# A dog with a black background Description automatically generated

# Grazioso Salvare

**Service Dog Search Dashboard**

## About the Project:

*Grazioso Salvare’s custom service dog search dashboard allows users to search for dogs quickly and easily with the specific filters needed for their three service dog types: water rescue, mountain or wilderness rescue, and disaster or Individual tracking. With the implementation of the histogram, users also have a visual aid that will show how many dogs of each breed are on the current page of the data table. Also, a geolocation widget updates a dog’s location, breed, and name when a specific row is selected. Finally, using MongoDB for creating, reading, updating, and deleting (CRUD) information, gives the user a seamless framework to keep their data up to date. This project consists of a database, an API, and a dashboard.*

## Motivation for this Project:

*The motivation for this project is to simplify the organization and management of this database for the user. Having this dashboard in place allows the users to search for specific dogs using filters, saving them hours of time and effort.*

**MongoDB Usage Motivation:**

*Mongo was specifically used for this program because of its Python-friendly interface and ability to quickly setup the database from a CSV document. Although Python can use SQL and other database tools, switching between them can be difficult due to the differences in syntax. When performing the fundamental CRUD operations of a database using Python in a Mongo DB, the selection tools are substantially simpler than in a SQL database.*

**DASH Usage Motivation:**

*Dash's dynamic nature made it an ideal tool for creating the dashboard. Dash is a JavaScript tool that is based on React and offers an exceptionally responsive framework. HTML Dash tags are used in Dash to regulate segment outputs. The Python module's programmed instructions are then used to update any of the target inputs that were specified during the app callbacks process.*

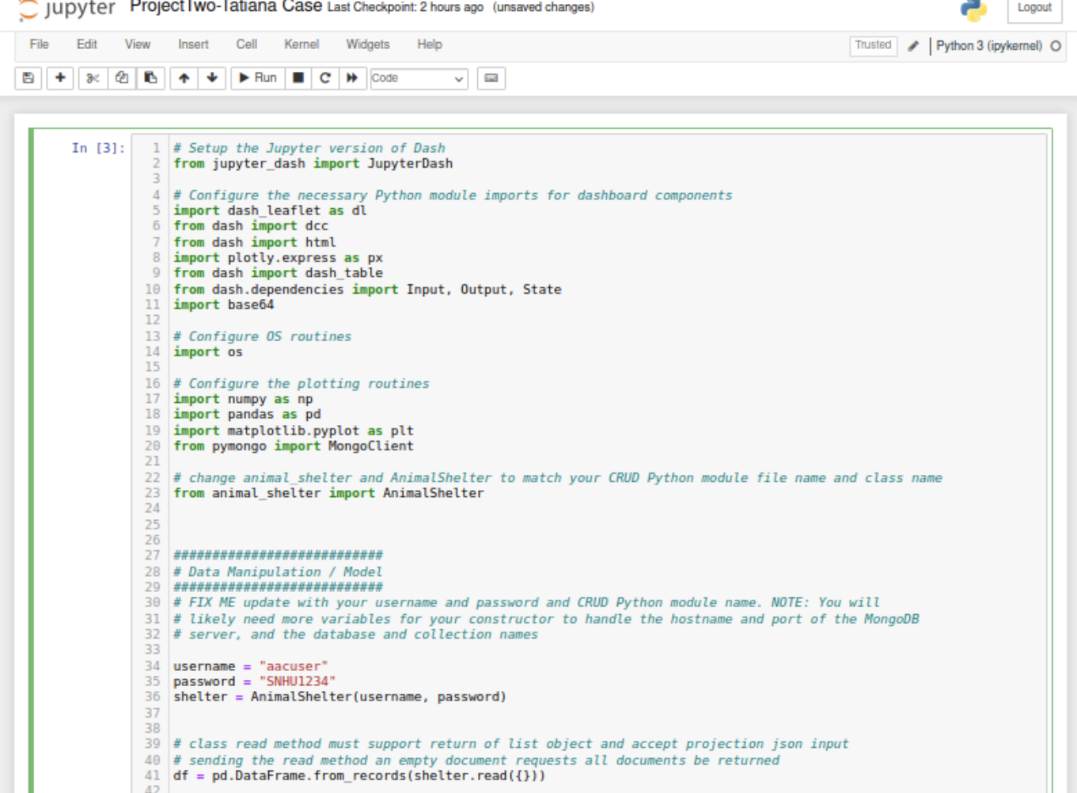
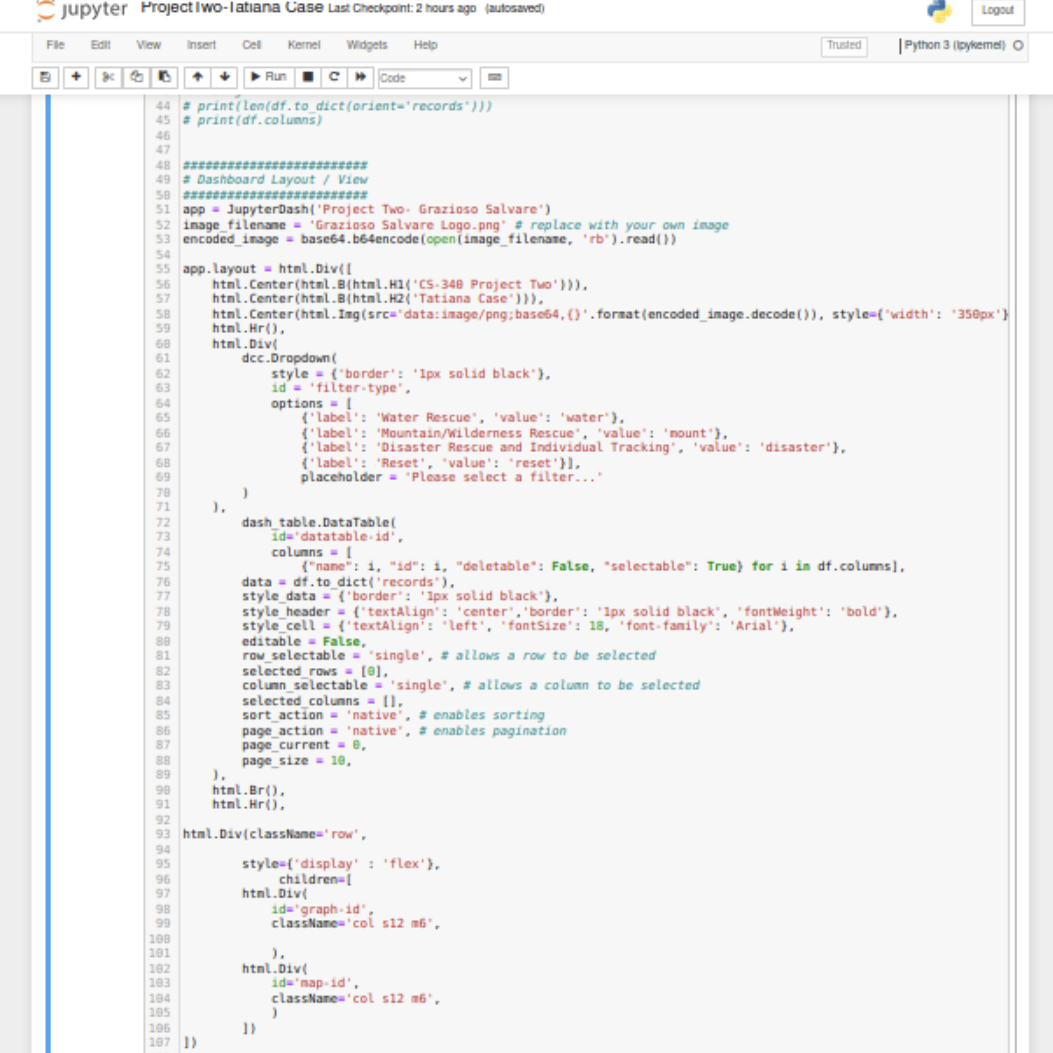
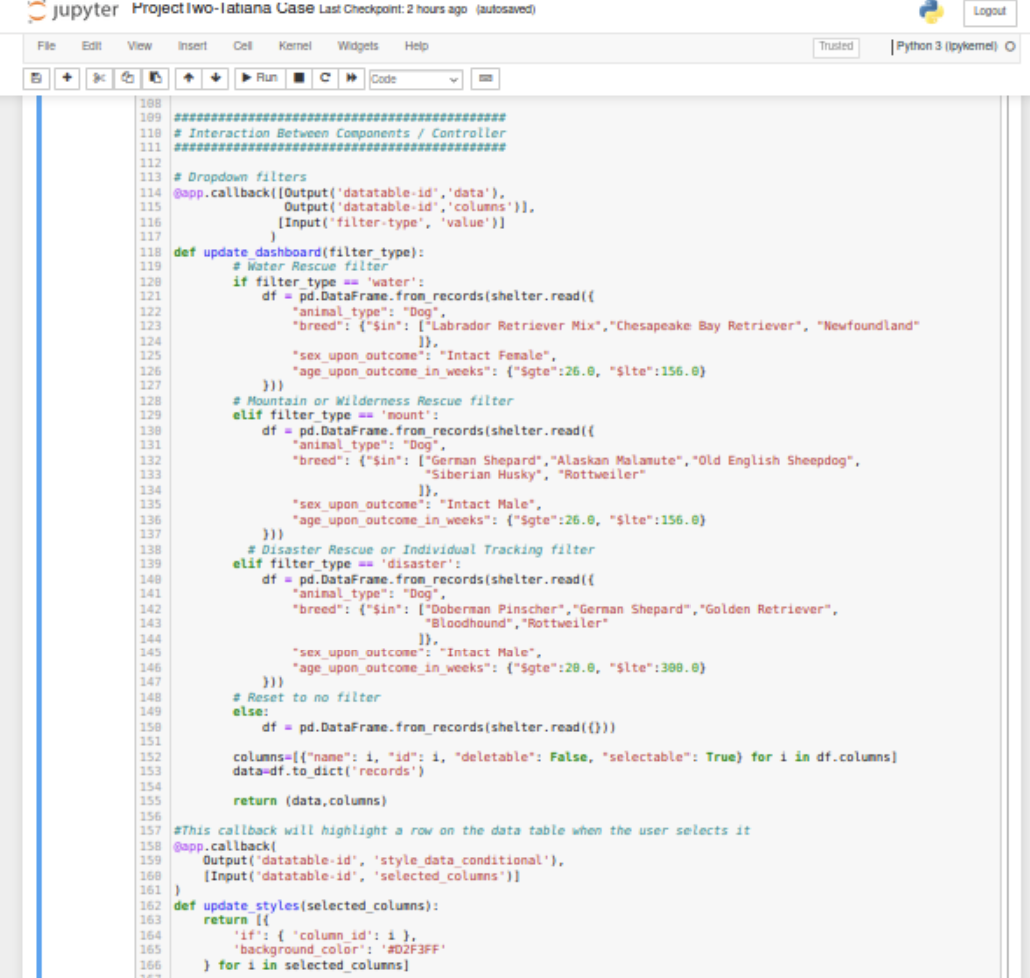
## Getting Started

*To get a local copy up and running, follow these simple example steps:*

1. *Enter MongoDB and upload the existing CSV file that contains the Animal Shelter Database:*
   1. *A screenshot of a computer screen

      Description automatically generated*
2. *Create a user for the AAC database that has readWrite permissions:*
   1. *A screenshot of a computer program

      Description automatically generated*
3. *Download the animal\_shelter.py file and the accompanying testing file and open using Jupyter Notebook.*
4. *Using the testing file, import the animal\_shelter file and AnimalShelter class and begin creating, reading, updating, and deleting documents:*
   1. *A screenshot of a computer

      Description automatically generated*
5. *After creating and testing the AnimalShelter class, we can begin creating the Dash web application. Begin my importing the necessary files, validating the username and password.*
   1. **
6. *Begin creating the app by adding the Grazioso Salvare Logo and using CSS and HTML to customize as desired. Add a dropdown menu that includes the necessary filters. Also, create a space on the application that will hold the histogram and the geolocation map.:*
   1. **
7. *Next, create the interaction between the filters, the dropdown menu, and the data. Also, create the interaction between column selection and the color of the column.:*
   1. **
8. *Finally, Include the code for the histogram and the geolocation map that will update with the filters and the animals actively displayed on the data table.:*
   1. *A screenshot of a computer

      Description automatically generated*

## Installation

*The tools you will need to use this program are as follows:*

1. *MongoDB – Download from their website:*
   1. [*https://www.mongodb.com/try/download/community*](https://www.mongodb.com/try/download/community)
2. *Jupyter Notebook – Use on their website:*
   1. [*https://jupyter.org/try-jupyter/lab/*](https://jupyter.org/try-jupyter/lab/) *or download the software on your device* [*https://jupyter.org/install*](https://jupyter.org/install)
3. *Plotly – This can be imported directly into the Python module using:*
   1. *Import plotly.express as px*
4. *Dash – Dash’s components can be imported into Jupyter notebook from their website:*
   1. [*https://pypi.org/project/dash/*](https://pypi.org/project/dash/)
5. *Pandas – Download from their website:*
   1. [*https://pandas.pydata.org/pandas-docs/stable/getting\_started/install.html*](https://pandas.pydata.org/pandas-docs/stable/getting_started/install.html)

## Usage

*The animal\_shelter.py file contains the Create, Read, Update, and Delete functions:*

1. *Create Method:*
   1. *Code Example:A screen shot of a computer

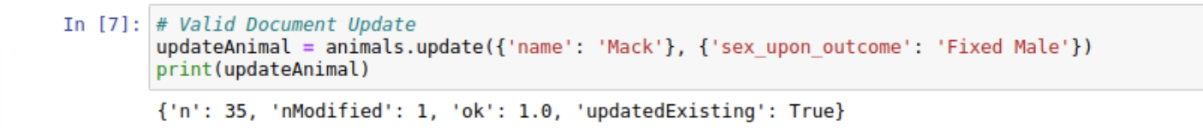
      Description automatically generated*
   2. *Code Implementation and Testing:A screenshot of a computer

      Description automatically generated*
2. *Read Method:*
   1. *Code Example: A screen shot of a computer

      Description automatically generated*
   2. *Code implementation and Testing – Valid Search:A screenshot of a computer

      Description automatically generated*
   3. *Invalid Search:A screenshot of a computer

      Description automatically generated*
3. *Update Method:*
   1. *Code Example: A screen shot of a computer error

      Description automatically generated*
   2. *Code Implementation and Testing – Valid Document Update: *
   3. *Code implementation and Testing – Invalid Document Update: A screenshot of a computer program

      Description automatically generated*
4. *Delete Method:*
   1. *Code Example: A screen shot of a computer code

      Description automatically generated*
   2. *Code Implementation and Testing – Valid Document Deletion: A close up of a computer screen

      Description automatically generated*
   3. *Code Implementation and Testing – Invalid Document Deletion: A computer screen shot of a program

      Description automatically generated*
5. *All code examples detailing the creation of this Dash application can be located in the “Getting Started’ section of this document. To view a screencast of the running application, please click on the Dropbox link below:*
   1. *https://www.dropbox.com/scl/fi/1b6vrku4ev2x1i7vyde6o/Project-Two-Screencast.mp4?rlkey=l0w98m4h9pk0jpv8bh1o0z8ny&st=e53s2owg&dl=0*

## Contact

Your name: Tatiana Case

Email: Tatiana.case@snhu.edu