# CS 340 README

## Grazioso Salvare Dog Search and Rescue Training Software

This project is using Austin Animal Center (AAC) dataset for the project using MongoDB and Python to demonstrate Database operations like Create, Read, Update and Delete operations.

## Motivation

This project is to learn how to implement database operations in MongoDB, like, Create, Read, Update and Delete operations.

## Getting Started

In the unix terminal,

1. I changed the directory to /usr/local/datasets.
2. Then using the “mongoimport” imported the AAC dataset “aac\_shelter\_outcomes.csv”
3. Created index for “breed”
4. Created compound index “breed” + “outcome”
5. Created “aacuser” and provided “readwrite” access to “AAC” database.
6. In Python implemented “aac\_crud.py” module
7. In juptyer using imported aac\_crud and “AnimalShelterCRUD” class
8. Checked the “create and “read” methods.

## Installation

To run this project we need:

1. MongoDB
2. Python 3
3. mongoimport
4. mongosh
5. Jupyter

CRUD Python Module:

* create(data) – This module inserts the data into the collection and returns True if successful and False if failed.
* read(query) – This module retrieves all the documents which matches the query.
* update(query, new\_data) – this module first find the document which matches the query and updates with new data.
* delete(query) – this document finds the documents which matches query and deletes all the records and return the number of records deleted or 0 if not record found.

These CRUD operatios are tested in Jupyter by importing the aac\_crud.py module and please see the “Usage” and screenshots below.

## Usage

Below example is executed in Jupyter.

### Code Example

*from aac\_crud import AnimalShelterCRUD*

*#MongoDB connection Details*

*USER = 'aacuser'*

*PASS = 'SNHU1234'*

*HOST = 'nv-desktop-services.apporto.com'*

*PORT = 31416*

*DB = 'AAC'*

*COL = 'animals'*

*#Instantiate CRUD class*

animal\_curd = AnimalShelterCRUD(USER,PASS,HOST,PORT,DB,COL)

#Test data to Insert

my\_animal\_data = {

"rec\_num": 2,

"age\_upon\_outcome": '1 year',

"animal\_id": 'A725717',

"animal\_type": 'Buffalo',

"breed": 'Wild Mix',

"color": 'Silver Tabby',

"date\_of\_birth": '2025-03-25',

"datetime": '2025-03-25 10:12:00',

"monthyear": '2025-03-25T10:12:00',

"name": '',

"outcome\_subtype": 'SCRP',

"outcome\_type": 'Transfer',

"sex\_upon\_outcome": 'Spayed Female',

"location\_lat": 30.6525984560228,

"localtion\_long": -97.7419963476444,

"age\_upon\_outcome\_in\_weeks": 52.92152777777778

}

#calling Create

animal\_curd.create(my\_animal\_data)

#calling Read

query = {"animal\_type": "Buffalo"}

results = animal\_curd.read(query)

print(f"Read operation for {query} and results:\n{results}\n\n")

### Tests

Read operation for {'animal\_type': 'Buffalo'} and

Results:

[{'\_id': ObjectId('67ef504cc3848d887ef5440d'), 'rec\_num': 2, 'age\_upon\_outcome': '1 year', 'animal\_id': 'A725717', 'animal\_type': 'Buffalo', 'breed': 'Wild Mix', 'color': 'Silver Tabby', 'date\_of\_birth': '2025-03-25', 'datetime': '2025-03-25 10:12:00', 'monthyear': '2025-03-25T10:12:00', 'name': '', 'outcome\_subtype': 'SCRP', 'outcome\_type': 'Transfer', 'sex\_upon\_outcome': 'Spayed Female', 'location\_lat': 30.6525984560228, 'localtion\_long': -97.7419963476444, 'age\_upon\_outcome\_in\_weeks': 52.92152777777778}, {'\_id': ObjectId('67ef5b1857f19a548b50d8fd'), 'rec\_num': 2, 'age\_upon\_outcome': '1 year', 'animal\_id': 'A725717', 'animal\_type': 'Buffalo', 'breed': 'Wild Mix', 'color': 'Silver Tabby', 'date\_of\_birth': '2025-03-25', 'datetime': '2025-03-25 10:12:00', 'monthyear': '2025-03-25T10:12:00', 'name': '', 'outcome\_subtype': 'SCRP', 'outcome\_type': 'Transfer', 'sex\_upon\_outcome': 'Spayed Female', 'location\_lat': 30.6525984560228, 'localtion\_long': -97.7419963476444, 'age\_upon\_outcome\_in\_weeks': 52.92152777777778}, {'\_id': ObjectId('67ef5b6257f19a548b50d8ff'), 'rec\_num': 2, 'age\_upon\_outcome': '1 year', 'animal\_id': 'A725717', 'animal\_type': 'Buffalo', 'breed': 'Wild Mix', 'color': 'Silver Tabby', 'date\_of\_birth': '2025-03-25', 'datetime': '2025-03-25 10:12:00', 'monthyear': '2025-03-25T10:12:00', 'name': '', 'outcome\_subtype': 'SCRP', 'outcome\_type': 'Transfer', 'sex\_upon\_outcome': 'Spayed Female', 'location\_lat': 30.6525984560228, 'localtion\_long': -97.7419963476444, 'age\_upon\_outcome\_in\_weeks': 52.92152777777778}, {'\_id': ObjectId('67ef5b8857f19a548b50d901'), 'rec\_num': 2, 'age\_upon\_outcome': '1 year', 'animal\_id': 'A725717', 'animal\_type': 'Buffalo', 'breed': 'Wild Mix', 'color': 'Silver Tabby', 'date\_of\_birth': '2025-03-25', 'datetime': '2025-03-25 10:12:00', 'monthyear': '2025-03-25T10:12:00', 'name': '', 'outcome\_subtype': 'SCRP', 'outcome\_type': 'Transfer', 'sex\_upon\_outcome': 'Spayed Female', 'location\_lat': 30.6525984560228, 'localtion\_long': -97.7419963476444, 'age\_upon\_outcome\_in\_weeks': 52.92152777777778}, {'\_id': ObjectId('67ef5bbc57f19a548b50d903'), 'rec\_num': 2, 'age\_upon\_outcome': '1 year', 'animal\_id': 'A725717', 'animal\_type': 'Buffalo', 'breed': 'Wild Mix', 'color': 'Silver Tabby', 'date\_of\_birth': '2025-03-25', 'datetime': '2025-03-25 10:12:00', 'monthyear': '2025-03-25T10:12:00', 'name': '', 'outcome\_subtype': 'SCRP', 'outcome\_type': 'Transfer', 'sex\_upon\_outcome': 'Spayed Female', 'location\_lat': 30.6525984560228, 'localtion\_long': -97.7419963476444, 'age\_upon\_outcome\_in\_weeks': 52.92152777777778}, {'\_id': ObjectId('67ef5beb57f19a548b50d905'), 'rec\_num': 2, 'age\_upon\_outcome': '1 year', 'animal\_id': 'A725717', 'animal\_type': 'Buffalo', 'breed': 'Wild Mix', 'color': 'Silver Tabby', 'date\_of\_birth': '2025-03-25', 'datetime': '2025-03-25 10:12:00', 'monthyear': '2025-03-25T10:12:00', 'name': '', 'outcome\_subtype': 'SCRP', 'outcome\_type': 'Transfer', 'sex\_upon\_outcome': 'Spayed Female', 'location\_lat': 30.6525984560228, 'localtion\_long': -97.7419963476444, 'age\_upon\_outcome\_in\_weeks': 52.92152777777778}, {'\_id': ObjectId('67ef5bfe57f19a548b50d907'), 'rec\_num': 2, 'age\_upon\_outcome': '1 year', 'animal\_id': 'A725717', 'animal\_type': 'Buffalo', 'breed': 'Wild Mix', 'color': 'Silver Tabby', 'date\_of\_birth': '2025-03-25', 'datetime': '2025-03-25 10:12:00', 'monthyear': '2025-03-25T10:12:00', 'name': '', 'outcome\_subtype': 'SCRP', 'outcome\_type': 'Transfer', 'sex\_upon\_outcome': 'Spayed Female', 'location\_lat': 30.6525984560228, 'localtion\_long': -97.7419963476444, 'age\_upon\_outcome\_in\_weeks': 52.92152777777778}, {'\_id': ObjectId('67ef5c4e57f19a548b50d909'), 'rec\_num': 2, 'age\_upon\_outcome': '1 year', 'animal\_id': 'A725717', 'animal\_type': 'Buffalo', 'breed': 'Wild Mix', 'color': 'Silver Tabby', 'date\_of\_birth': '2025-03-25', 'datetime': '2025-03-25 10:12:00', 'monthyear': '2025-03-25T10:12:00', 'name': '', 'outcome\_subtype': 'SCRP', 'outcome\_type': 'Transfer', 'sex\_upon\_outcome': 'Spayed Female', 'location\_lat': 30.6525984560228, 'localtion\_long': -97.7419963476444, 'age\_upon\_outcome\_in\_weeks': 52.92152777777778}]

new\_data = {"age\_upon\_outcome": "5 years"}

animal\_crud.update(my\_animal\_data, new\_data)

animal\_crud.delete(my\_animal\_data)

### Screenshots

CSV file – “mongoimport” command and command output

A screenshot of a computer

AI-generated content may be incorrect.

Create new user “aacuser” for database AAC.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

Create and Read

*A screenshot of a computer

AI-generated content may be incorrect.*

Read, Update and read screen shot

A screenshot of a computer

AI-generated content may be incorrect.

Reading with new “5 years” input

A screenshot of a computer

AI-generated content may be incorrect.

“delete” command

A screenshot of a computer

AI-generated content may be incorrect.

## Final Project:

Copy the ProjectTwoDashboard.ipynb, aac\_crud.py, Grazioso\_Salvare\_Logo.png to a folder where Jupyter notebook is installed and Run “ProjectTwoDashboard.ipynb”.

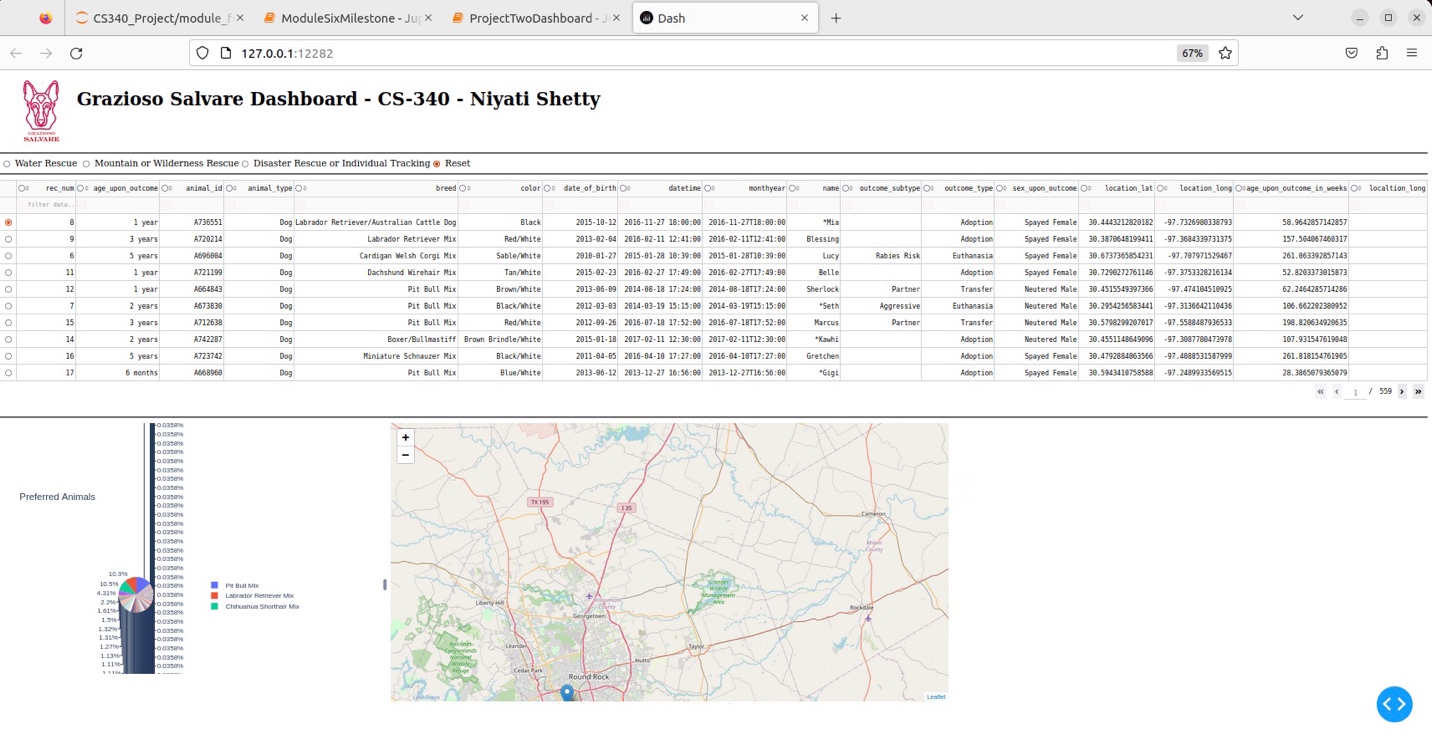
This will launch the web browser, You will see “Grazioso Salvare Dashboard – CS-340 – Niyati Shetty” with logo on the left side.

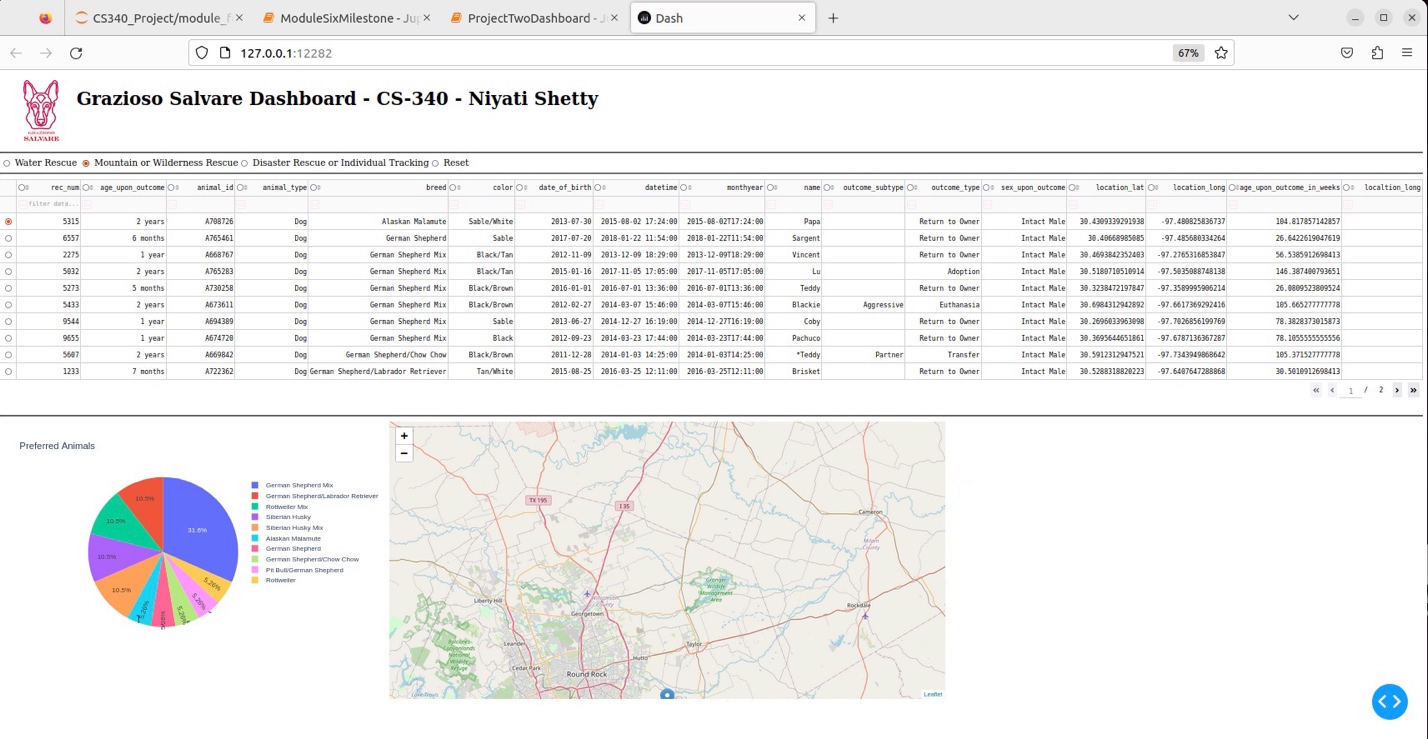
You can interact with the dashboard by filtering the options, like

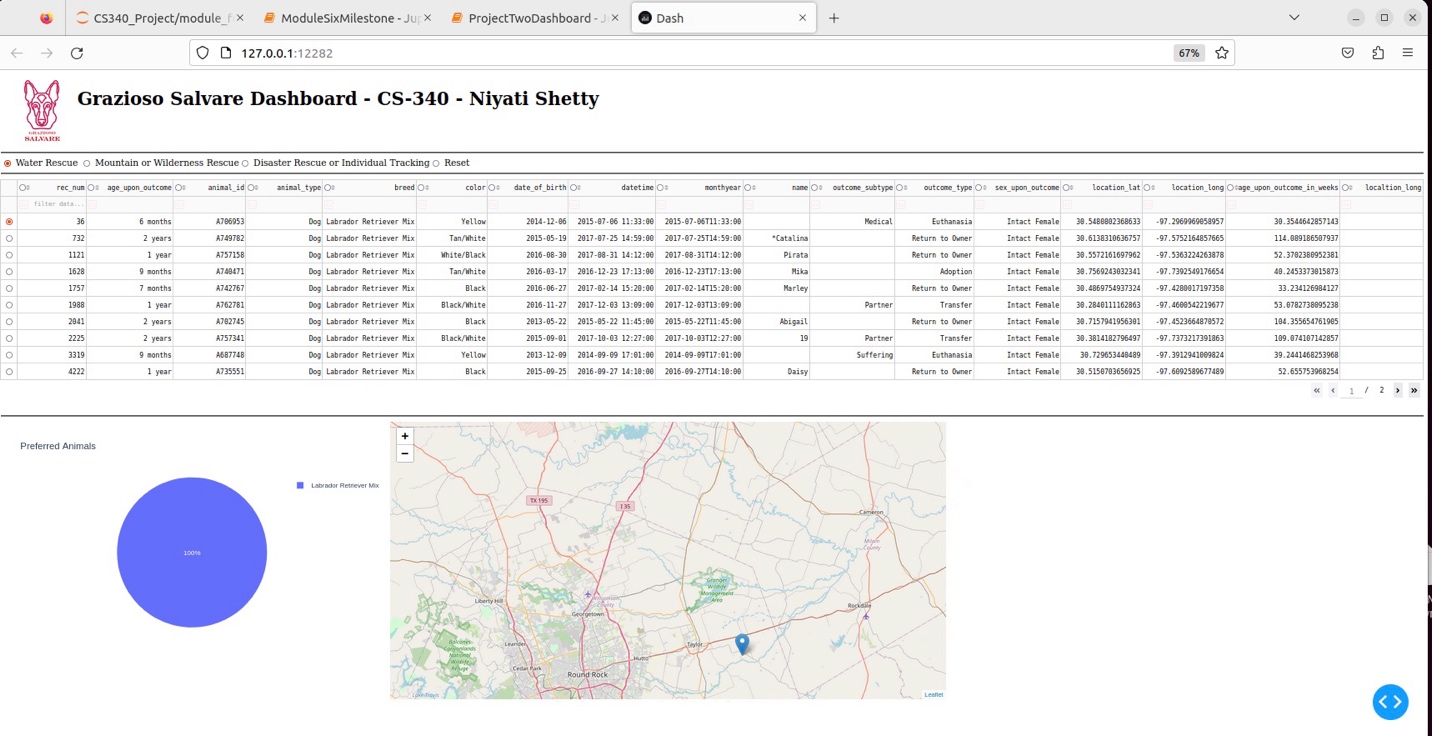
“Water Rescue”, “Mountain or Wilderness Rescue”, “Disaster Rescue or Individual Tracking” or “Reset”

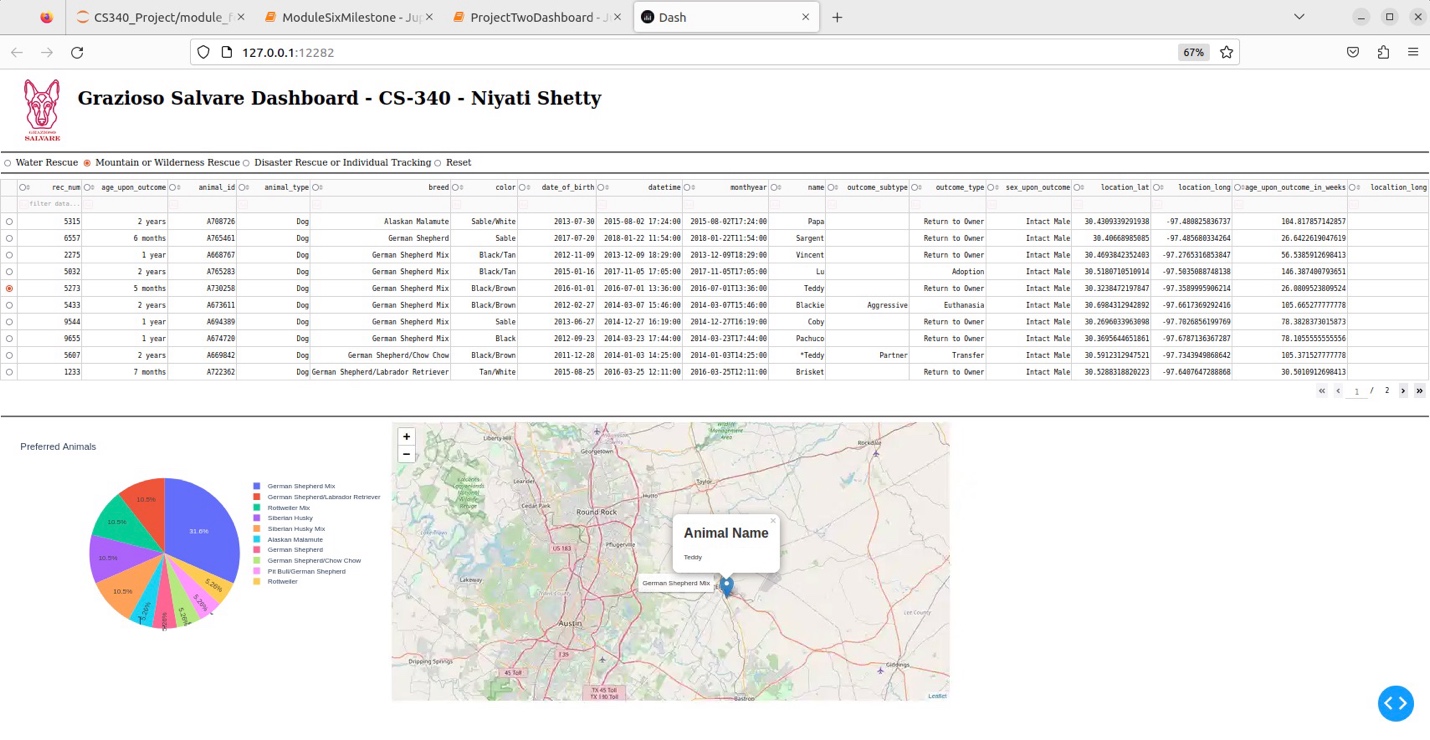
Depending on the filter selected, the table is populated and when you select a row, the location is displayed on bottom right side. On Bottom left side a pie chart with all the types of dogs will be displayed in the filter selected.

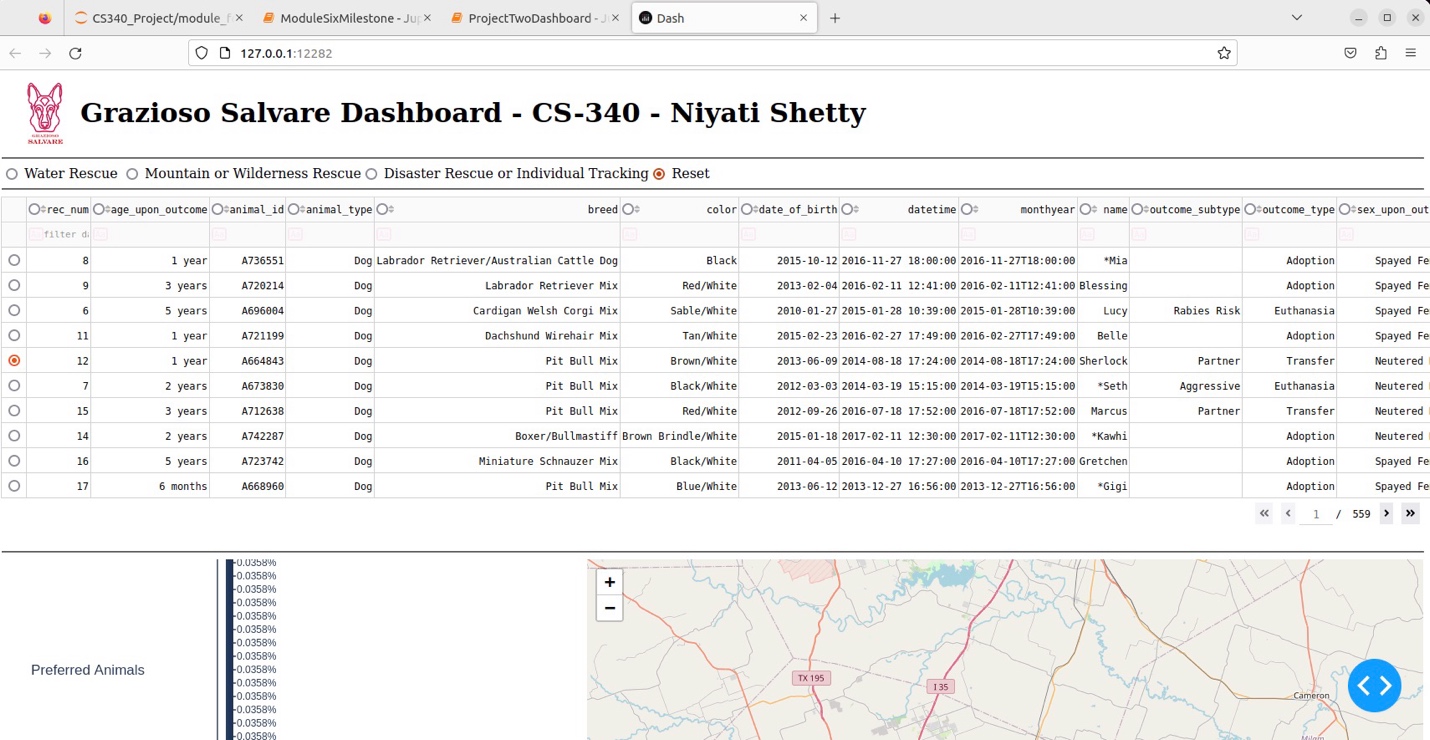
Attaching screen shots of the Dashboard.

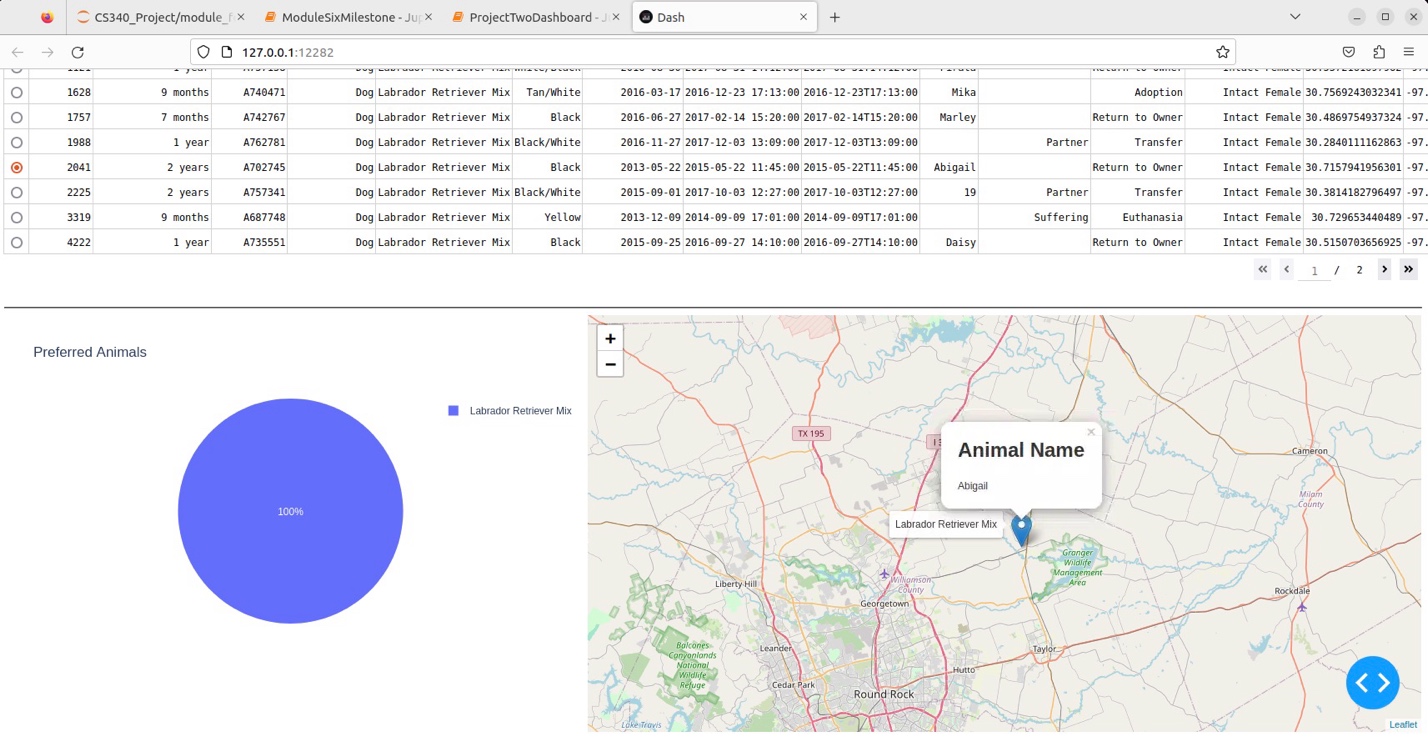


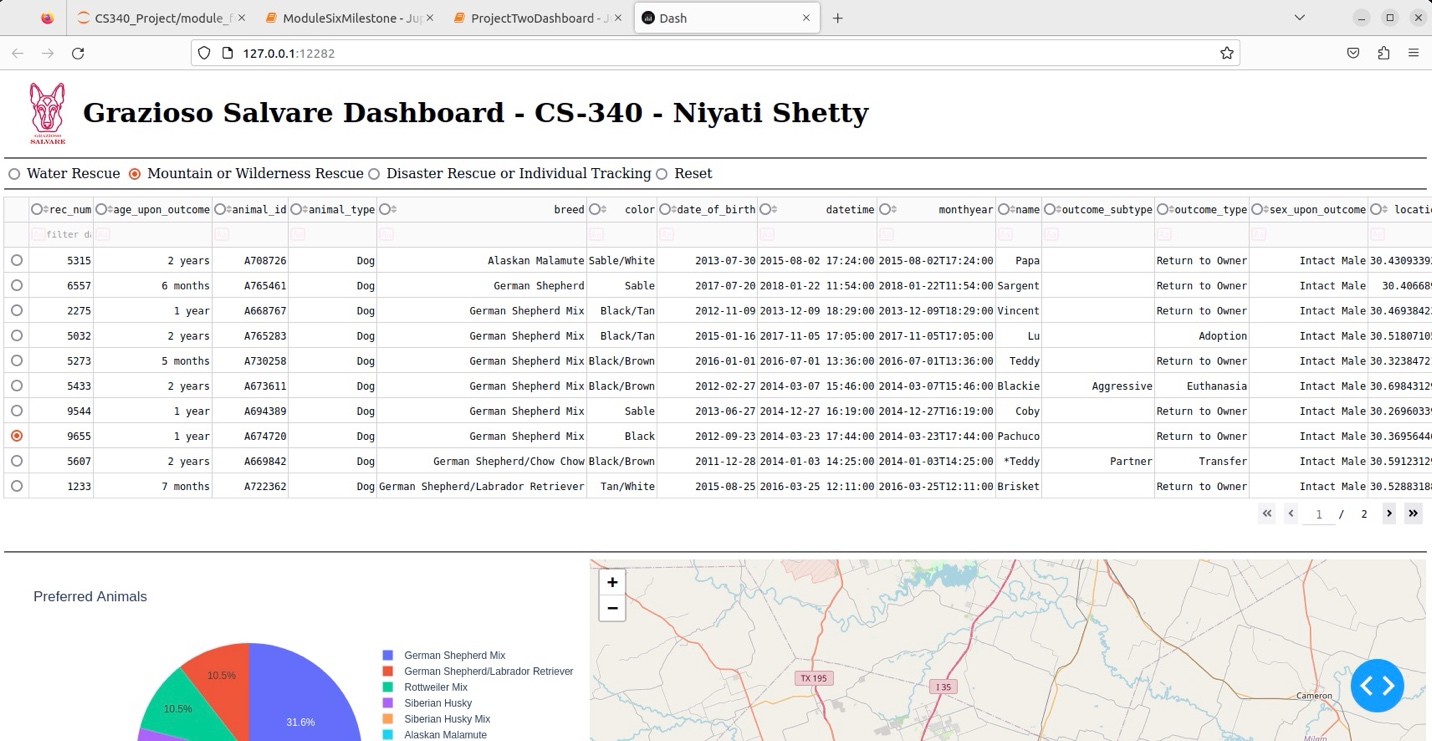


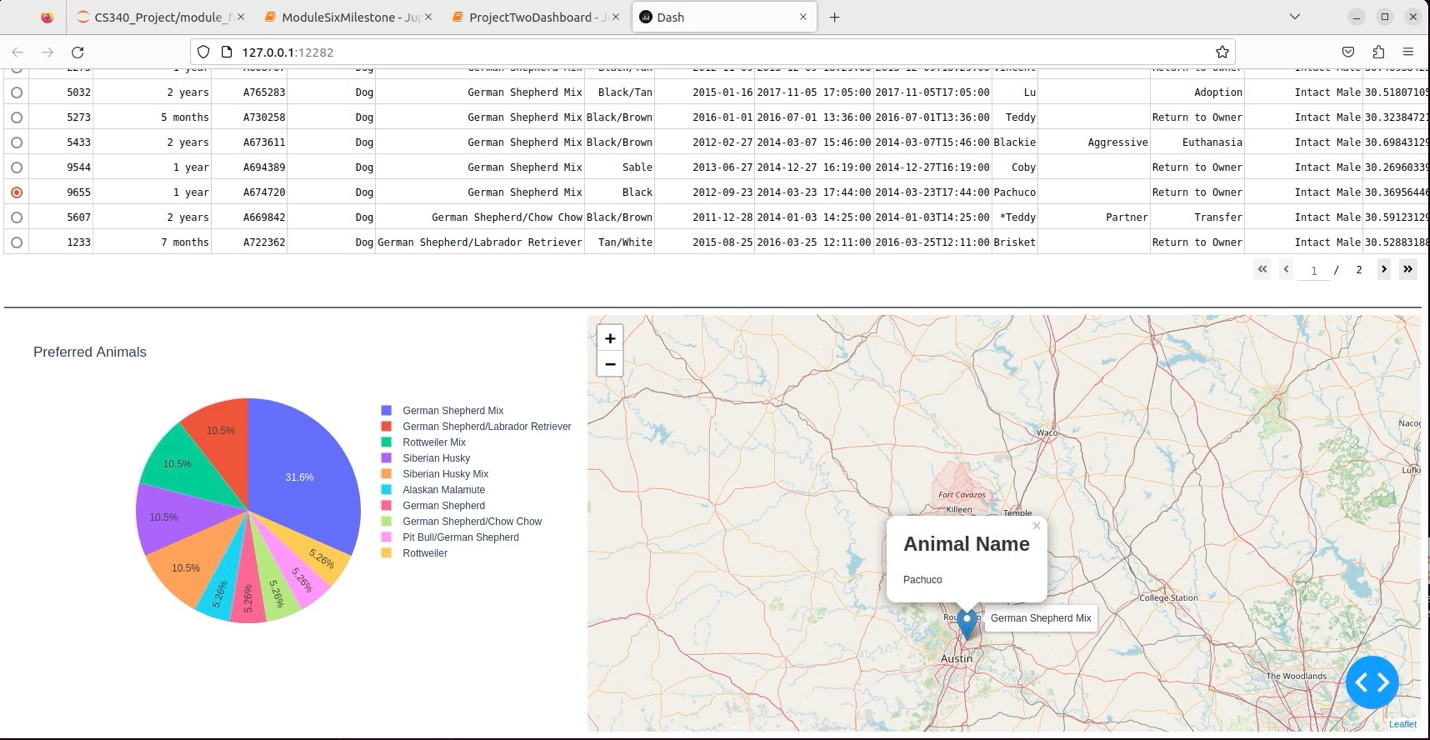












## Roadmap/Features (Optional)

## Contact

Your name: Niyati Shetty