# CS 340 README

## About the Project/Project Title

*This project is a software application that works with existing data from animal shelters to identify and categorize available dogs. It allows to display and filter the data through a CRUD python module interacting with a database and a Dashboard user interface through the Dash framework.*

## Motivation

*The project provides a portable module that enables CRUD functionality to a web application that connects a client-side user interface to a database. It allows filtering data to identify dogs for rescue missions’ training.*

## Getting Started

*Start by uploading a data file into MongoDB by using the MongoDB import tool.*

## Installation

*This project uses Python, MongoDB and Dash framework.*

*Refer to the installation guide and documentations:*

*Python:* [*https://www.python.org/downloads/*](https://www.python.org/downloads/)

*MongoDB:* [*https://docs.mongodb.com/manual/installation/*](https://docs.mongodb.com/manual/installation/)

*Dash Python user guide:* [*https://dash.plotly.com/*](https://dash.plotly.com/)

## Usage

### Code Example 1

def create(self, data):

if data is not None:

self.database.animals.insert(data) # data should be dictionary

print('True')

else:

raise Exception('False')

*This code example shows the function to create a new data entry in the database. It returns True when the insert is successful and False otherwise.*

### Tests

*To run tests on the code example, create a Jupyter Notebook. Start by importing the Python module. Create an AnimalShelter object and make sure to hardcode the username and password for the database (as shown in the screenshot below). Use the functions create() and read() to test the insert and find operations.*

### A picture containing text Description automatically generatedCode Example 2

*This code example shows the filtering options through a dropdown menu.*

*Graphical user interface, text

Description automatically generated****Code Example 3***

*This example shows the code to populate the pie chart from the data table with extra filtering options. Screenshots for tests on these features can be seen below in the next section.*

### Graphical user interface, application Description automatically generatedScreenshots

Graphical user interface, application

Description automatically generatedThis screenshot shows the initial display of the application before any user input. It shows Grazioso Salavare’s logo, displays the data imported from the database (10 items per page), a dropdown to select filtering options, and a pie chart and geolocation chart side by side.

This screenshot shows the dropdown menu with the filtering options: Water rescue, Mountain rescue, Disaster rescue and Reset.Application

Description automatically generated with low confidence

Graphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generatedThis screenshot shows the application with Water Rescue filter applied. The pie chart and the geolocation chart update accordingly.

This screenshot shows the application with the Mountain Rescue filter applied and updated charts.

Application

Description automatically generated with medium confidenceThis screenshot shows the application with the Disaster Rescue filter applied and updated charts.

When selected, the Reset dropdown choice brings back the original unfiltered data.

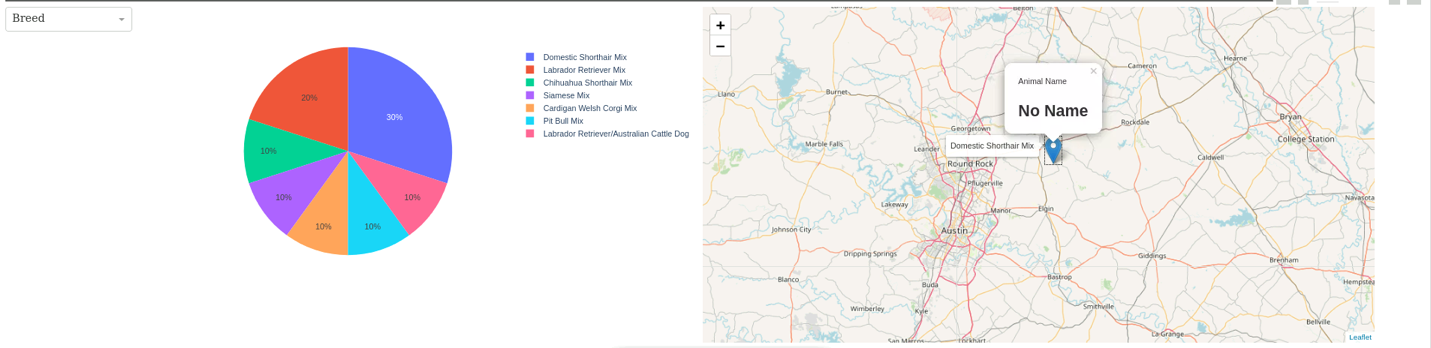
Graphical user interface, map

Description automatically generatedThe charts are interactive, and they update with different filtering options. This screenshot shows charts for the original unfiltered data set.

A picture containing map

Description automatically generatedThis screenshot shows an example of interactivity. The pie chart displays the different breeds in the category “Mountain Rescue” and their numbers in the current page displayed.

## Roadmap/Features (Optional)

*An extra feature is to filter available dogs in the pie chart not only by breed, but also by age and outcome type.*

*Graphical user interface

Description automatically generated with medium confidence  
Graphical user interface

Description automatically generated with medium confidence*

## Contact

Your name: Salah Bedoui