# Animal Shelter Database README

## About the Project/Purpose

Grazioso Salvare, an innovative international rescue-animal training company, has tasked us at Global Rain software engineering to design a dashboard which displays data relating to animals in their custody. Grazioso Salvare will use the dashboard to identify dogs that are good candidates for search-and-rescue training based on their breed, sex, and age. The project is a full-stack solution to allow the company to handle all of their data and allow for ease of use. The dashboard has three key features which include a data table which provides all data relating to the animals, a chart to identify percentages of breeds available, and a map showing the location of a specific animal.

**Installations:**

The tools needed for this project include MongoDB, Python, Jupyter Notebooks, Dash, and Plotly:

MongoDB: Installation instructions can be found at <https://docs.mongodb.com/manual/installation/>.

Python: Installation instructions can be found at <https://realpython.com/installing-python/>.

Jupyter Notebooks: Jupyter can be installed from the command line using <https://jupyter.org/install>. More instructions can be found at: <https://jupyterlab.readthedocs.io/en/stable/getting_started/installation.html>

Dash: Installation instructions can be found at <https://pypi.org/project/dash/>. Importing Dash Core Components into Jupyter Notebook is essential to creating this project.

Plotly: This is essential to creating charts in Python applications. This can be installed using <https://www.journaldev.com/19692/python-plotly-tutorial#:~:text=Installation.%20To%20install%20plotly%2C%20open%20a%20terminal%20window,to%20install%20to%20collect%20dependencies%20and%20download%20them%3A>

## Getting Started

MongoDB, a source-available document-oriented database program, was used for the backend of the project. We chose MongoDB because it allows for a quick setup of databases and collaborates well with Python and Dash framework. The front-end was developed using Dash through Jupyter notebook, a web-based interactive computing program. Dash is an open-source Python framework which allows for easily building data visualization interfaces, such as the dashboard in this project.

To get a local copy of this program up and running, you will need to follow these steps:

1. Create a database in MongoDB by importing the csv file and create authenticated user(s)
   * Open terminal window and start MongoDB using *usr/local/bin/mongod\_ctl start*
   * Text

     Description automatically generatedImport csv file as shown below using your port number:
   * Create user with read/write permissions and enable user authentication:

Text

Description automatically generated

1. Download CRUD python file

* Install animalShelter.py file into directory
* Change the following line to your own port:

*self.client = MongoClient(‘mongodb://%s:%s@localhost:38888/AAC’ % (username, password))*

**Code example:**

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated



1. Download Dash framework
   * Install ProjectTwoDashboard.ipynb and place in directory with animalShelter.py
   * Ensure the CRUD module is imported properly as shown below:

Logo

Description automatically generated with low confidence

* + Change username and password to your created user/pass:

Text

Description automatically generated with low confidence

The Dash framework uses html and Python to create the dashboard. After importing the CRUD module and updating the username and password, we first had to create the data table layout as shown below:

Text

Description automatically generated with low confidence

Then, we had to create the radio items for the dashboard within the app.layout as shown below:

A picture containing text

Description automatically generated

After that, we had to create an app.callback method to instantiate the radio buttons. Personally, I had trouble with this aspect of the module. I had trouble getting the functionality of the radio buttons to work because of the way it was required to type it out. After testing each value, I was able to determine how to get the radio buttons to work efficiently. Example code is shown below:

Text, timeline

Description automatically generated

Graphical user interface, text

Description automatically generated

Finally, we were required to develop code to create a chart and a map.

Chart:

Graphical user interface, text, application

Description automatically generated

Map:

Text

Description automatically generated

## Usage:

After successfully creating a MongoDB database and downloading the animalShelter.py and ProjectTwoDashboard.ipynb modules, you will be able to run this program.

Open the ProjectTwoDashbard.ipynb file and run it. You will see this:

Graphical user interface, application, table

Description automatically generated

Graphical user interface, application

Description automatically generated

To sort through the Different rescue types, the user can select one of the four radio buttons at the top of the dashboard. This will change the data table, chart, and map according to the rescue type that it selected. The default setting is “Reset” so it shows the entire database without filtering.



## Functionality Examples:

Selecting Water Rescue option:

Table

Description automatically generated

Graphical user interface, application

Description automatically generated

Selecting Mountain/Wilderness Rescue:

Table

Description automatically generated

Graphical user interface, application, Excel

Description automatically generated

Selecting Disaster/Individual Tracking:

Table

Description automatically generated

Graphical user interface, application

Description automatically generated

Selecting Reset button:

Graphical user interface, text, application

Description automatically generated

Graphical user interface, application

Description automatically generated

## Contact

Jaelyn Sloan