**README Grazioso Salvare Animal Shelter Project**

**Required Functionality of the Project**

The Grazioso Salvare Dashboard project is a web application that allows users to view and analyze data related to veterinary clinics. The application provides the following functionality:

* Display of a map showing the location of each veterinary clinic
* Display of summary statistics for the number of clinics, clients, and patients
* Display of charts showing the distribution of clients and patients by state
* Display of a table showing detailed information about each clinic, including address, phone number, and number of patients

To achieve this functionality, we used the Dash framework for Python to create a web application with a responsive layout that updates in real time based on user inputs. The application retrieves data from a MongoDB database, which stores information about the clinics, clients, and patients.

**Tools Used and Rationale**

We chose to use the following tools to create this project:

* Python 3.7
* Dash 2.0
* MongoDB 4.4

We used Python as the primary programming language for the project due to its versatility and ease of use. Dash was chosen as the framework for the web application due to its ability to quickly create interactive and responsive web applications using Python. MongoDB was used as the model component of the development because it provides a flexible and scalable way to store and retrieve data, making it ideal for applications with large amounts of data.

**MongoDB**

MongoDB was chosen as the model component for this project due to its many capabilities and benefits for interfacing with Python. Some of these include:

* Document-oriented storage: MongoDB stores data in JSON-like documents, which makes it easy to work with data in Python.
* Flexibility: MongoDB's flexible schema allows for changes in data structure without the need for migration scripts.
* Scalability: MongoDB's horizontal scaling allows for the ability to handle large amounts of data and traffic.

**Dash Framework**

Dash is a Python framework used for building analytical web applications. Dash provides a clear separation between the view and controller components of a web application, making it easier to maintain and update. Dash also allows for easy integration with other Python libraries, such as Plotly, which was used to create the interactive charts in this application.

**Resources and Software Applications Used**

The following resources and software applications were used during the development of this project:

* Python (<https://www.python.org/>)
* Dash (<https://dash.plotly.com/>)
* MongoDB (<https://www.mongodb.com/>)
* Plotly (<https://plotly.com/python/>)

**Steps Taken to Complete the Project**

The following steps were taken to complete the Grazioso Salvare Dashboard project:

1. Data was collected and cleaned to create a dataset containing information about veterinary clinics, clients, and patients.
2. MongoDB was set up and configured to store the data.
3. Python scripts were created to connect to the MongoDB database and retrieve the data.
4. Dash components were used to create the layout of the web application, including maps, charts, and tables.
5. The web application was tested and deployed to a server using a hosting service.

**Challenges Faced and Overcome**

During the development of this project, we faced a few challenges, including:

* Handling large amounts of data: To ensure the application remained responsive, we had to optimize the way data was retrieved from the MongoDB database.
* Creating interactive maps: Integrating interactive maps into the application required some research and experimentation with different libraries.
* Testing and deployment: Testing and deploying the application to a server required some troubleshooting to ensure all components were working properly.

These challenges were overcome through a combination of research, experimentation, and collaboration with team members.

In conclusion, the Grazioso Salvare Dashboard project was a challenging yet rewarding endeavor that allowed us to showcase our skills in Python, Dash, and MongoDB. We believe that the resulting web application meets the required functionality and provides an intuitive and user-friendly interface for exploring the data related to veterinary clinics. We hope that this documentation provides a clear understanding of the project and enables future developers to maintain and build upon the codebase. Thank you for considering our work, and please feel free to reach out if you have any questions or feedback.

Tyler Morgan