# CS 340 README Template

## About the Project/Project Title

Creatign dashboard and database interface logic

## Motivation

The motivation for this project is to work through a scenario to supply a customer with a product that meets their needs. In this scenario, the customer is looking to have a dashboard that will interact with an animal shelter database. They would like to be able to sort and find animals based on their ability to be rescue animals. They would also like each animal’s location to be displayed and a chart to show the percentage of each breed in the rescue filter. This will involve creating a front-end application that utilizes MongoDB for the customer to accomplish their goals.

## Getting Started

To get started with using this, the connection variables need to be changed to the values that relate to your MongoDB environment. These variables are the username and password that will connect to the database. If an account is not created to connect to the database than one will need to be created and given access. Once that is done, the program can be run as is or modified to produce more specific results. To accomplish this, radio buttons can be added that will filter on the criteria that is needed.

## Installation

Python 3 is required

A PyMongo driver is also required to run this.

Dash framework to utilize the application functions.

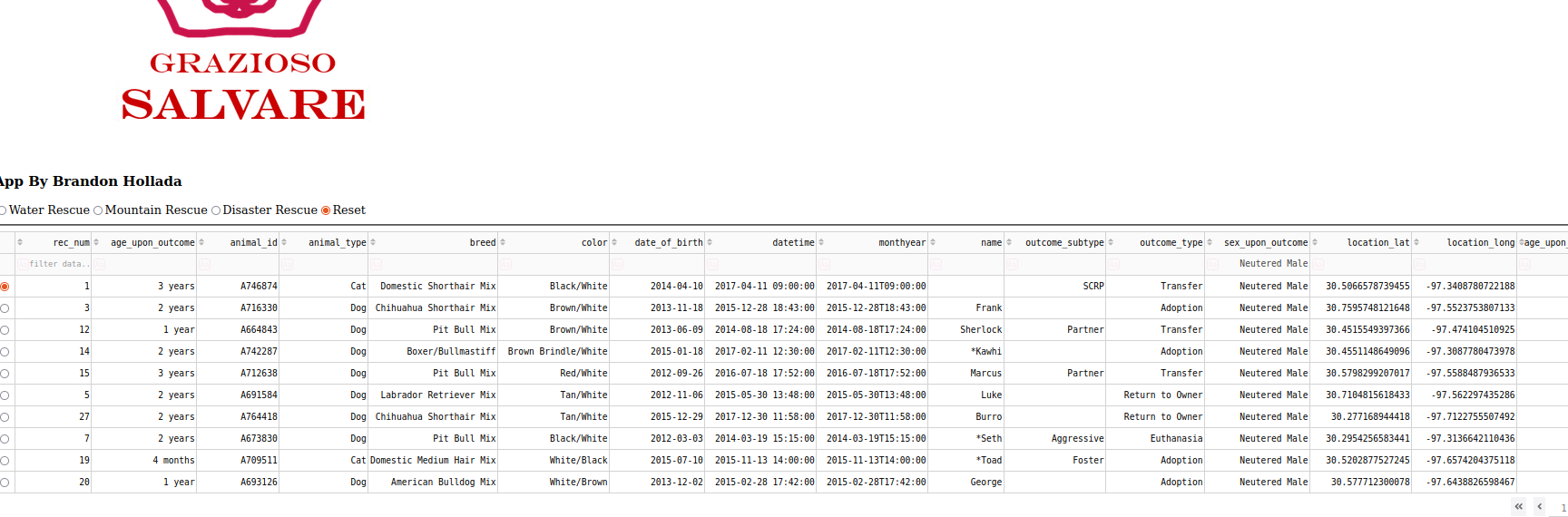
## Usage

The application is used to access a database to present a table of the data like below

A computer screen shot of a computer

AI-generated content may be incorrect.

From this data the user should be able to sort the fields based on information that may be important. An example might be that they are looking for animals that are neutered.



Another way that this can be utilized is to find animals that can be used for rescues. The user may be in need of animals that are good for mountain rescues. This is specific breeds in a specific age range. Another benefit is that once an animal is selected, it will show where is was found with a pin on a map.

A screenshot of a computer

AI-generated content may be incorrect.

A map with a blue pin on it

AI-generated content may be incorrect.

The last major use of the application is to determine what breed has the most or least available animals for the type of rescue that is needed. This information is presented in a pie chart at the bottom of the page.

A screenshot of a computer

AI-generated content may be incorrect.

### Code Example

This piece of code is used to setup the table look and its ability to be filtered. It shows how the dtaa can be filtered or sorted, and that the rows have the ability to be selected. None of the data is able to be deleted or modified from this application

A screen shot of a computer code

AI-generated content may be incorrect.

This next section of code shows the criteria for each radio button selection and the data it will display. Each are unique to what Grazioso Salvare determined as ideal for each category.

A computer screen shot of text

AI-generated content may be incorrect.

This is the other major driver of the application. This is the area that creates the pie chart and determines what to display based on which radio button is selected.

A screenshot of a computer

AI-generated content may be incorrect.

### Tests

To test the application, it was run to ensure that it functioned correctly. Each radio button selection was tested to ensure that no errors occurred, and the correct data was displayed.

### Screenshots

A close-up of a computer screen

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer screen

AI-generated content may be incorrect.

A computer screen with a white background

AI-generated content may be incorrect.

## Contact

Brandon Hollada

Email: brandon.hollada@snhu.edu