# CS 340 README Template

## About the Project/Project Title

The title of this project is Project Two. The purpose is to create an interactive dashboard that allows the user to visualize the shelter database.

## Motivation

This project was created in order to allow Grazioso Salvare the ability to search for specific queries in the MongoDB database through a simplified, interactive dashboard. This includes a dynamic data table that responds to filtered options, a geolocation chart, and a pie chart.

## Getting Started

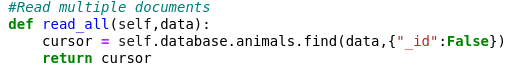
Before beginning the script, the data must be imported into MongoDB. Next, the user has to be authenticated through MongoDB with a username and password that gives the user read and write capabilities of the imported database. I was able to create my Python modules by adding create and read methods that will add and search for data within the database. I also added update and delete functionality that will allow the user to change or remove single or multiple documents in a database. Next, was writing the code to create the dashboard. I linked in the Gravioso photo with a reference to the SNHU website. The first challenge I had was writing in Python as it’s been a while since I’ve used this language. I did have to use resources online for help in creating the modules. The second challenge was the errors I was getting. Every time I corrected one error, there would be another that would pop up. Unfortunately, I wasn’t able to solve all of the errors such as the pie chart not functioning. In order to submit something, I plugged in portions of the code in the Module Six assignment so at least the dashboard would show some of the required elements.

## Installation

The tools needed for this project are the terminal for importing data into MongoDB and setting up user authentication and the Jupyter Notebook for writing and executing the Python script. Dash framework was also used as it works with Python to build web applications. The libraries used were “pymongo” and “objectid”. Python and pymongo were chosen because of the tool set it provides in communicating with the MongoDB server. It makes it very simple to perform functions in Mongo through Python scripts. The Dash framework was chosen to build the application because of how ideal it is for creating visualizations of data in an application.

## Usage

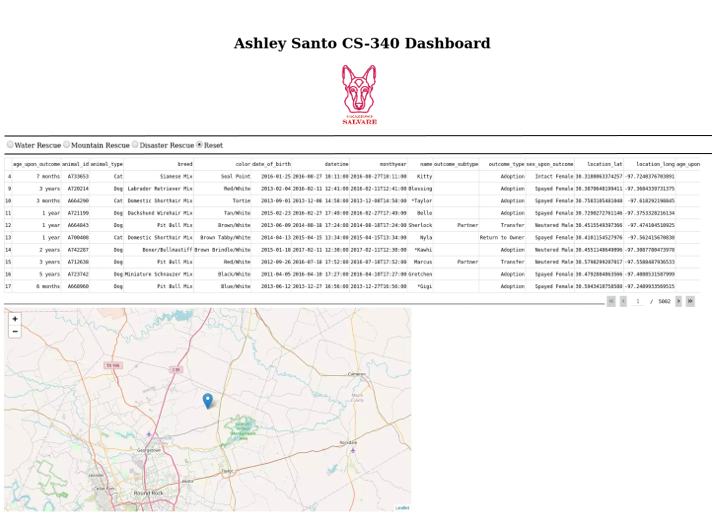
### Code Example

The code below is an example of one of the Python methods used to read data in the database. The method will locate the criteria in the database and load every matching item and return it to the cursor for output in the data table.

### Tests

Tests were done using the Jupyter Notebook where the script was written. For testing, sample user and data were inserted in order to make sure the module was functioning correctly. Then, to make sure it was inserted correctly, I checked the MongoDB database to see the document. For the update function, I used information in existing documents and updated them with data. So, for any “breed” of “Samoyed”, I had it update the “age\_upon\_outcome” to “1.367 years”. Similar to creating documents, I deleted documents containing specified data. These can be seen in the screenshots below. Testing on the geolocation and data table was run to make sure it loaded correctly. Unfortunately, I was unable to test functionality or the pie chart or the filtering options.

### Screenshots



## Contact

Your name: Ashley Santo