HL DESIGN

Team No Stress
10/06/2021
Daniel Ramirez
Luke Sunaoka
Matthew Quinn (Team Leader)
Michelle Vo
Tyler Thorin

.

1 Project Overview

1.1 Purpose

The High Level Design (HLD) provides an analysis overview of our project, What the Food?, and breaks down its overall functionality by components.

1.2 Audience

The High Level Design's main target audience is the project group members Daniel Ramirez, Luke Sunaoka, Matthew Quinn, Michelle Vo, Tyler Thorin.

1.3 Design Process

The design of our What The Food? revolves around our primary feature within the app which is the scanning feature built from Javascript. From there we designed a database using SQL(Structured Query Language**)** to search an item that has been scanned and store any additional information necessary. The database can also be accessed through a simple user input search. What The Food? is primarily a web application and therefore requires Internet connection in order for the app to be used. The web application will be created from HTML 5. Furthermore, the user interface will be created from Javascript and HTML 5 and CSS allow for quick access to the app's multiple features which include scanning a product, providing alternatives to a product, allowing a user to filter out or flag ingredients, providing a personalized daily AMR calculator/customized nutrition percentages, and aggregated news presented by the preference of the user.

2 Requirements

- To provide users with easy access to a large amount of information about the products they consume.
- Provide users with the information they need to make more educated choices
- Give users tools to discover new products that better fit their needs and lifestyle
- Help users avoid products containing ingredients they believe to be unhealthy, or otherwise wish to not consume.
- Quick load times and easy access to core features
- Intuitive UI that just about anyone can pick up and understand
- Comprehensive SQL Database containing many of the most popular food products that we expect to be scanned
- Centrally located Web Server to minimize latency to the user

2.1 Proposed Solution

- Through the use of HTML, CSS, and React JS, What The Food? Will provide simple transition amongst each feature as well as accessing the back end of our application

2.2 Capacity Planning

- We will separate the web server database and the SQL database server in order to minimize strain on our hardware through over consumption of resources.

3 Architecture

3 Design

3.1 Web Application Architecture

- What The Food? is a web application and more specifically a progressive web app that will support Google Chrome exclusively. Users will be able to access the app with or without a login but with varying features amongst the two.

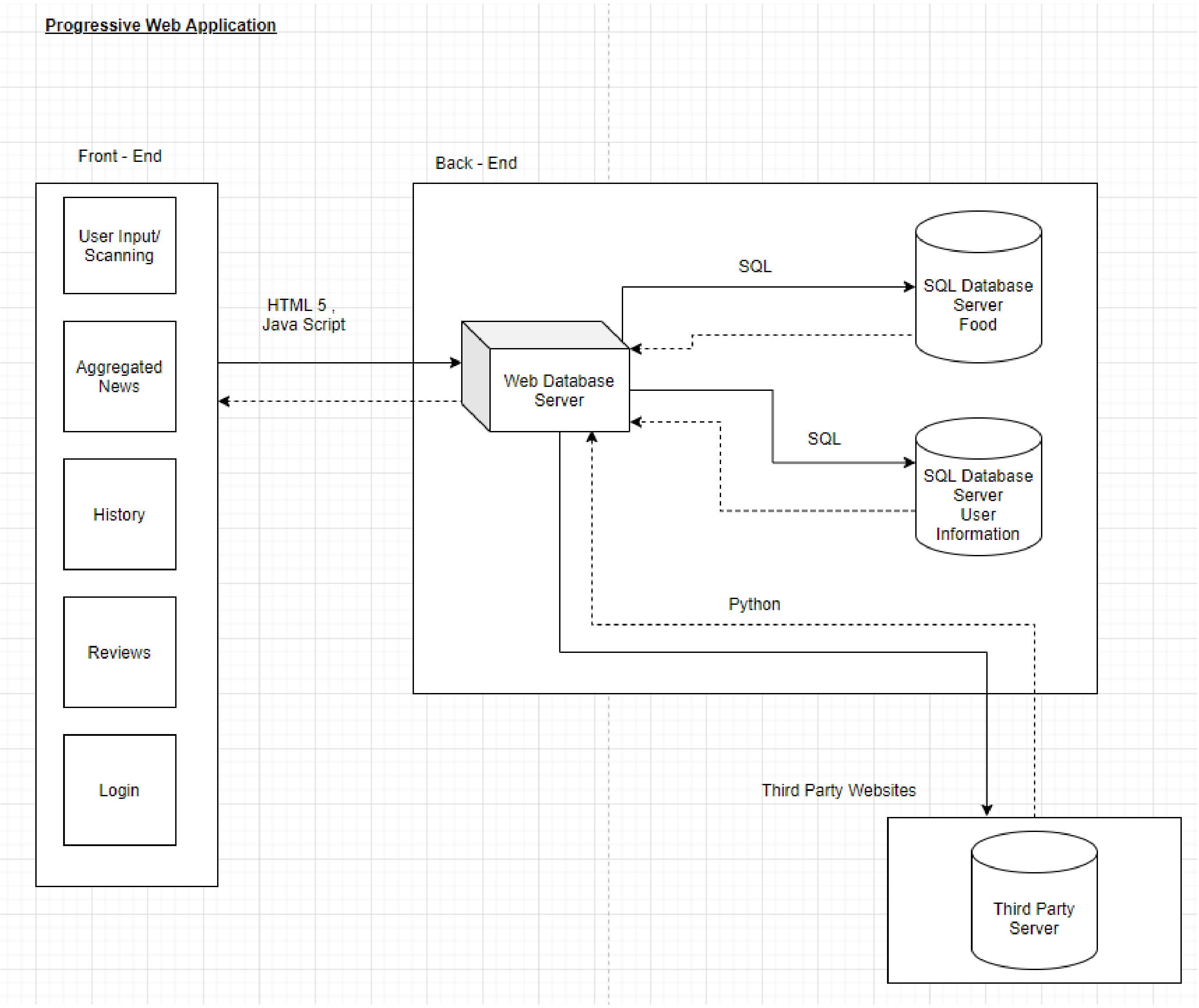
3.2 Presentation Layer

- Users will be presented with simplified information regarding their product upon scanning or searching as well as given an informational graph which compares their previous scanned products to recommended health guidelines.

3.2 Data Access Layer

- The database will be accessible by all members of the group listed in 1.2 as well as the application itself. Users will only be able to retrieve information through the use of the application and not directly.
- Front End Specification
    - PWA format that will display everything for the user as well as create any interactions between the client and the application.

- Back- End Specification
  - Web Server that handles any incoming request made from the client
  - SQL Database server that stores all information regarding our application which will be called on by the web server when necessary.



3.4 Hardware and Platform Requirements

- Front End
    - Progressive Web App
        - What languages we Use
            - SQL
            - HTML 5
            - CSS
            - Java Script
            - Python
- Back End
    - Web Server
        - JavaScript
        - Python
    - Database Servers
        - SQL

3.5 System Connectivity

- The user will be able to connect to the app using only Google Chrome from any device. Compatibility will be ensured for Chrome on Windows, Android version 94.0.4606.61, or iOS with Google Chrome version 94.0.4606.52. Furthermore, if an update occurred that deprecated an essential component, or the user's installed version is too old, the user will be prompted to change to a version of chrome that our application supports

4 Standards

4.1 Security Standards

- We will uphold a certain standard of security to prevent any breach of access throughout our application

4.2 Disaster Recovery

- Upon crashing or any unexpected error, the application will return to its last safe state. Aswell as present the user with a notification explaining that an error occurred.

- In the event that the application is not able to return to a safe state, the application will instead inform the user that the application is currently unavailable and to try restarting or to come back later.