Progress Report: Study Yelp

An application to help students find the best study spaces in their surrounding area based on their preferences.

Meerza Ahmed, mhahmed@usc.edu	Data collection, Firebase Implementation, Data Modeling (ER / EER), CLI Implementation
Melissa Perkins, perkinsm@usc.edu	Data Visualization, UI Design, Website Component Implementation, CLI implementation
Alan Trinh, alantrin@usc.edu	Data collection, Data Cleaning, UI Design, Data Prediction Modeling, Python queries, CLI Implementation

Implementation Questions

[Answer the following question as short as possible] [Do not exceed 5-10 lines] [Explain straight to the point]

- Which database are you using?
  - We will be using information from a yelp dataset (https://www.yelp.com/dataset) and inputting it into Firebase RealTime Database.
- What is the approach for Distributed scaling of data?
  - Our database is quite large so we will be using a hash function to efficiently retrieve the data
    - We are still finalizing the exact hash function that we will be using
- Which application did you choose to implement?
  - We will be creating a website to help students find the best study spaces in their area. Students will be able to search based on their zipcode and sort / filter by certain features (ie ratings, etc).

Planned implementation

1. Dataset & Data Cleaning
   - Download the Yelp dataset and perform data cleaning, handling null values, features selection, etc
2. Database Setup
   - ~~Setup Firebase databases (2) will separate the data by Eastern and Western states.~~
   - Setup Firebase databases (at least 3) using 3 separate datasets that came from the Yelp dataset.
   - We initially used firebase SDK to setup our hosting and talk to our firebase real time database but it was always our plan to do the data administration portion of the project in python writing queries.
3. Backend Development
   - ~~Create a hash function to split the data by East and West~~
   - Write Python functions for data management (add, delete, modify) study spots

- ● Create hash function that will create smaller databases to make data retrieval more efficient and avoid crashing (which has happened when we load the entire database)
- ● Express.js framework

4. Develop CLI interface
   - ● Write Python functions to develop a CLI menu for data managers for adding, deleting, and modifying
5. Frontend Development
   - ● ~~Deciding between Javascript or Python framework~~
   - ● Using Angular to generate the front-end website, which will make calls to the RealTime Database
6. Testing & Debugging
   - ● Unit tests
   - ● Documentation
   - ● User guide and code documentation

Status of Project

Implementation:

Data cleaning: Our 3 Yelp datasets are cleaned and ready to use. Features have been selected, Unicode has been converted, and the data has been nested for search. Datasets have been implemented into our database

Front-end: The homepage is set, up with a sidebar, components have been created, icons have been found, need to implement tiles for individual study locations, and need to include a filtering feature

Back-end: Our database is stored in Firebase, and we are going to change how we communicate with our front-end. The project will still be hosted in Firebase but we won't be relying on the Firebase SDK to query our data and have it appear on the front-end. We won't be using any of Firebase's products to implement features in our project. The data will be stored in the real-time database so that we each have access to it for our parts of the project, all data pipelines queried, and the user interaction we built will be done using Python.

Challenges

Implementation:

Data cleaning:
- Could not use it out of the box for Firebase
- Had to convert Unicode and apply lowercasing and double quotes
- Nesting features

Front-end:
- First time using Angular, quite different from Vue
- Tried to merge a template but did not work due to different versions that were used in the template and the Firebase project set-up.
- To find answers to questions and get some assistance with debugging, I've been searching Google, Stack Overflow, Angular documentation, example projects, and ChatGPT.

Back-end
- We need to add the remaining tables to our database, also our yelp dataset is very large so we need to write appropriate rules as well as parsing functions so that our table returns data correctly.
- We also have to be careful of Firebase free-cost plans limit to not go over the limit, and incur a charge.

Timeline

Weekly Meetings: Mon & Thur (Started on 1/25)

Newly added items in green

Removed items in red

February - 02	Submit project proposal ✅
February - 03	Start date cleaning ✅
February - 09	Finish Firebase skeleton and upload Yelp dataset into dedicated tables (one table uploaded) ✅
February - 23	Build a working front-end skeleton (the home-page is running) ✅
March - 01	Finish data cleaning ✅

	Midterm Report
March - 06	Incorporate hash function, display backend information on the frontend Ex. Show business, address, hours, attributes
March - 08 March - 10	Program and test data queries; Upload all 3 datasets to Firebase; Data to be displayed on the frontend
March - 15	Build the function calls for the CLI part of the project
April - 06	Finish front-end display working data, and test user interactions
April - 12	Testing / Debugging
April - 17	Implementation & Demo
May - 03	Final Report