# CS 340 Project Two README

## Getting Started

First, I had to import the information from the Austin Animal Shelter and create an index for it. I then had to create a user account and password that has a specific port to access the AAC database and animals collection for authentication and security purposes. After that I created code that would allow the information from the database to be updated by having the data created, read, updated, or deleted(CRUD). I then created a testing script to make sure that the code functions properly.

In order to create a new user account in the terminal you must open the terminal. Then type ‘cd /usr/local/datasets’ to go into the correct folder. Type ‘mongosh’ so that you can input commands. Type ‘use admin’ to go into the admin account. Then use the command below to add a user and roles:

db.createuser({user: “\_\_\_\_\_\_”, pwd: “\_\_\_\_\_\_\_”, roles: [{role: “readwrite”, db: “AAC”},{role: “read”, db:”AAC”}]})

You can then verify that the account is active by logging into a new terminal and typing

1. Cd /usr/local/datasets
2. MONGO\_USER=aacuser
3. MONGO\_PASS=’*Your password here’*
4. printenv | grep -I mongo
5. mongosh
6. db.runcommand({connectionStatus:1})

## Installation

The tools I used were the command line/terminal, and Jupyter notebook. The libraries that I used were pymongo and Objectid. I used a Linux computer to create the program. You will need a python ide to be able to use this program.

## Usage

### Code Example

*A screen shot of a computer code

Description automatically generated*

*A screenshot of a computer program

Description automatically generated*

*A screen shot of a computer error

Description automatically generated*

### Tests

*Tests the CRUD(Create, Read, Update, and Delete Functionality)*

*A screen shot of a computer program

Description automatically generated*

### Screenshots

THESE SCREENSHOTS SHOW MY WORK IN THE TERMINAL

Importing the documentsA screenshot of a computer screen

Description automatically generated

Create Index

A screenshot of a computer

Description automatically generated

Create Compound Index and use Example Query

A screenshot of a computer screen

Description automatically generated

Created new user

A screenshot of a computer screen

Description automatically generated

Logged in as new user and verified

A screenshot of a computer screen

Description automatically generated



* ***Describe the required functionality****of the project. Include the screenshots or screencast taken while testing and deploying your dashboard (Step 6) as proof that you have achieved the required functionality.*

Grazioso Salvare is seeking a software application that can work with existing data from the animal shelters to identify and categorize available dogs. Global Rain has contracted for a full stack development of this application that will include a database and a client-facing web application dashboard. Grazioso Salvare will use this dashboard to interact with and visualize data from a MongoDB database. The dashboard must be a user-friendly, intuitive interface that will reduce user errors and training time. The Dashboard must also include a geolocation chart and another chart of your choice. I chose a column chart as my second chart.

PROOF OF LOGO RUNNING:

A red line drawing of a dog

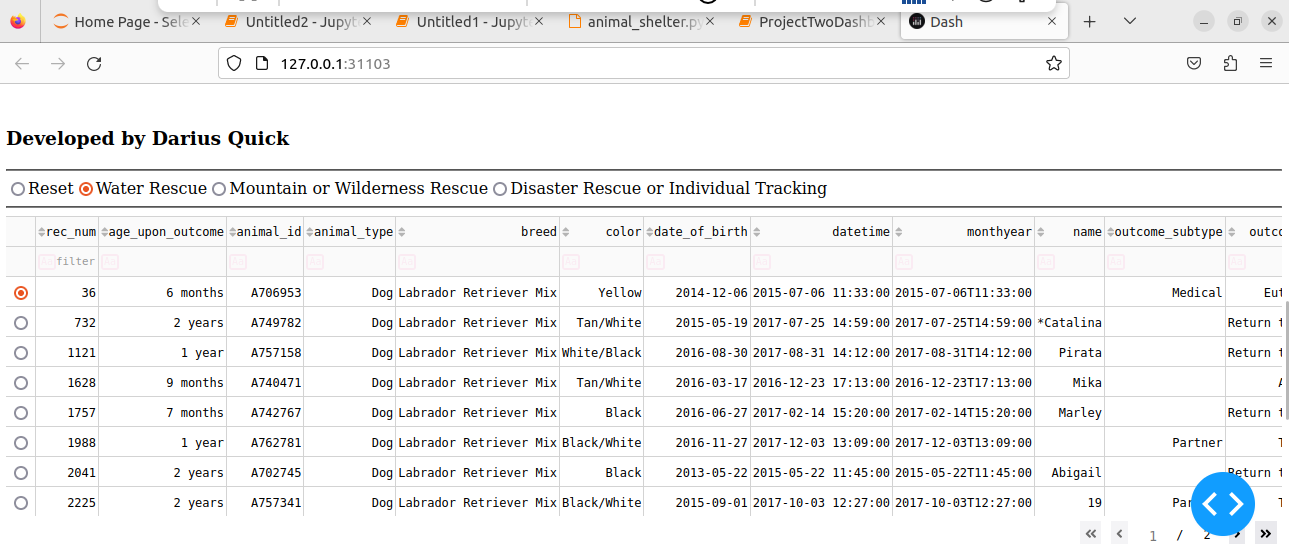
Description automatically generated

PROOF OF DASHBOARD RUNNING CORRECTLY UNDER EACH FILTER:

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

PROOF OF BOTH CHARTS UNDER EACH FILTER:

A screenshot of a map

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a map

Description automatically generatedA screenshot of a map

Description automatically generated

* ***Describe the tools used to achieve this functionality and a rationale for why these tools were used.***

In order to complete this assignment, I used the Linux Terminal with MongoDB as the database system, Jupyter Notebook to create and run the code for the dashboard, and dash framework in order to create the layout, structures, and images necessary for the geolocation and column chart.

MongoDB was used as the model component of the development because it interfaces well as a database system with dashboards and the python programming language.

The Dash Framework was used to create a layout where a logo, unique identifier, dashboard title, database/dataframe information, 2 charts, and unique features could be used.

* ***Explain the steps that were taken to complete the project.***

After the database was created in Project One (See other project and ReadMe File) we had to start working on the Dashboard by properly importing all the necessary Dash Framework and Animal Shelter Database files from my CRUD py file. Then we had to set up the layout for the Dashboard and make sure that the unique features including the logo, dashboard title, unique identifiers, radio options, and unique features/filtering options were set up correctly. I then had to add the necessary information for a Geolocation Chart and Column Chart to be populated next to each other under the dashboard information. I then created the last line of code to run the program. Now all that needs to be done is for the Project File to be run in order for the Dashboard and Charts to be shown with the Animal Shelter information.

* ***Identify any challenges that were encountered and explain how those challenges were overcome.***

I had trouble creating the features and filtering options of the dashboard because I didn’t initially know how I was supposed to add them into my python code. But after extensive study in the dash framework website under layouts I was able to apply what I had learned to add both options.

## Contact

Your name: Darius Quick

**Dash Framework Website:**

<https://dash.plotly.com/>

**How to download Jupyter Notebook:**

<https://jupyter.org/install>

**How to download MongoDB:**

<https://www.mongodb.com/docs/manual/administration/install-on-linux/>