TNPG:Senior SLEePeR
Roster: Ryan Lau, Lauren Lee, Elizabeth Paperno, Sasha Shifrina (PM)
SoftDev Pd 7
P05 - 🥦
2023-05-30
Time Spent: 3 hrs

Idea
A web app that aggregates prices of produce onto our website to allow the customer to compare prices from their nearby grocery stores.

- User will be able to search and select stores based on their location
- Search for produce, and price compare between stores selected
- Add to cart and view cart breakdown

Possible stores:

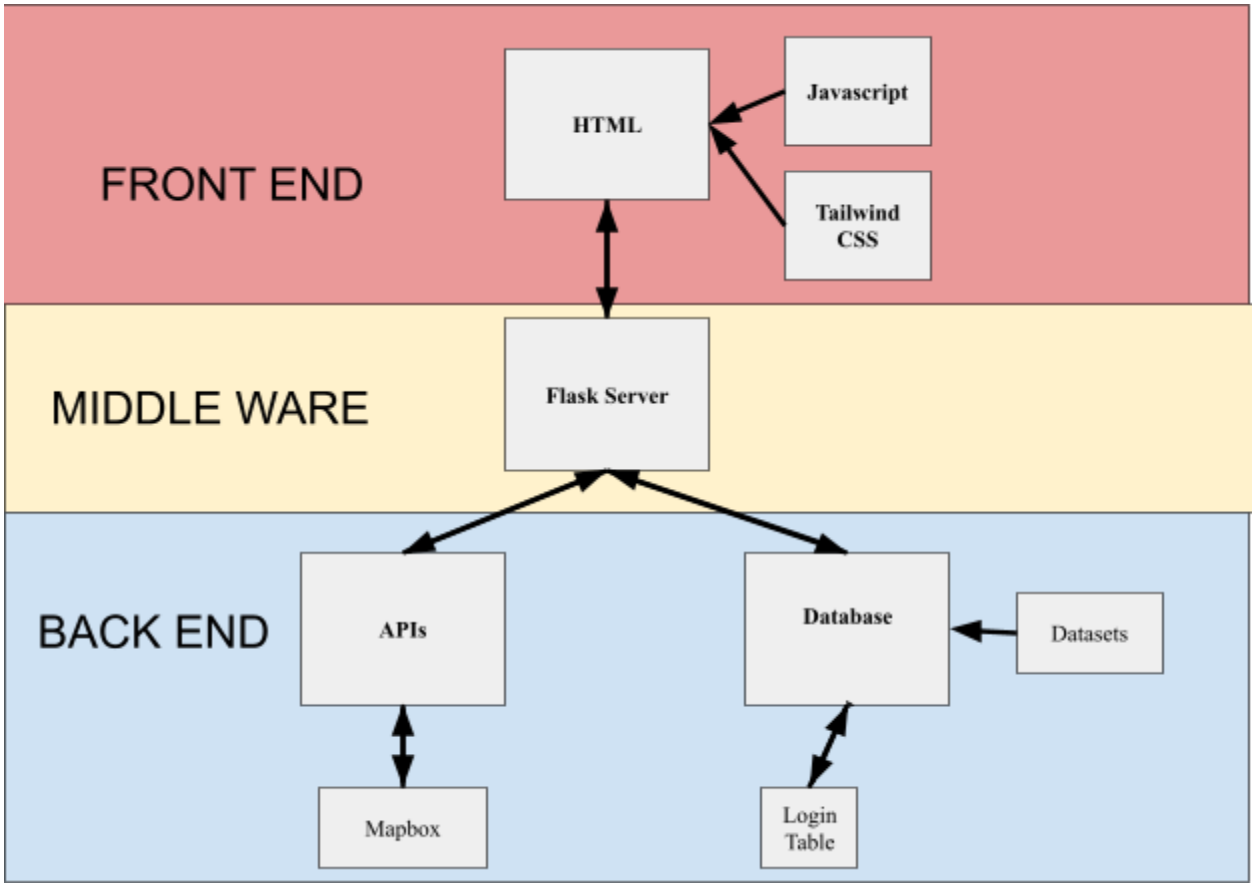
- WholeFoods
- Walmart
- Target
- BJs
- Costco
- Kroger
- TJ
- KeyFood
- Wegmans
- Stop&Shop
- ShopRite

Program Components

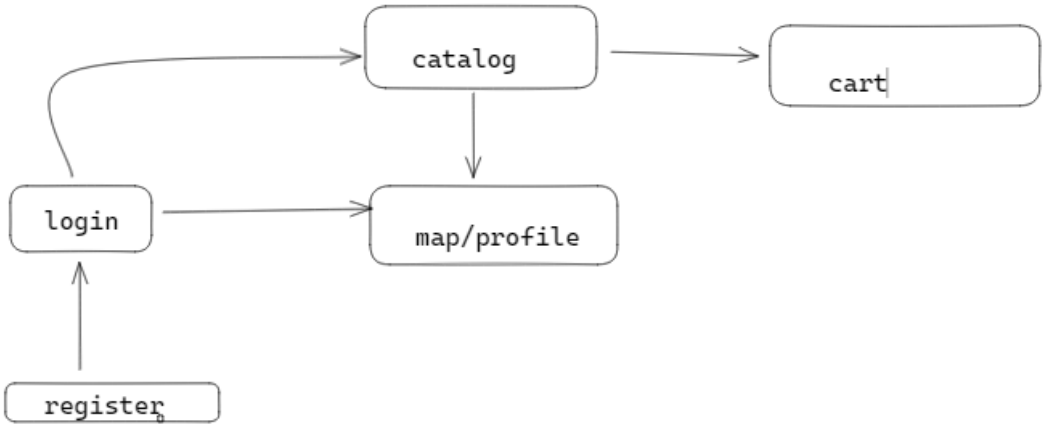
- app/
    - templates/
        - login.html
        - register.html
        - catalog.html
            - **Potentially** search by image, would use existing pytorch image recognition model
        - results.html
        - cart.html
            - Each item selected, how many items at each store, best route (potentially)
        - profile.html
            - User chooses home stores
    - static/

        - css/
            - style.css
        - js/
            - script.js
    - db/
        - P5.db
            - Database itself
            - Contains user, cart, and produce tables
        - auth.py
            - Login, register, etc.
        - cart.py
            - Functions to work with cart
        - produce.py
            - Create produce table through pulling scraped data
        - input_check.py
            - Checks username and password strength

Component Map



Sitemap



Database Structure

Users Table

username TEXT	**password TEXT**

Produce Table

id UNIQUE INT	**product_name TEXT**	**img_url TEXT**	**weight FLOAT**	**price FLOAT**	**store_id INT**

Cart Table

username TEXT	**id INT**	**quantity INT**

Stores Table

id UNIQUE INT	retailer TEXT	retailer_id INT	lon FLOAT	lat FLOAT	address TEXT

API

- Mapbox to display stores on a map so users are able to choose stores visually

FEF

- TailwindCSS, Ryan is more comfortable with it compared to Bootstrap or Foundation

Potential Problems Faced:

- SKUs are not standard
- 3lb onion vs 1 onion - compare only those that are the same, display all onion options,

Roles

- Frontend: map (working with mapbox api), render pages, cart page
    - Ryan Lau
- Backend
    - Lauren: scrape supermarket data (probably split further)
    - Sasha: produce db using scraped data
    - Elizabeth: users db for login, query db/search
- Flask: Sasha, Ryan