**CS 340 Project 2: ReadME Mark Mulder**

Note: Unfortunately, due to a connection issue, I was not able to gain access to screen shots.

The project takes in data from a mongo DB database, specifically the AAC database, and uses it to create an interactive user experience. First, the proper libraries need to be implemented. After that, begin setting up the layout of the screen using a graph, radio button, or any other method of preference. In my case, I used submit-buttons, though in hindsight, a radio button and proper coding could allow for the selection of more than one category. Be sure to display out options as given by the project, being labels for Water, mountain, or disaster rescue along with a reset option and setting up a label sheet to organize them into position. Next, set up the submenu as you see fit, preferably as a dash table, including the categories/options of name, id, deletable, and selectable.

At first, I couldn’t get all the pre-given functions to play nice together, so I ended up combining them under update\_graphs in my prototype file while I researched how to get them to function properly. Its fine to have two different files, one for final turn in and fine tuning with another prototype one for testing. One filled in, these functions will create, fill, and updates graphs and tables as the options are selected.

* The update\_dashboard function will bring up the items under the initial selected choice and bring them onscreen.
* update\_map adds a geolocation module
* update\_graphs will update the graphs for each choice used

The steps used will largely be research then implementation for each step. I began by testing out different type of buttons in a proto document. I ended up settling on the submit-buttons because I didn’t like mixing the button types I was using, and that I wasn’t comfortable using a radio button as a reset. Once that was functional, I moved on to update dashboard which would change the displayed text into an interactive table that was under the category of the button selected, all the information being drawn from the mongoDB database. To do this, just have the code re-write what is has written in the table format described earlier int the layout, with information dredged up from the database under defined categories and types. When doing all this, be sure to use the functions from the CRUD module to read the items properly. Lastly, use plotly imports to create and update a graph while feeding it data for it to use from the database. The main reason for the usage of these tools is that they are not only recommended in online tutorials and manuals, but also some are required for this project to work or be graded properly.

Many issues I had were a lack of knowledge on how to conduct most of these actions. Using the jupyter tutorials, manuals, articles, references, suggestions, and plenty of time on youtube was able to give me a good idea of what to do. My main issue in the second half of testing was the inexplicable loss of the code to use its starter information. That I was unable to find a fix for.

**Sources:**

Handful of the material resources for this project: Did not start reocording until after coding was finished and work on the readME began: Note for future self, take note of resources and read instructions of all parts, including the ReadME

<https://www.youtube.com/watch?v=C0fzIsZA2r8>

<https://www.youtube.com/watch?v=rE_bJl2GAY8>

<https://medium.com/@technologger/how-to-interact-with-jupyter-33a98686f24e>

<https://www.youtube.com/watch?v=nuQD3Xfr0KY>

<https://plotly.com/python/ipython-notebook-tutorial/>

Note: The jupyter link has other pages not listed that are also extremely helpful to this project