# CS 340 README Project 2

## About the Project/Project Title

## The purpose of this project was to develop a databased and Python module capable of CRUD functionality for MongoDB. The project required development of a dashboard and database interface logic; the dashboard includes attributes and a user-friendly interface that allows easy navigation with limited learning time. Additionally, the code used in this project is open sourced and will be uploaded on GitHub so that it may be replicated in future, similar projects. This README will serve as an instructional guide for future replication and contain all the sections of a standardized document.

## Motivation

The creation of this MongoDB database project allows the developer to gain comprehensive insight onto various methods used for creating efficient databases as well as setting user and admin access. Furthermore, the animal database created serves as a foundation for developing additional databases or improving the function by improving the methods implemented. Additionally, the database has real-world application for numerous projects that may require the storage of data and documents. While the project is solely based on the AAC and CRUD, these principles can be applied to inventory management systems or company logs of employee history. Therefore, the project not only serves as a great foundational insight into developing a MongoDB database but also a template for designing additional client/server databases in the future.

## Getting Started

The AAC database project contains two files. A Python file titled shelter.py and a Jupyter Notebook file titled app.ipynb. In order to execute the database, both of these files must be present. Once both files have been downloaded, the database can be opened by uploading to Mongo through the Terminal and opening the database.

## Installation

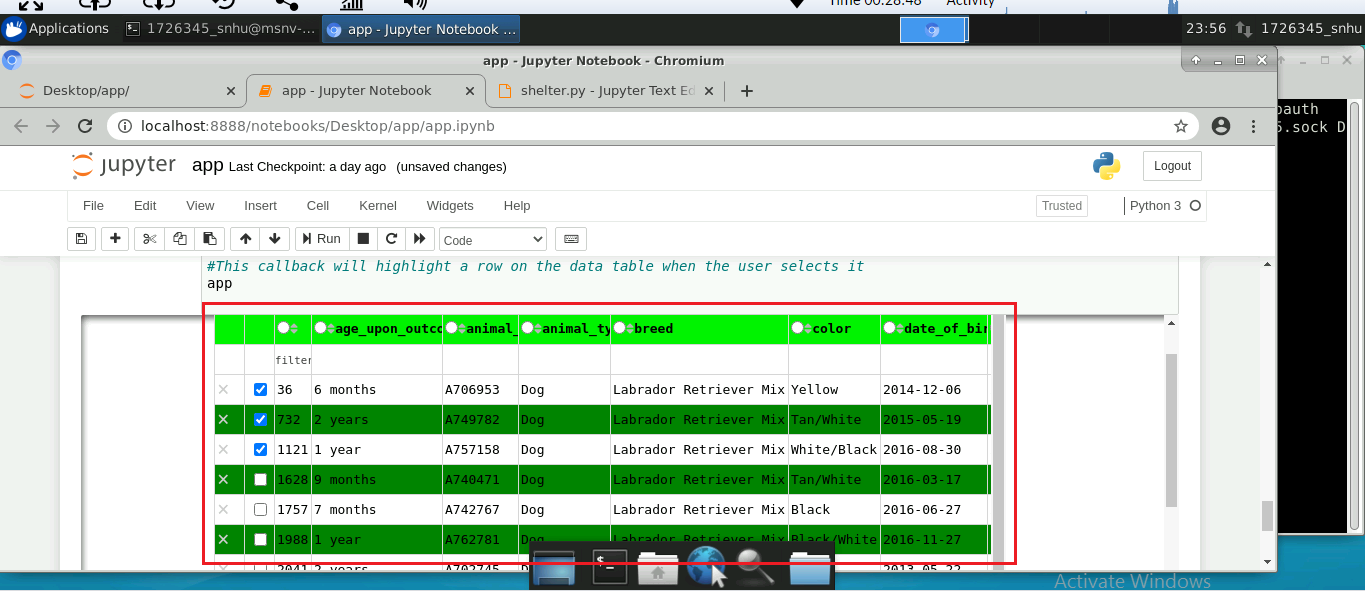
The tools required to use the software are the files mentioned in the “Getting Started” section as well as Python, Jupyter Notebook, and Mongo database. In order to install Python, go to the following link: <https://www.python.org/downloads/>. In order to install Jupyter Notebook, visit: <https://jupyter.org/install>. Once the files and applications have been installed, proceed to the “Usage” section for details on how the project functions in tandem.

## Usage

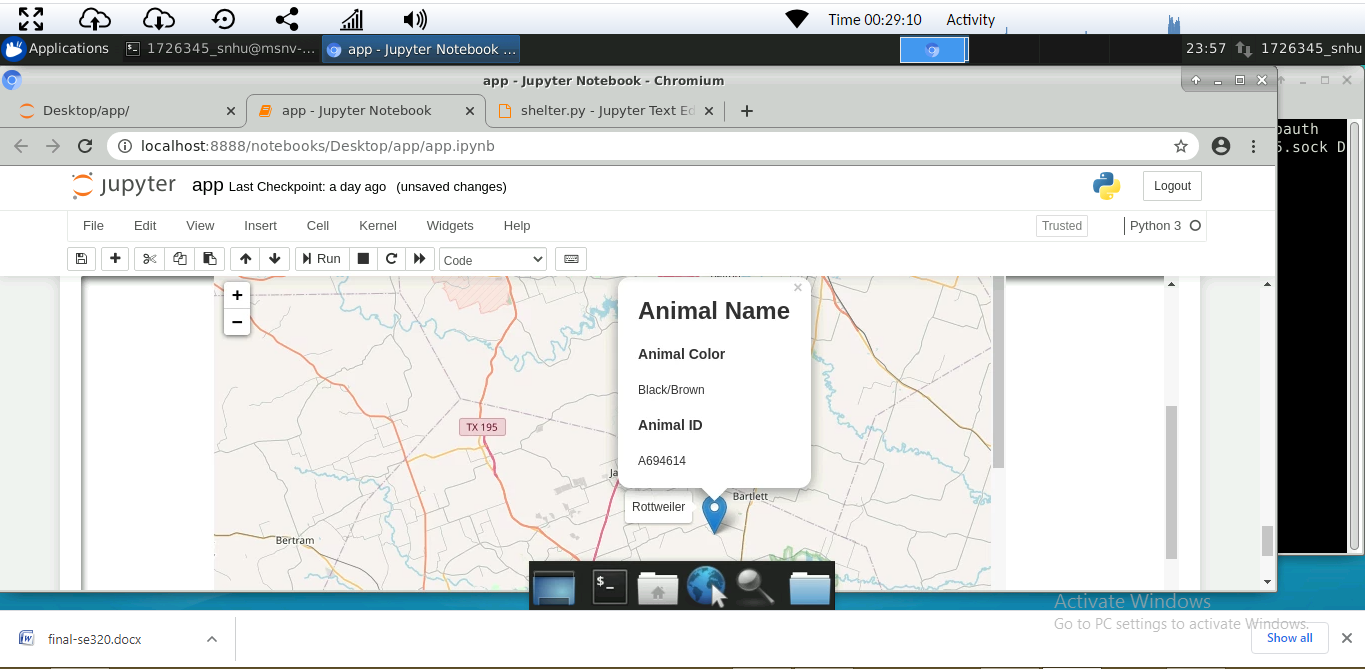
The first image below shows how the database can be searched by filtering data. This is achieved through a simple interface that is user-friendly and displays all required categories of Water Rescue, Mountain, Disaster Rescue, and Reset.

## Screenshot (3756).png

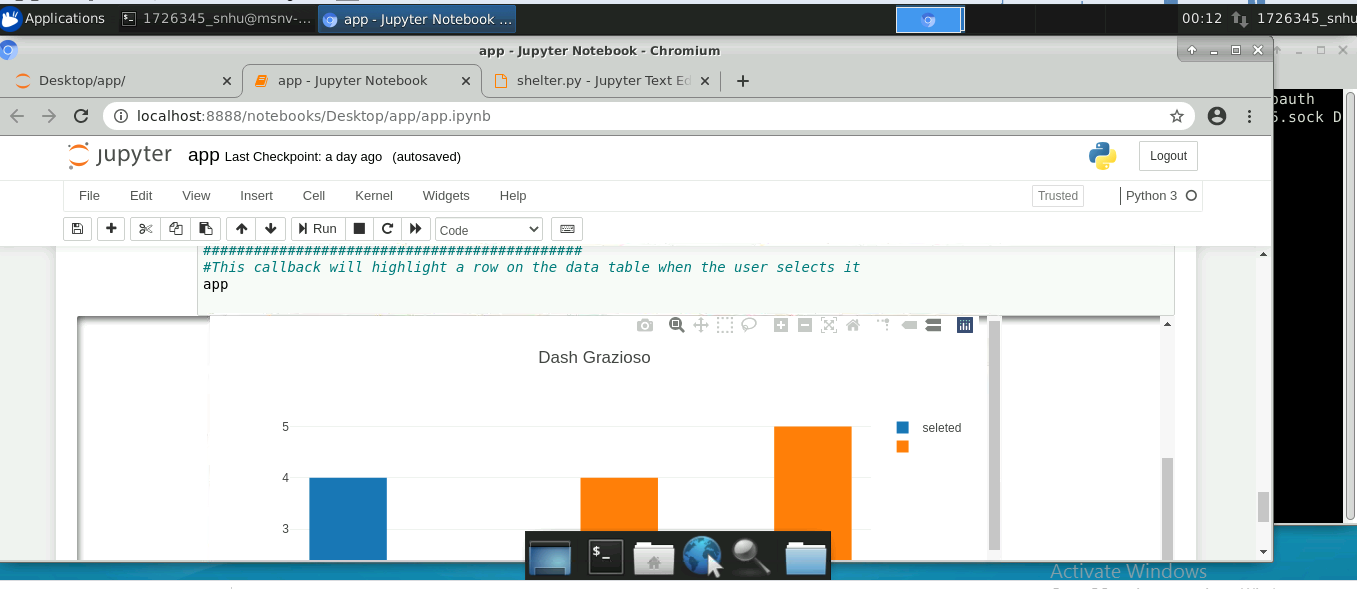
Furthermore, the database is displayed with a detailed interface that allows the user to quickly identify the age, animal ID, animal type, breed, color, and date of birth. The database is separated by colored rows to ensure that the user is capable of identifying all information pertaining to a specific entry.



Additionally, it is important to note the database displays animals that belongs to specifically selected categories, which enables the user to only search for animals that they require information on rather than being required to view the entire databased without search parameters. The database also contains geo charts, which is a web-based map that highlights the location of specific animal IDs and details pertaining to the animal. The map is an effective tool for users who seek locational data and can enable them to make choices based on their relevant distance to a specific animal.



The final image shows how users can gain a visual representation of the data through the form of a bar graph. This feature is useful because it enables the user to quickly analyze their filtered database and gain better insight into the animals being analyzed.



**Dash Library**

It is also important to mention that the dash library incorporated is an efficient tool to use that contains documentation that is integrated with internet capabilities. This enables the user to be presented with an easy to learn interface that provides data representation and visualization. Additionally, the dash library provides more capabilities for highlighting inputs from the user, which can improve interaction with the animal database. The components of the database are formatted using CSS, which not only produces a user-friendly database but also looks engaging while being detailed yet simple to understand.

**Challenges**

While developing the project, the major challenges I encountered were error handling and capturing exceptions. This challenge was overcome by carefully analyze the code and searching for syntax errors and the improper use of characters that caused the code to not execute as intended. Besides these challenges, the project was straightforward and should be easily reproducible by running the attached Python and Jupyter Notebook files.

## Contact

Your name: Jordan Matsumoto