# CS 340 README

## Project Overview

This project encompasses the development of an interactive web application using the Dash framework. The application serves as an interactive dashboard for Grazioso Salvare, aiding in the identification and categorization of potential search-and-rescue dogs. The dashboard derives data from animal shelters' repositories.

## Required Functionality

The dashboard includes the following essential functionalities:

**Filter Options:** Users can choose from radio button options to filter dogs based on different rescue types: Water Rescue, Mountain or Wilderness Rescue, Disaster or Individual Tracking. A "Reset" option is available to restore the original view.

**Data Table:** The data table is interactive and responsive to user preferences, including features like limiting rows, pagination, and sorting.

**Charts:** The dashboard includes a pie chart representing the distribution of preferred dog breeds for the selected rescue type. Additionally, a geolocation map displays the selected dog's location on the map.

## Tools Used

To achieve the project's functionality, the following tools and technologies were utilized:

**Dash Framework:** The Dash framework facilitated the creation of an interactive web interface using Python, seamlessly blending view and controller aspects.

**dash\_leaflet:** This library enriched the dashboard by embedding interactive maps, providing users with a visual representation of geolocation data.

**plotly.express:** Employed for data visualization, the plotly.express library enabled the creation of a pie chart that visually communicates breed distribution.

**Pandas Library:** For effective data manipulation and transition between various data formats, the Pandas library was indispensable.

## MongoDB Integration

MongoDB was chosen as the model component due to its adaptability and compatibility with Python. The data from animal shelters is stored in a MongoDB database, and the application interfaces with it using the AnimalShelter Python module.

## Dash Framework

Dash is a web application framework that enables the creation of interactive web applications using pure Python. It encompasses both the view and controller components, enhancing development speed while maintaining a modular structure.

## Steps Taken to Complete the Project

The project was completed through the following stages:

**Database Integration:** The application was linked to the MongoDB database using the AnimalShelter module.

**Dashboard Layout:** HTML and Dash components were utilized to design the dashboard layout. This encompassed branding elements and an exclusive identifier.

**Interactive Filters:** Interactive filtering mechanisms were implemented using Dash's dcc.RadioItems. These options allow users to refine their data preferences.

**Data Table:** The data table created in Step 1 was modified to be interactive and responsive to filter options. Pagination and sorting features were incorporated.

**Charts:** A pie chart was generated using Plotly Express to illustrate preferred dog breed distribution. A geolocation map, powered by dash\_leaflet, was used to show geographical locations.

**Callback Functions:** Callback functions were developed to update dashboard components based on filter selections and row clicks in the data table.

## Challenges and Solutions

There were several challenges during the project:

**Data Querying:** Ensuring accurate data presentation across various filters was a primary challenge. To address this, filters were mapped to the corresponding preferred breeds, sex, and age ranges.

**Chart and Map Integration:** Integrating interactive charts and maps necessitated familiarity with diverse visualization libraries and understanding their input requirements. This was overcome by consulting library documentation and leveraging Dash components' capabilities.

By tackling these challenges and adhering to best practices, the interactive dashboard was successfully created, meeting Grazioso Salvare's requirements.

Contact

Name: Mariam Haji

## Screenshots

Below are screenshots demonstrating the functionality of the interactive dashboard:

## Starting State:

## 

## Water Rescue Filter:

## 

## Mountain or Wilderness Rescue Filter:

## Disaster or Individual Tracking Filter:

## Reset Filters:

