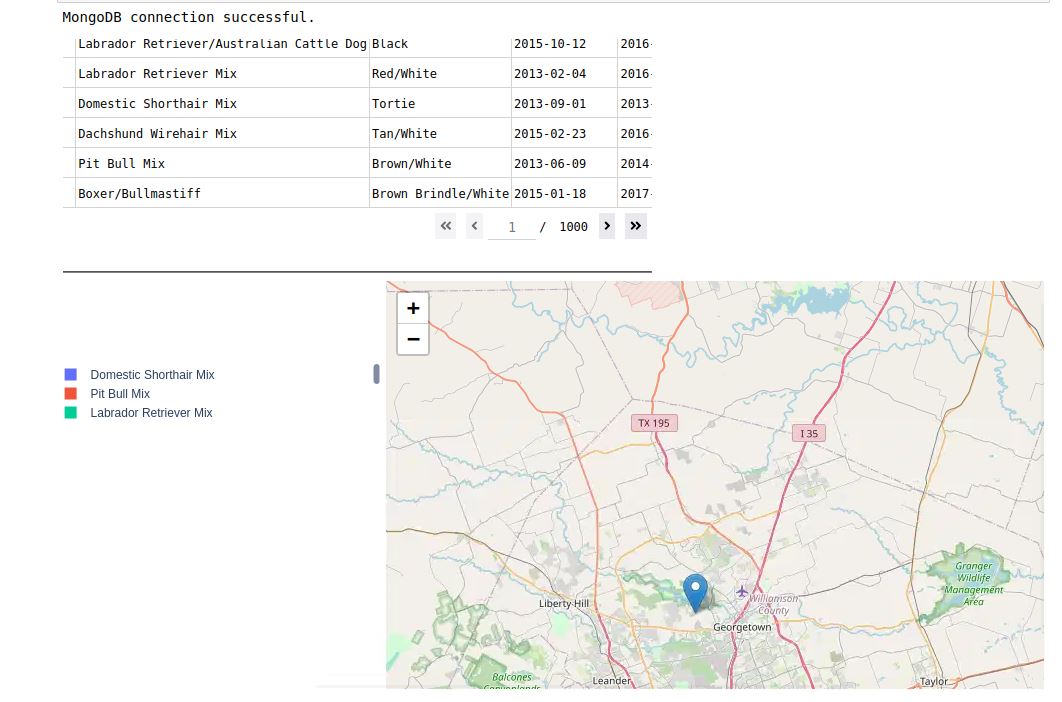


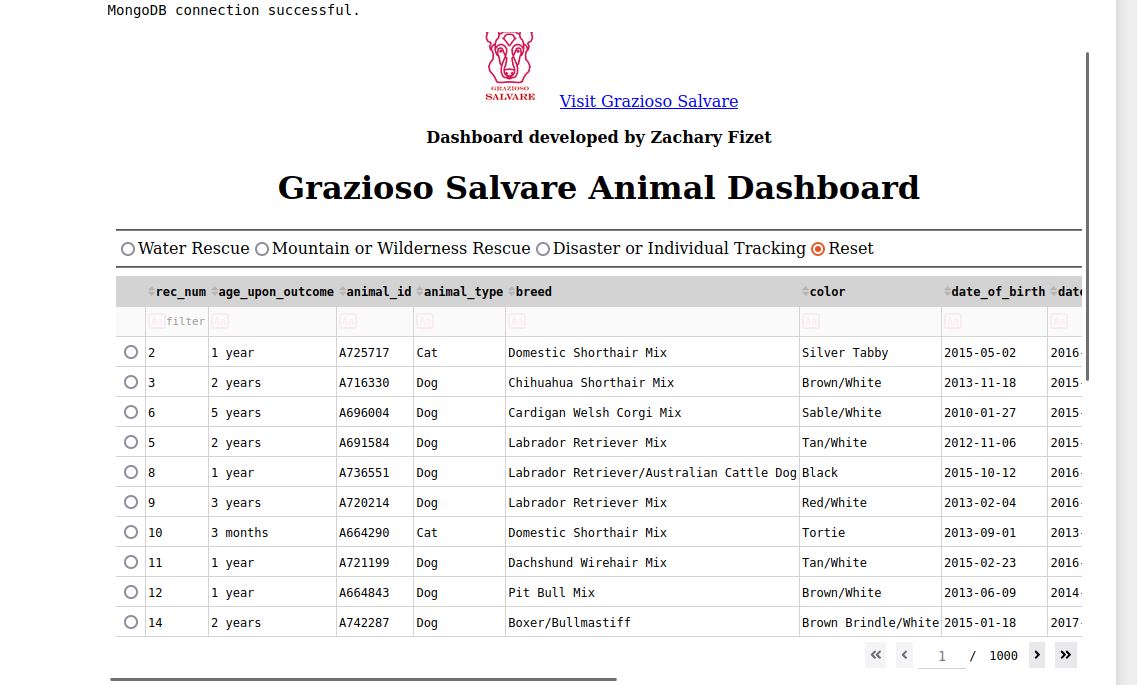












Contact

Developed by Zachary Fizet – zachary.fizet@snhu.edu