PulsePlus v2
Pak Ming Lau, Ryan Ma, Renee Mui, Victoria Gao
P3 - ArRESTed Development, JuSt in Time
2021-04-23
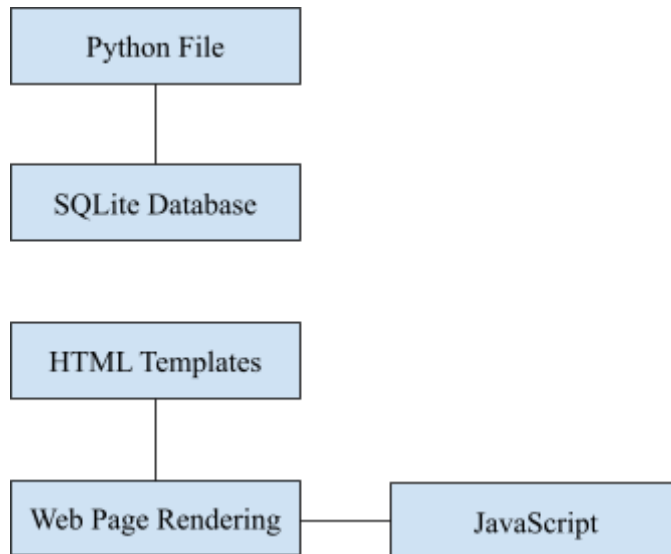
Program Components
- **Flask**: The framework for the web-based app which connects the frontend to the backend
- **SQLite Database:** Caches information from API requests
- **HTML**: Styles the frontend pages
- **Python file**: Connects Flask with SQLite and our web pages
- **Bootstrap**: Adds extra styling to HTML pages
- **JS**: Makes the web pages interactive.

APIs:
- NYT Movie Reviews
- MovieDB
- StudioGhibli
- Yelp
- Edamam
- Google Maps Geocoder

Component Map

Databases

Movies table:

Title	Genre	Description	Review
Text	Text	Text	Text

Recipes table:

Name	Cuisine Type	Recipe
Text	Text	Text

Restaurants table:

Name	Cuisine Type	Reviews	Phone Number	Website
Text	Text	Text	Text	Text

Site Map

Home Page

- All features are accessible through the homepage
    - There will be a checkbox for different genres of films that the user wants to watch, that will then show a listing of films and their reviews in that genre(s)
    - Underneath the film would be recommendations of restaurants or recipes to food
    - Search bar to search for a specific movie through its title and returning reviews and recommended food (case-sensitive and spelled correctly)

Breakdown of Tasks + Deadlines
- Victoria: HTML forms, CSS and Bootstrap
    - 04/23 - Have basic HTML
    - 04/25 - Add instructions above fields about format of inputs, warning messages
    - 04/26 - Add CSS and style webpages
- Ryan: Recipes-related API, databasing, JS
    - 04/25 - be able to return a random recipe from the API given cuisine type
    - 04/26 - Write functions to cache data from APIs

- Renee: Restaurant-related API, databasing, JS
    - 04/25 - be able to return a random restaurant from the API given cuisine type
    - 04/26 - Write functions to cache data from APIs
- Pak: Movie-related API, databasing, and JS
    - 04/25 - be familiar with API and be able to search for movies on the API and their reviews, then return that data to the website
    - 04/26 - Write functions to cache data from APIs
- 04/27, "functioning" website
    - Test for bugs and functionality