**Final Project: Final Report**

**README**

Project code: <https://github.com/jingcao33/SI-507-Final-Project>

In order to run the app, you have to request a Yelp Fusion API first. You have to have a yelp account to log in to perform further actions. After you log in, you can create a new app using the link below.

<https://www.yelp.com/developers/v3/manage_app>

You can simply run this app in your local computer with any browsers.

http://127.0.0.1:5000/

Python packages this project used includes: requests, json, secrets, bs4, csv, sqlite3, flask, and plotly.graph\_objects.

**Data sources**

1. San Francisco Restaurants Score (csv)

<https://data.sfgov.org/Health-and-Social-Services/Restaurant-Scores-LIVES-Standard/pyih-qa8i?row_index=0>

This dataset contains the records of each inspection of restaurants in San Francisco, CA, provided by the Health Department. The dataset is a public source available on DataSF. There are more than 50 thousands of records in this dataset. The time periods it covers is from October 2016 to October 2019, plus one record in November 2019. I only chose the data from November 2018 to October 2019, and removed all the null values and unnecessary variables for my project. So the final dataset “Inspection” contains 38684 records and 7 columns, including “id”, “business\_id”, “business\_name”, “business\_address”, “business\_zipcode”, “inspection\_date”, and “inspection\_score”.

I used csv library to access the dataset and loaded the data to the table “Inspection” in the database “sf\_restaurants.sqlite”. “business\_id” is the original id in the old dataset. “inspection\_date” is for data cleaning purpose. “business\_name” and “business\_address” are used for matching to Yelp Fusion API. “Inspection\_score” is directly used for later data visualization.

1. Yelp Fusion API (JSON)

<https://www.yelp.com/developers/documentation/v3/business_match>

<https://www.yelp.com/developers/documentation/v3/business>

I want to check if the sanitary condition of a restaurant has a relationship with the evaluation from customers. So I use Yelp Fusion API to extract those information for the restaurants in the database. Not all the records in the “Inspection” table can be retrieved and due to the access limit, the total number of restaurants I retrieved is 3480. And the total number of categories I retrieved is 182.

I first used request library to find the business id for each restaurant in the Yelp business match API. Then used the id I retrieved to get detail information about this restaurant in the Yelp business details API. I cached all the information to a json file “api\_cache.json”. To be specific, I used the information of “name”, “zipcode”, “is\_closed”, “phone”, “review\_count”, “rating”, “price” and “categories” in the API. All of those will be displayed as a text or graph in the final visualization.

1. SF Department of Public Health

<https://www.sfdph.org/dph/EH/Food/default.asp>

I used requests and BeautifulSoup to scrape the above URL to extract the contact information about making a foodborne illness complaint. I want to give the user an option to contribute to the inspection dataset.

**Database:**

I added the information of “name”, “zipcode”, “is\_closed”, “phone”, “review\_count”, “rating”, and “price” to the “Business” table. I created a separate table called “Cat” to track each kinds of category of the business. Since one business may have zero to several categories, and one category could have many businesses. So I created a third table “Categories” to track the relationship between table “Business” and table “Cat”, only containing two foreign keys.

Besides, each restaurant may have zero or several categories, like “Coffee & Tea”, “France”, etc. So, I created another table “Cat” to save those values.

**Interaction and Presentation Plans:**

I’m going to use Flask to display numbers using HTML tables and graphs using plotly.

1. User can choose one category in a drop-down list, then the website will display 10 random restaurants in that category in San Francisco.
2. User can choose either rating, review counts, or inspection score, then the website will display a bar chart of top 10 restaurants based on the choice. Besides, there will be an input box to let the user enter one of the restaurants above and display more details, like restaurant name, address, phone, rating, review\_count, inspection score, is\_closed.
3. User can choose which kinds of complaint they want to make, then the website will display the related contact info.