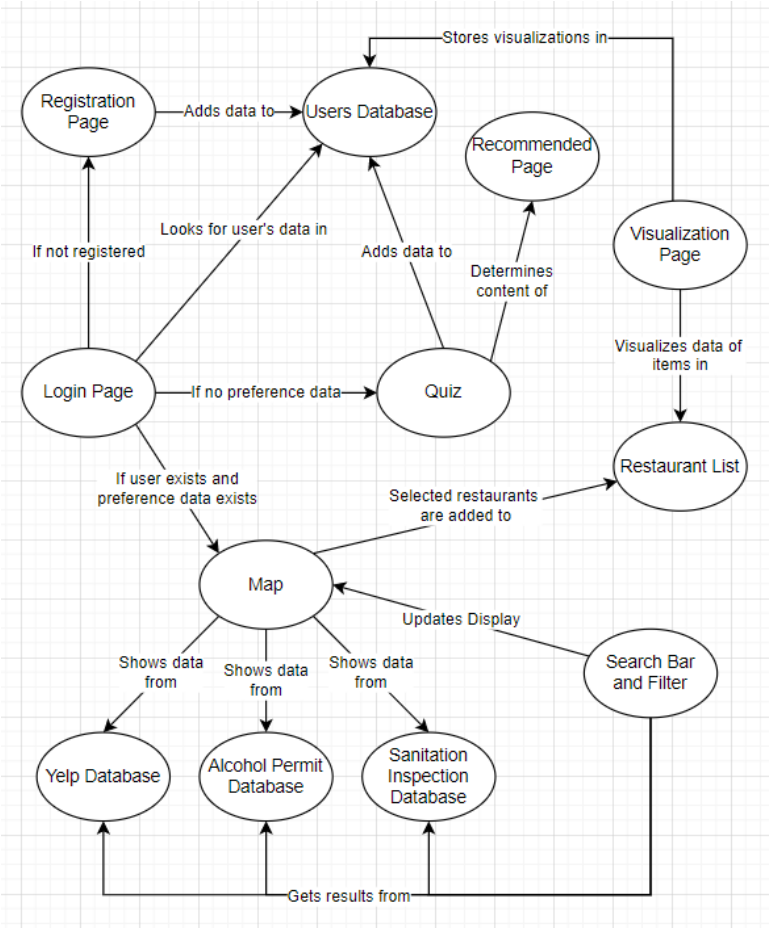
Electron: Elizabeth Paperno, Jeffery Tang, Kevin Li, Abid Talukder
SoftDev
Period 7
2023-05-03
Target Ship date: 2023-05-24
NYC Restaurant Data Visualizer

Program Components:

- templates/
  - login.html, register.html
    - Allows users to login and register
  - preferencesQuiz.html
    - Allows users to list preferences, so we can later provide personalized recommendations (using forms)
  - dashboard.html
    - Map with filters like food category, takeout/delivery, parking, location, sanitation grade, location, etc.
    - Search bar to search restaurants by name or address
    - Data about restaurants in map radius
      - Graphical representations of food categories etc.
  - myRestaurants.html
    - Displays saved restaurants and visited restaurants (allows users to review them)
  - pastVisualizations.html
    - A page to store saved visualizations (snapshots of filtered data from dashboard) so it can be compared to other visualizations
  - recommendations.html
    - Recommended page to show a list of restaurants based on restaurants the user has previously visited and given good reviews to which also includes brief descriptions of the restaurants like a star rating, seating availability, food category, and dietary restrictions
- __init__.py
  - Runs app
- users.py
  - Deals with user data
- restaurants.py
  - Deals with restaurant data
- db
  - users.db
  - restaurants.db
- static/

    - css/style.css
    - js/script.js
- keys/
    - maps.txt

Component Map

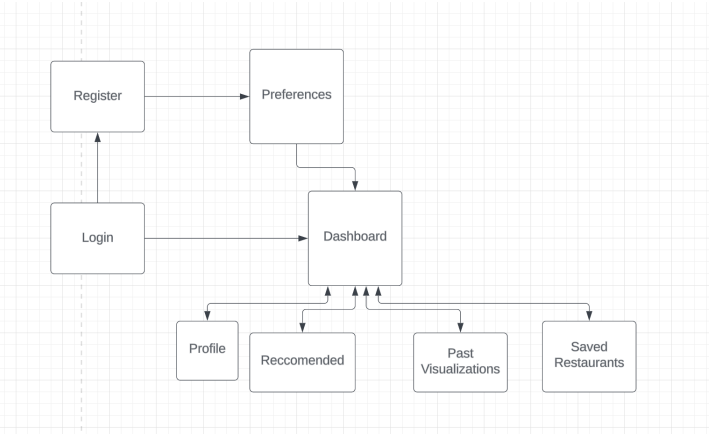


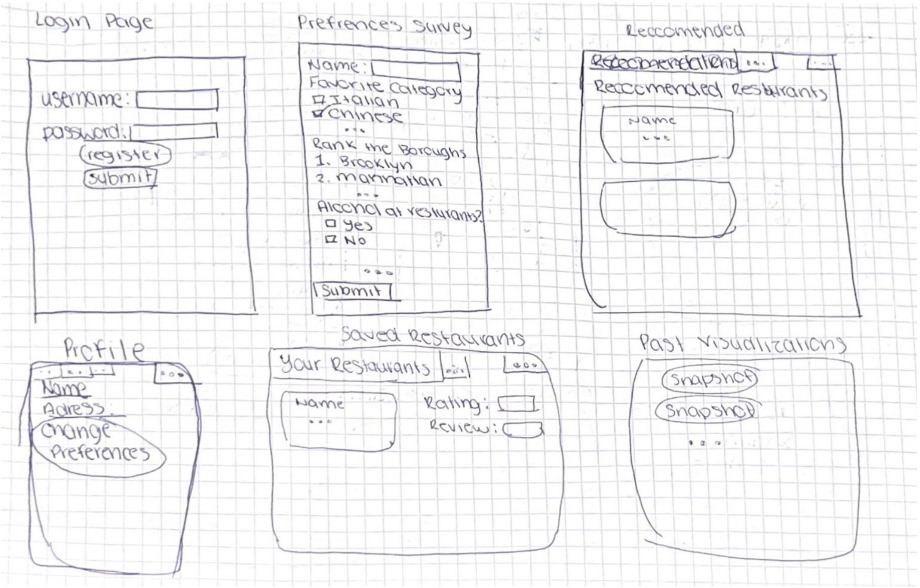
Relation Between Components

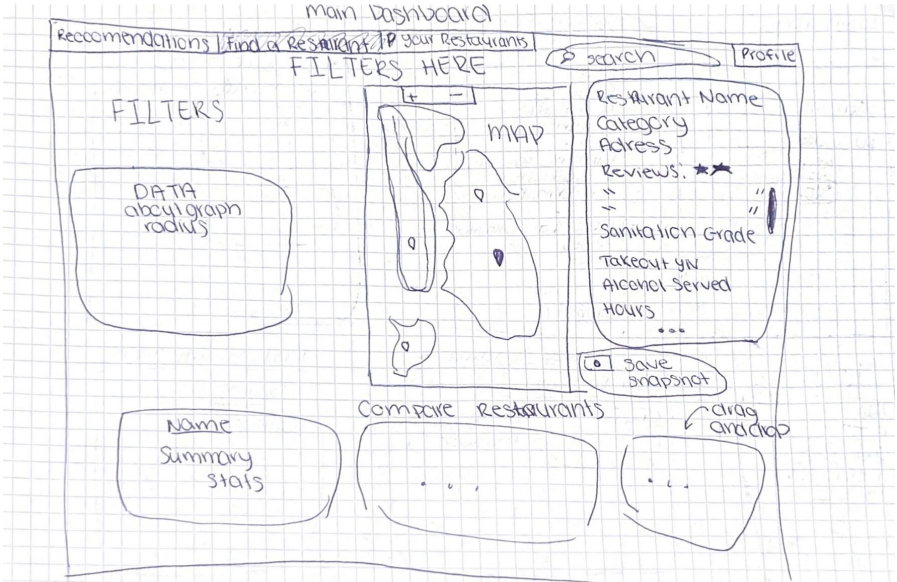
- Login and registration pages create accounts with their own lists and saved visualizations
- Quiz for what users like in restaurants + past reviews → section for recommended restaurants
- Allows users to save restaurants they've been to and write reviews
- Display yelp reviews + sentiment of review
- Map displays filterable information about all restaurants in view
- In the map, users can select a set of restaurants to save to a list for data visualization later. Users can also choose a location and select all restaurants in a radius around that location
- Search bar takes users to a specific restaurant on the map

- Users can take a list they've created and visualize the data, then save that visualization to the users database

Site Map







APIs

Embedded Google Maps API (subject to change)

- https://developers.google.com/maps/documentation/embed/get-started

Database Structure

- USER TABLE
  - username TEXT
  - password TEXT
  - restaurants saved TEXT
  - restaurants visited TEXT
  - reviews TEXT
  - visualizations ?
  - Food category: [item0, item1] LIST
  - location TEXT
  - borough preference (ranked 1-5) DICT (like object)
  - alcohol preferences BOOL
  - preferred restaurant seating TEXT
  - dietary restrictions LIST
- RESTAURANTS TABLE
  - Restaurant name TEXT
  - Address TEXT
  - Takeout BOOL
  - Parking BOOL
  - Categories LIST
  - Hours DICT (like object)
  - Reviews DICT (like object)
    - Stars
    - Tag for tone
    - Date
    - Text
  - Alcohol BOOL
  - Sanitation inspection grades TEXT
  - Seating type TEXT
  - Dietary restrictions LIST
  - User reviews LIST

Databases Used

- Yelp Database - JSON: https://www.yelp.com/dataset/documentation/main
    - business.json
        - Name
        - Address
        - Takeout
        - Parking
        - Categories
        - Hours
    - review.json
        - Stars
        - Tag for tone
        - Date
        - Text
- Active and pending alcohol permits - JSON: https://sla.ny.gov/public-query
    - Active alcohol permits
        - Premise name
        - Method of operation
- Sanitation inspection grades: https://data.cityofnewyork.us/Health/Restaurant-Grades/gra9-xbjk/data
    - Inspection Results
        - DBA
        - Violation description
        - Grade

Front End Framework

Using JS and potentially Node.js to provide easy interactivity with the page for the user.

- JS: interaction with user
    - Event listeners
    - Events
    - Functions
- Bootstrap

Tasks

- Front End Design (JS, Bootstrap) - Abid, Kevin
- Database Work - Jeffery, Elizabeth
- Flask - Kevin, Jeffery
- API - Elizabeth, Abid