**CS 340 Project Two ReadMe**

Harshilkumar Jayswal

**Project Overview**

The Grazioso Salvare Dashboard project aims to facilitate categorization of dogs suitable for search-and-rescue training by providing a user-friendly web interface. Grazioso Salvare, an international rescue-animal training company, collaborates with a non-profit agency operating animal shelter in the Austin, Texas region to access relevant data. The dashboard allows users to filter dogs based on rescue types, view detailed information about the dogs, and visualize breed distributions and geographic locations.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Tools and Technologies**

* MongoDB serves as the database model for this project. MongoDB is a NoSQL database that provides flexibility, scalability, and ease of integration with Python through libraries like PyMongo. It was chosen for its schema-less design, which allows for dynamic and unstructured data storage. MongoDB's documents and query language make it well-suited for handling and querying diverse datasets.
* Dash is a productive Python framework for building web applications. It enables the creation of interactive, web-based dashboards using Python, HTML, and CSS. Dash was chosen for its simplicity, flexibility, and ability to integrate seamlessly with Python data analysis libraries like Pandas and Plotly.
* Plotly, a graphing library, is used for creating interactive and dynamic visualizations in the dashboard, enhancing data exploration and presentation.

**Project Steps**

* Starting with Python CRUD module developed in earlier modules we have database connection and CRUD functionality.
* Then we use the Python Dashboard testing script to access our database and display information with categorization.

A screenshot of a computer

Description automatically generated

* The Grazioso Salvare logo, title, and author information is displayed using HTML Img and H1 elements.
* Radio buttons (dcc.RadioItems) allow users to filter dogs based on rescue types. Callback functions are defined to update the dashboard based on user selections.
* The dash\_table.DataTable component displays filtered data in tabular format. It enables sorting, filtering, pagination, and row selection for user interaction.

A screenshot of a computer program

Description automatically generated

* The components (dcc.Graph) visualize breed distributions and geographic locations, respectively. Callback functions update these visualizations dynamically based on user interactions with the data table.
* **update\_styles(selected\_columns)** updates the style of selected columns in the data table to highlight them when clicked by the user.
* **update\_breed\_distribution\_chart(data)**Updates the pie chart with the distribution of preferred dog breeds based on the filtered data from the data table.

A screenshot of a computer program

Description automatically generated

* **update\_table(selected\_type)** Updates the data displayed in the data table based on the selected rescue type filter.
* **update\_map(viewData, selected\_rows)** Updates the geographic map with the location of selected dogs from the data table.

**Challenges and Solutions**

* During the pie chart visualization, the data labels were scattered causing the chart to be very distracting and hard to interpret. This was resolved with update\_traces(textposition='inside') which adjusts the position of text labels inside pie chart slices for better visibility.
* In the database query, $in was resulting in only the exact matches of the breed, leaving the mixed breeds from getting selected. So $regex operator was used since allows you to search for documents where a specific field matches a regular expression pattern, making it useful for finding documents based on incomplete or partial information.

**References**

Learning Python. Python tutorial. (n.d.). https://www.w3schools.com/python

PEP 8 – style guide for python code. Python Enhancement Proposals (PEPs). (n.d.). https://peps.python.org/pep-0008/