A dog with a black background

Description automatically generated

**Web Application Dashboard**

**By: Emalee Mendoza**

**About the project**

Grazioso Salavre commissioned this project to provide their organization with a web-based dashboard to expertly navigate the databases of animal shelters to find the animals for their specific needs. This project utilizes MongoDB for the raw, a CRUD python module to manage the data, and Plotly-Dash application to organize everything.

**Motivation**

Grazioso Salavare identifies dogs that are good candidates to be trained in search and rescue. They have partnered with a non-profit agency that operates five shelters in the Austin, TX area and have received data on the shelter’s occupants. The organization is seeking software to assist them with filtering through the data to find dogs that meet their specific training criteria, such as dogs no more than two years old, and specific breeds for different tasks. This software should be open source and available on GitHub so that it may be utilized by similar organizations.

**Tools Used**

* Jupyter Notebooks: <https://jupyter.org/>
* Python: <https://www.python.org/>
* MongoDB: <https://www.mongodb.com/>
* Plotly-Dash: <https://dash.plotly.com/>

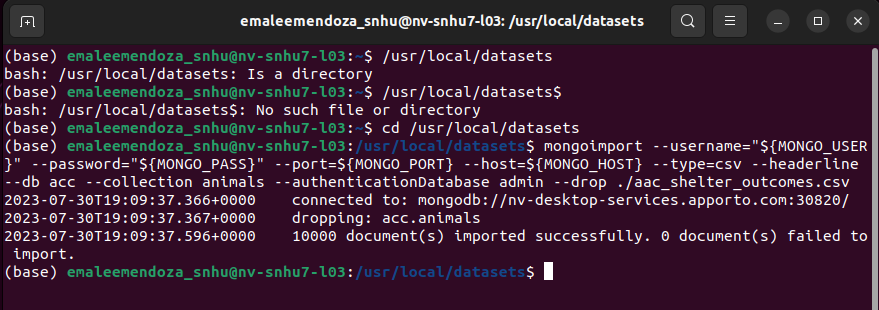
These tools were specifically chosen for this task because of their unique characteristics and ease of use. MongoDB was chosen because it is cost effective, has powerful querying abilities, and is easily accessible. Python was chosen because it works well for data analytics, has extensive libraries, and is easy to read and write. Jupyter Notebook was used to develop the code and test the script. Dash by Plotly offered great compatibility with Python and pre-built interactive components, such as graphs and maps.

**Getting Started**

1. Create an environment by uploading a test database.
2. Create an authentication admin account.
3. Create and test a Python CRUD module.
4. Create the dashboard components.
5. Run the application on the web.

**Usage**

Import the test database:



Create the admin account:

A screenshot of a computer program

Description automatically generated

Develop and test the CRUD module:



Example of testing the READ functionality:

A screenshot of a computer

Description automatically generated

Dashboard implementation:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Project Challenges**

I was not able to implement all features of the dashboard, I was getting an error: “Callback error updating map-id.children”. The geolocation feature would not display.

A screenshot of a computer

Description automatically generated

**Contact**

Emalee Mendoza | Emalee.mendoza@snhu.edu