**README for Grazioso Salvare Dashboard**

**Project Overview**

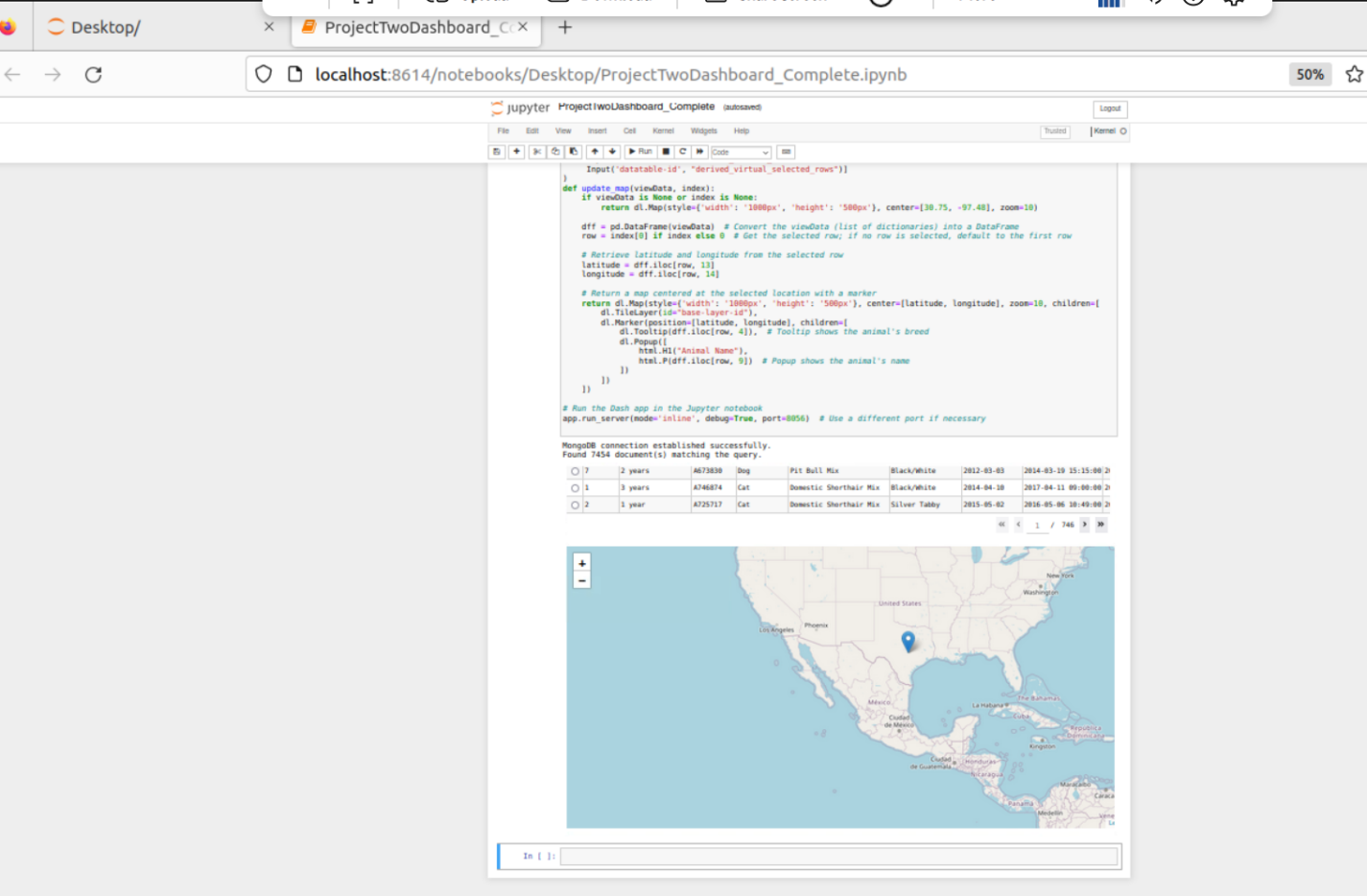
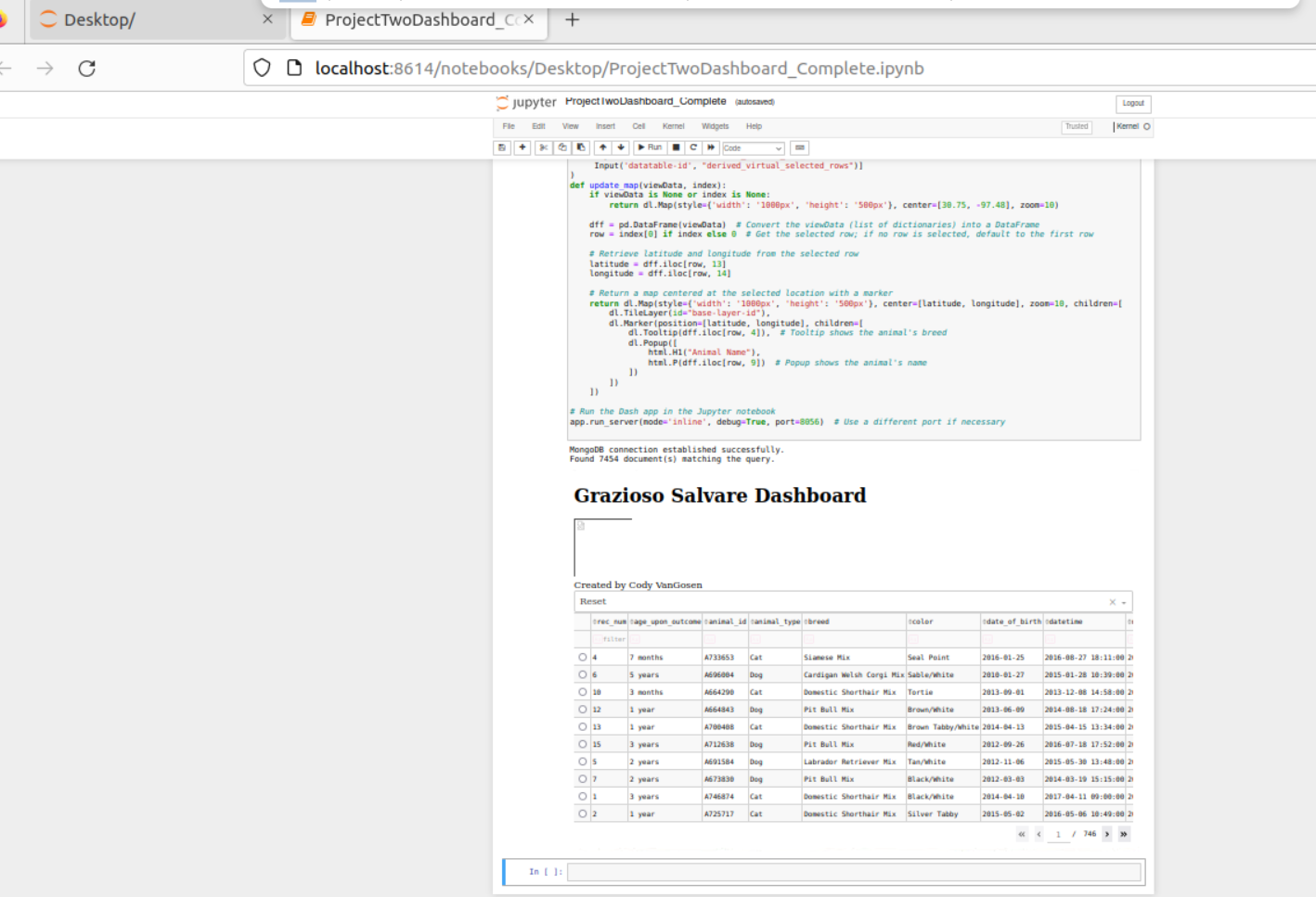
The Grazioso Salvare Dashboard is an interactive web application designed to visualize data from the Austin Animal Center Outcomes dataset. The dashboard provides users with filtering options to explore various animal outcomes based on rescue types, enabling insights into animal adoptions and rescues.

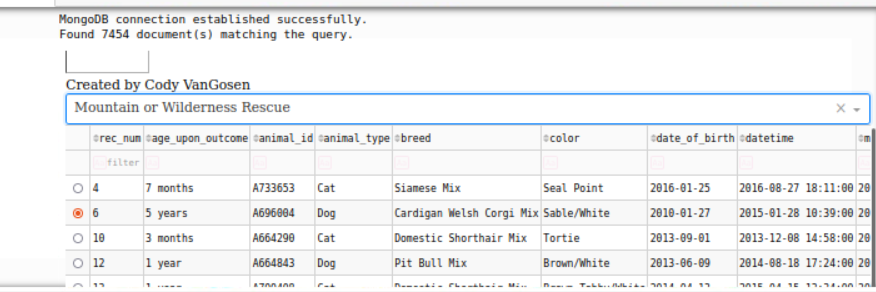
**Required Functionality**

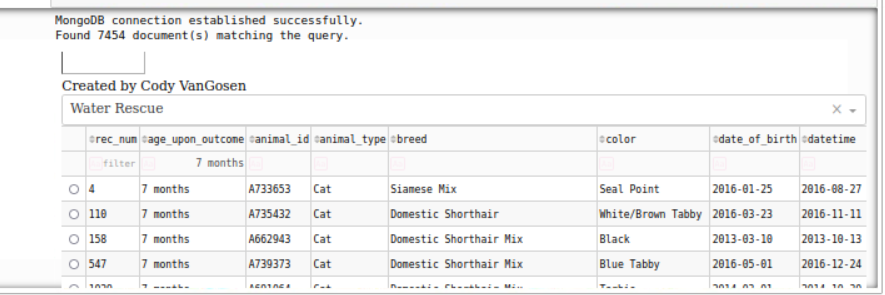
The project achieves the following functionalities:

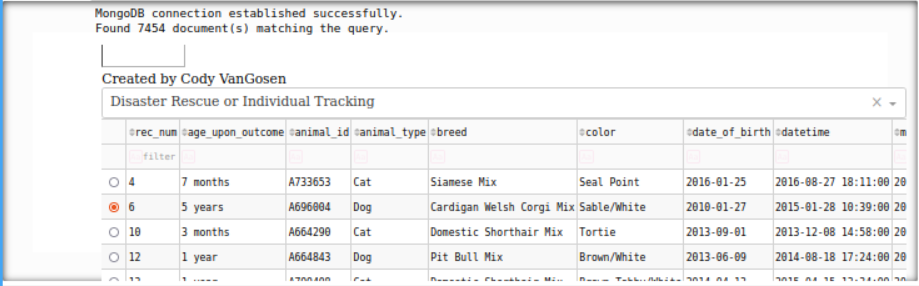
1. **Data Visualization**: Users can view a dynamic data table that displays the animal shelter outcomes.
2. **Filtering Options**: The dashboard includes interactive filtering options, allowing users to filter data by:
   * Water Rescue
   * Mountain or Wilderness Rescue
   * Disaster Rescue or Individual Tracking
3. **Geolocation Mapping**: The dashboard features two maps:
   * A geolocation chart that displays markers for animal outcomes based on the selected filter.
   * An additional map to visualize another aspect of the data (e.g., breed distribution).
4. **Responsive Design**: The dashboard layout adapts to various screen sizes for optimal viewing.

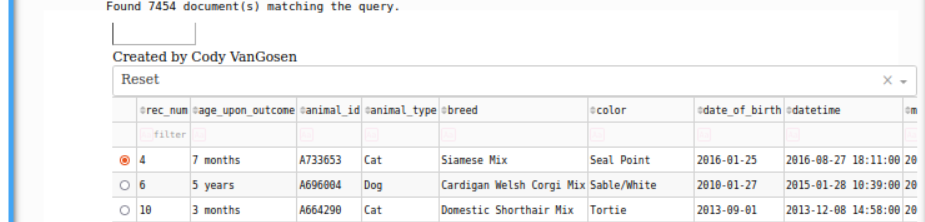
**Screenshots**

**  
*Figure 1: Default state of the dashboard displaying all animals.*

  
*Figure 2: Dashboard state after filtering by Mountain or Wilderness Rescue.*

  
*Figure 3: Dashboard state after filtering by Water Rescue.*

  
*Figure 4: Dashboard state after filtering by Disaster Rescue or Individual Tracking.*

  
*Figure 5: Dashboard state after resetting all filters.*

**Tools Used**

* **Jupyter Dash**: Used for creating the dashboard interface. The Jupyter Dash framework integrates seamlessly with Jupyter Notebook, allowing for easy prototyping and real-time updates.
* **Dash**: A Python framework for building web applications. It provides the structure necessary for creating interactive web interfaces by separating the user interface (view) from business logic (controller).
* **Plotly**: Utilized for creating interactive plots and maps. Plotly’s capabilities enhance the visualization aspects of the dashboard, making data exploration more intuitive.
* **MongoDB**: A NoSQL database used to store and manage the animal shelter data. Its document-based structure is flexible for unstructured data and allows for efficient querying and retrieval of complex datasets.

**Rationale for Tool Selection**

MongoDB was chosen for its ability to handle large volumes of unstructured data, which is common in animal shelter records. Its flexible schema allows for easy adjustments to the data structure without extensive migrations. Additionally, its Python driver integrates smoothly with the application, facilitating easy data manipulation.

The Dash framework provides a robust platform for developing interactive web applications. Its components enable the easy creation of dashboards and visualizations, essential for the Grazioso Salvare project.

**Resources**

* [Plotly Documentation](https://dash.plotly.com)
* [MongoDB Documentation](https://docs.mongodb.com/)
* [Jupyter Documentation](https://docs.jupyter.org/en/latest/)
* [Pandas Documentation](https://pandas.pydata.org/docs/)

**Steps to Complete the Project**

1. **Setup Environment**: Installed necessary libraries including Dash, Plotly, and MongoDB driver.
2. **Design Database Schema**: Created a suitable schema for the Austin Animal Center Outcomes data in MongoDB.
3. **Develop CRUD Operations**: Implemented a CRUD module to handle database operations.
4. **Create Dashboard Layout**: Defined the layout of the dashboard using Dash components.
5. **Implement Callbacks**: Developed callback functions to handle user interactions and data filtering.
6. **Test Functionality**: Tested the dashboard to ensure all functionalities work as intended.

**Challenges Encountered**

During the development process, several challenges arose:

1. **Database Connection Issues**: Initial attempts to connect to MongoDB were met with authentication errors. This was resolved by verifying the connection parameters and ensuring the correct credentials were used.
2. **Data Formatting**: Some fields in the dataset contained unexpected values (e.g., negative sizes). This was addressed by implementing data validation checks during the data retrieval process.
3. **Interactivity**: Ensuring the dashboard remained responsive while processing filters required optimization of the callback functions.

**Conclusion**

The Grazioso Salvare Dashboard successfully meets the project's requirements by providing an interactive platform for exploring animal outcomes. The implementation of filtering options and geolocation mapping facilitates a deeper understanding of shelter data, contributing positively to animal welfare efforts.