README

Chris Richards  
[Christopher.richards4@snhu.edu](mailto:Christopher.richards4@snhu.edu)  
06/20/2023

Animal Shelter Dashboard

The Animal Shelter Dashboard is a web application that allows users to interact with animal shelter data. It provides functionalities to filter and visualize the data, view it in a table, and explore the geographical distribution of the animals on a map.

To use the Animal Shelter Dashboard, follow the steps below:

Prerequisites

Python 3.7 or above installed on your machine

pip package manager

Installation Steps

Clone or download the project repository to your local machine.

Navigate to the project directory in your terminal:

Install the required dependencies by running the following commands one by one:

pip install dash

pip install pandas

pip install plotly

pip install pymongo

pip install dnspython

The above commands will install the necessary packages for running the Animal Shelter Dashboard.

Note: If you encounter any issues during the installation, make sure you have the correct permissions or consult the documentation for each package for further troubleshooting.

Usage

Run the application:

python app.py

Open a web browser and go to http://localhost:8050 to access the Animal Shelter Dashboard.

Use the dashboard to interact with the animal shelter data.

Filtering Options

You can filter animals by the intended rescue activity.

Chart

The chart section of the dashboard displays a pie chart that visualizes the distribution of dogs based on their breeds. Each pie piece represents a breed, and the size of the pie piece represents the number of animals of that species in the shelter.

Hover over the pie piece to see the exact count for each breed.

Map

The map section of the dashboard shows the geographical distribution of the dogs in the shelter. Each marker on the map represents a dog, and its location corresponds to the latitude and longitude data.

Click on a marker to view information about the dog.

AnimalShelter Module

The AnimalShelter module (AnimalShelter.py) provides a convenient interface for performing CRUD (Create, Read, Update, Delete) operations on a MongoDB database. It allows you to interact with the animal shelter data and perform operations such as creating new records, retrieving records based on specific criteria, updating existing records, and deleting records from the database.

To use the AnimalShelter module, follow these steps:

Import the module into your Python script:

**from** AnimalShelter **import** AnimalShelter

Create an instance of the AnimalShelter class, providing the necessary credentials and connection details:

username **=** "your\_username"

password **=** "your\_password"

shelter **=** AnimalShelter**(**username**,** password**)**

Use the various methods of the AnimalShelter class to perform operations on the animal shelter data:

# Create a new record

data **=** **{**"name"**:** "Max"**,** "species"**:** "dog"**,** "breed"**:** "Labrador Retriever"**,** "age"**:** 24**}**

shelter**.**create**(**data**)**

# Retrieve records based on criteria

criteria **=** **{**"species"**:** "cat"**}**

cats **=** shelter**.**read**(**criteria**)**

# Update existing records

update\_data **=** **{**"age"**:** 36**}**

criteria **=** **{**"species"**:** "dog"**,** "breed"**:** "Labrador Retriever"**}**

shelter**.**update**(**update\_data**,** criteria**)**

# Delete records based on criteria

delete\_criteria **=** **{**"species"**:** "bird"**}**

deleted\_count **=** shelter**.**delete**(**delete\_criteria**)**

For detailed documentation and usage examples, refer to the comments within the AnimalShelter.py file.

Dashboard Code

The dashboard code (app.py) is responsible for creating the web application using the Dash framework. It defines the layout of the dashboard, including filters, tables, graphs, and maps, and specifies the callbacks that handle user interactions and update the visualizations accordingly.

You can customize the dashboard code to fit your specific requirements. Refer to the Dash documentation for more information on how to build and extend the functionality of the dashboard.