From: Piruz Alemi

To: Team #2

Subject: The Architecture of Project #2

Date: March 9th, 2020

Greetings to All. This architecture grew out of our discussion with Json\_TA and his guidance.

1. Briefly the full architecture consisted of a front end (Leaflet) + a Backend {SQL/MONGO DB} + Flask, as was noted by Illya:

A sign on a whiteboard

Description automatically generated

1. What we needed was to use Flask as an in intermediary step between SQL/MongoDB of our presidential elections, to safeguard our data or use a micro framework like Flask to cut the overhead, instead of accessing the full database directly.

Following code from presidents.js in line 247 connects to Flask url.

In the first place, to run presidents.js we had to activate

python -m http.server

The above command would open <http://0.0.0.0:8000/index.html>

The 8000 port, whereby we could activate the index.html.

However to run index.html that runs presidents.js we still needed to have made the

flask active. To activate the Flask component we did the following:

* 1. Open a new terminal window
  2. CD to where the Flask program (api.py) resided

A screenshot of a cell phone

Description automatically generated

* 1. Run python api.py

A screenshot of a cell phone screen with text

Description automatically generated

1. This breakdown also helps in division of labor and minimizing down-time and “localizing” the errors and working in teams.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. If you go to Git-Lab/ClassWork 17/logic.Js we have a good protocal of a program that reads from a url (line #2) (see below) which creates a jsonified file.
2. Once the file is jsonified, via d3.json(url), we can access the the key-\_value pairs and extract the needed data as input to Leaflet/front-end program, which in our case was called earthquake.js and will be called presidents.js
3. The earthquake.js file already reads a jsonified file of presidents.polling data that was created by AlemiMerge2.js program
4. Recall, The Alemimerge2.js program read a CSV file of presidential polling data and left joined it with State-GPS Coordinates files, based on State Codes. In addition it created an SQL data base.
5. We may change the program to write to a MONGODB. In this case we created presidents DB and added presidents\_coordinates\_poll.csv file to its collection with 18,296 documents

A screenshot of a cell phone

Description automatically generated

In our case line 2 becomes the Flask URL that was accessing/interfacing with DB: SQL/MONGODB

A screenshot of a social media post

Description automatically generated