Justin Chevalier

CS340 Client Server Development

Southern New Hampshire University

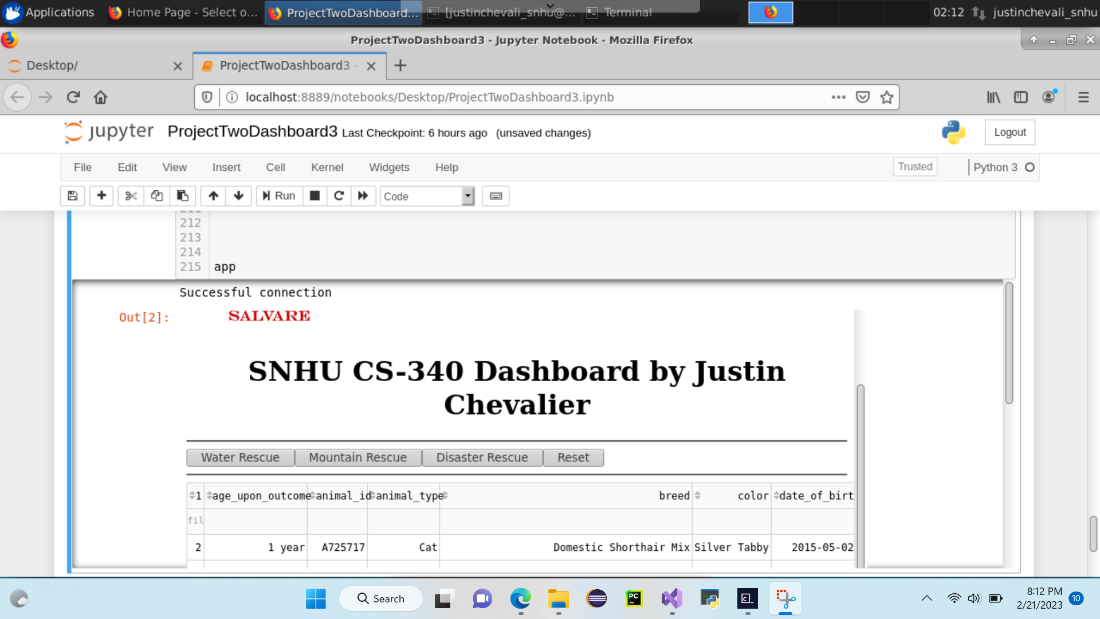
February 21 2023

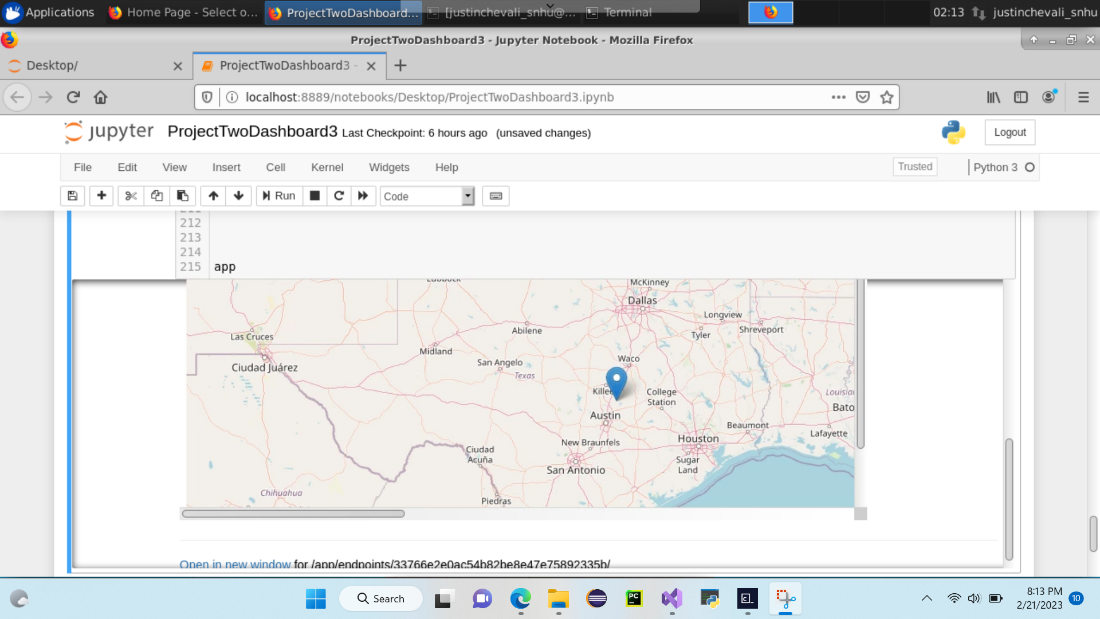
Project Two Read Me

The Grazioso Salvare Dashboard

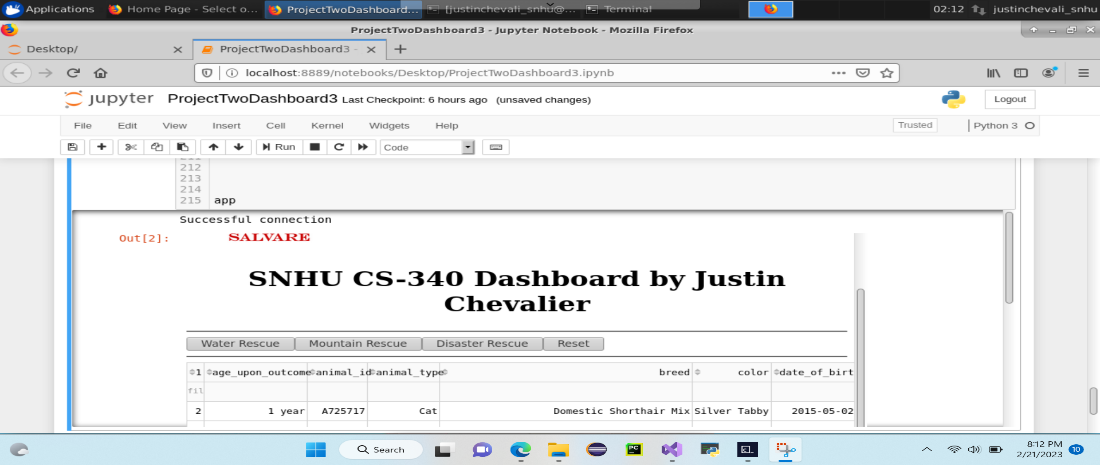
The Grazioso Salvare Dashboard was created utilizing the features of python to provide users a simple but powerful way to sort through potential dogs that may qualify for the training program. The Grazioso Salvare Dashboard features the dash framework that was built upon the python syntax and showcases the CRUD functionality that was created. The CRUD functions uses the attributes create, read, update, and delete to alter or change based on the request. The goal for creating the dashboard was to easily find dogs who fit the description of what the training facility is searching for in a rescue dog.

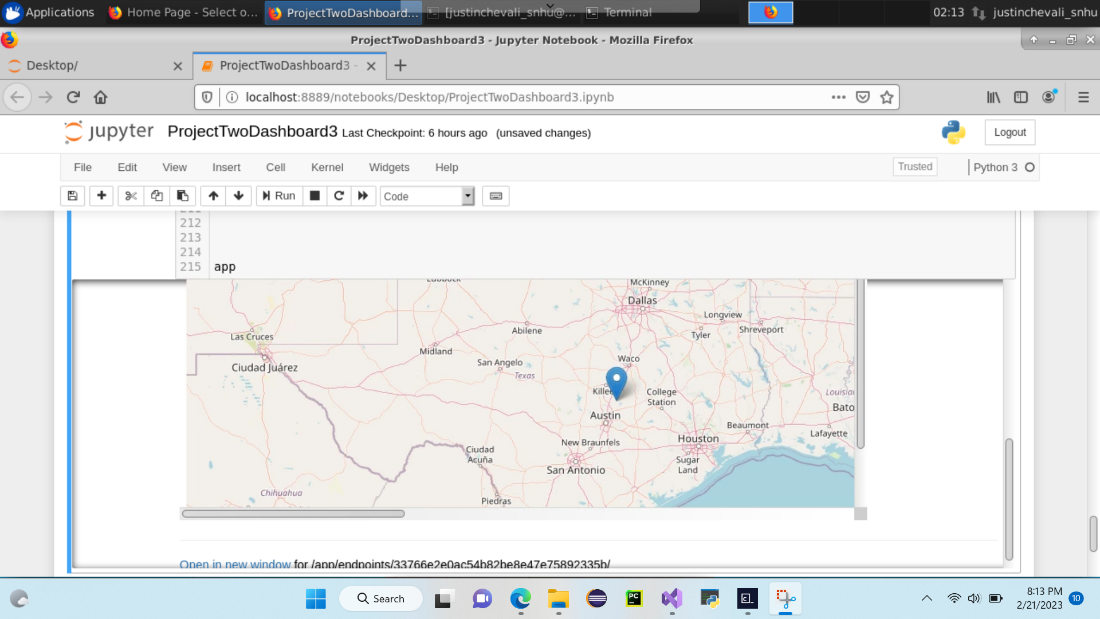
To start the search for all dogs that can be relevant to the particular operation for water rescue



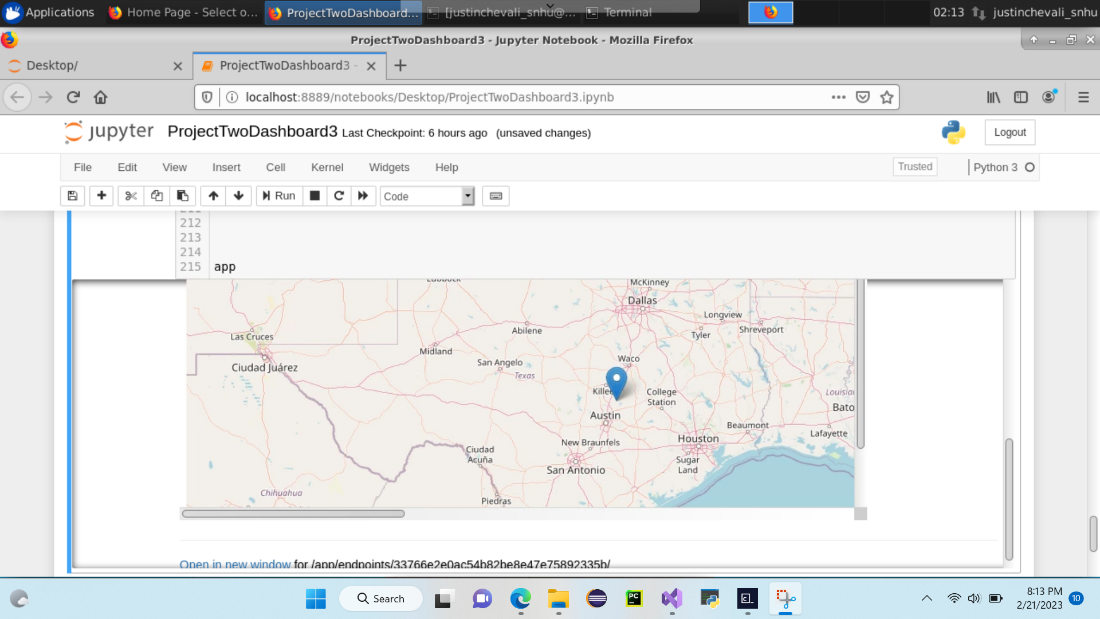
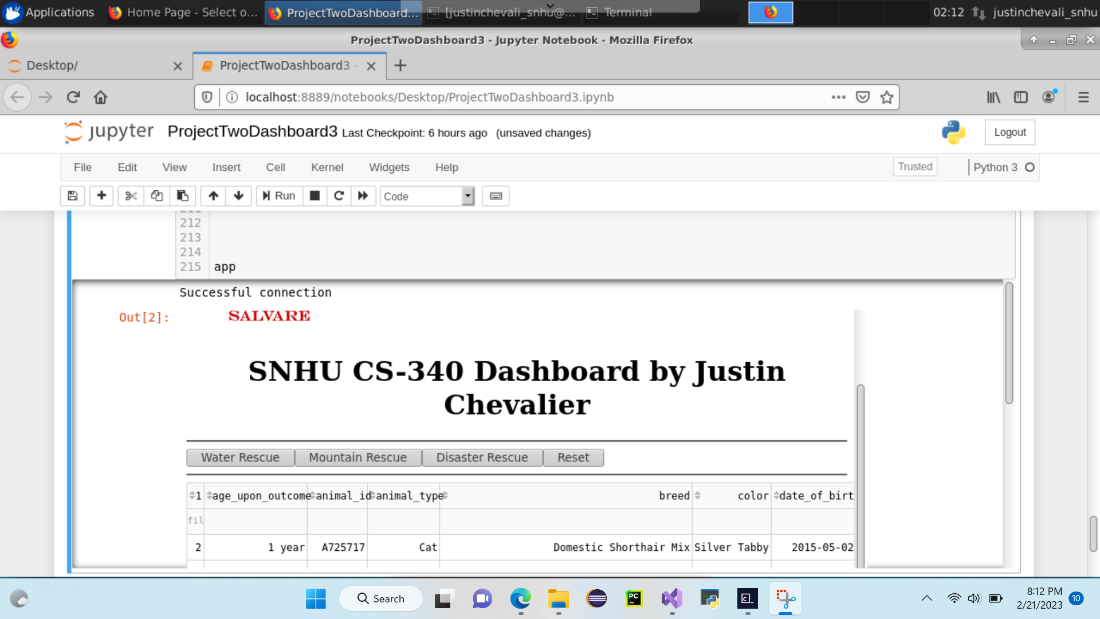


The next search would be for the Mountain Rescue dogs to filter.





The Disaster Rescue filter button for the Grazioso Salvare Dashboard will filter out all dogs but those who fit the disaster rescue description.



The tools that were used to complete the dashboard included the dash framework. This provided the structure for the dashboard. Jupiter Notebook was used for the implementation of the python script responsible for the CRUD functionality. Pandas was utilized to access the dataframe and with the addition of matplotlib these two are the reason our dashboard has visualization of the data that’s being requested. Mongo database was utilized to store our data and we used the mongoclient import from pymongo to bridge the mongo database with the Jupiter Notebook so our dashboard could access the data required directly from the database.

The steps that were taken to complete the project included first creating the CRUD functionality that was going to connect the mongo database to the Jupiter notebook so my dashboard could access the aac collection that was housed in mongo. Once the CRUD functionality was implemented and I was able to effectively create, read, update, and delete data I started working on rendering a dashboard with the capabilities to use the CRUD functionality. Once the dashboard was successfully connected to the mongo database it was time to structure the layout of dashboard. I went with the button functionality for the dashboard because my thinking was just to get the dashboard working and the examples were utilizing the button functions so that would provide a starting point. After creating the layout according to the requirements of Granzioso Salvare. Then I implemented the functions to make the dashboard interactive with the user.

My time producing project two came with numerous roadblocks. For starters I had trouble with the mongo socket. It kept me logged into mongo even when I stop it but the information in the general questions section helped me resolve that problem. Another problem I had was with the callback feature. I couldn’t get the dashboard to interact with the buttons and I figureed that it was because of the callback feature. When I would try to change the callback the screen would stay blank. I have worked on this part the longest and to no prevail I was defeated in accomplishing the task of making the dashboard interactive.