Module 7 ReadMe Assignment

William Neal

SNHU

CS340: Client/Server Development

Professor Kellogg

10/22/2023

A computer screen with a message box

Description automatically generated

A computer screen with a red circle

Description automatically generated

Unfortunately Jupyter failed to load the JSON file included in the Rubric and Guidelines, I attempted to download the file again make the FIX ME changes and upload that, only to get a similar error. Jupyter/Apporto has been giving nothing but grief since day one. I did make the changes to the ipynb, which I can attach in the zipfile. I also took screenshots A screenshot of a computer

Description automatically generated

I will also include those in the zipfile just in case there are any issues.

As for a mock write up of a README file for this project. I believe the Dashboard was intended to build out an interaction space for the animal shelter data we had been working with using python/dash. Going over the rubric we can see the required functionality would be An interaction Filtering option, a friendly UI dashboard layout, a customizable table, a geolocation chart and optimizaed data retrieval from our CRUD modules. Now if jupyter/apporto were functional I would display my screen shots like in step 6 but unfortunately that is not that case. I can go on to talk about the tools used to build this program. Python is a really powerful scripting language with some object-oriented attributes. Dash is a python web framework that is great for building web-based data dashboards. Dash has awesome customization features ( <https://dash.plotly.com/plugins> and <https://medium.com/codex/creating-custom-plotly-dash-components-933b405abd21> ). Dash without being said can produce data tables, checkboxes and dropdowns for filtering and giving the client/user ease of access while moving through the data. Dash is also used by a multitude of individuals so there is a solid community for resources and open documentation ( <https://dash.plotly.com/minimal-app> ). As I mentioned before Dash has an infinity for python, and python has great connectivity to MongoDB. Now MongoDB is a NoSQL DB that we have been using since day one of our architecture for the project. Like I was saying MongoDB has great flexibility with python, such as the driver PyMongo. One of the best core components of MongoDB is its ability to work with JSON documents, more specifically its BSON documents. (<https://www.mongodb.com/docs/drivers/java/sync/current/fundamentals/data-formats/document-data-format-bson/#:~:text=BSON%2C%20or%20Binary%20JSON%2C%20is,%2C%20ObjectIds%2C%20and%20binary%20data>.) Python also has the ability to work with JSON data structures so the two can pair extremely well. MongoDB is also used by professionals around the world in fortune 500 companies down to individual novelists. The community support is top tier and its ability to adapt to scalable projects is great. As for building out this project, if you were to do this here are the steps you would need to take. Setup, Modeling, Dashboard Design, Interactivity, Testing, Deployment. Your setup you would need to build out a project directory, make sure that all your dependencies are in order. Pycharm, Apporto, Jupyter, etc. Also make sure that you have your packages and libraries (PyMongo, Dash, etc). Now onto Modeling. You would need to build out a class for Animal Shelter for your operations with MongoDB. You would also need to create methods that handle CRUD or create, read, update and delete. You should also handle a secure connection to the DB with a username and password. Next you need to be able to retrieve, analyze, and manipulate the data from the DB for visualization. Taking a look at that dashboard now, you would need to design a layout, build out your headers, title, display sections, etc. You should also be thinking about what filters you want to add, whether it be checkboxes or dropdowns or both. Also, time to think about the visualizations for the data, I think this project included a pie chart so that will need to be presented. Moving back to the code you’ll need callback functions to make all of this filtering work. Testing out everything you’ll need to do manual QA on the actual project itself while also looking for any errors or spaghetti code in your JSON. And lastly deploying the application to be used. The last question you asked were what challenges I had to overcome, well the biggest to date is jupyter/apporto. I have been having issues with it since day one actually. I emailed you about it during the first week of class. While I know I am too late at this time to fix anything with it, I was still able to build out my ipynb file, take some screenshots and write up a readme. It’s not what I would have wanted for my final project but I need to be agile and work with what situation I have at hand.