Project01: ArRESTed Development
TNPG: APInsomniacs
Roster: Amanda Tan, Naomi Lai, Jacob Lukose, Kishi Wijaya
TARGET SHIP DATE: 2024-12-20

DESIGN DOCUMENT

A. Description

1. This project implements a website all about food, with the following:
2. Visitors will be able to access:
   a. Catalog of foods with nutritional information, substitutions, and product searches
   b. General information on cooking tips
   c. Catalog of recipes
   d. Page of recent food news
3. After registration, logged in users will be able to:
   a. Find grocery stores near user
   b. Favorite catalog items
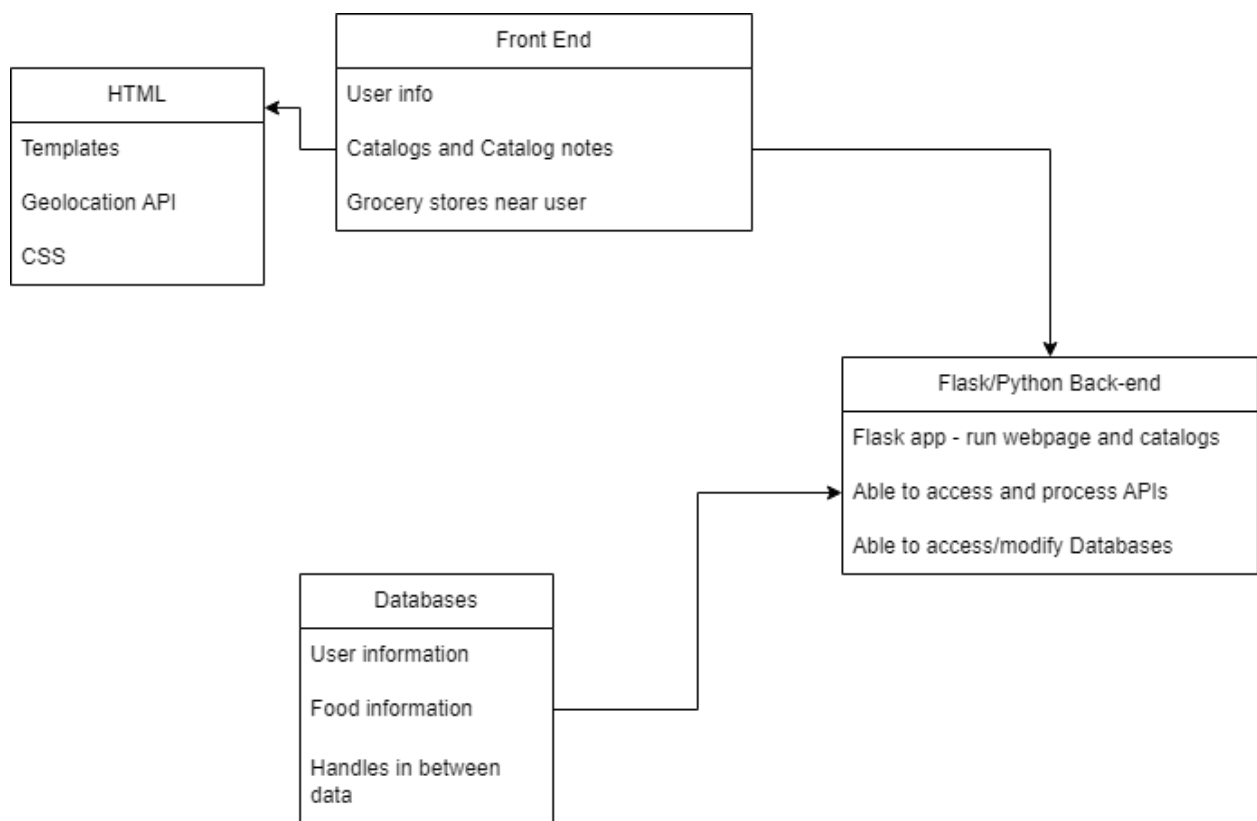   c. Write notes on catalog items visible to user

B. Functionalities

1. Management of Individual Favorites Page:
   a. Ability to favorite and unfavorite information on foods, recipes, news article entries
   b. User ability to leave notes to personal reference
2. Home Page: Page with all the existing food entries
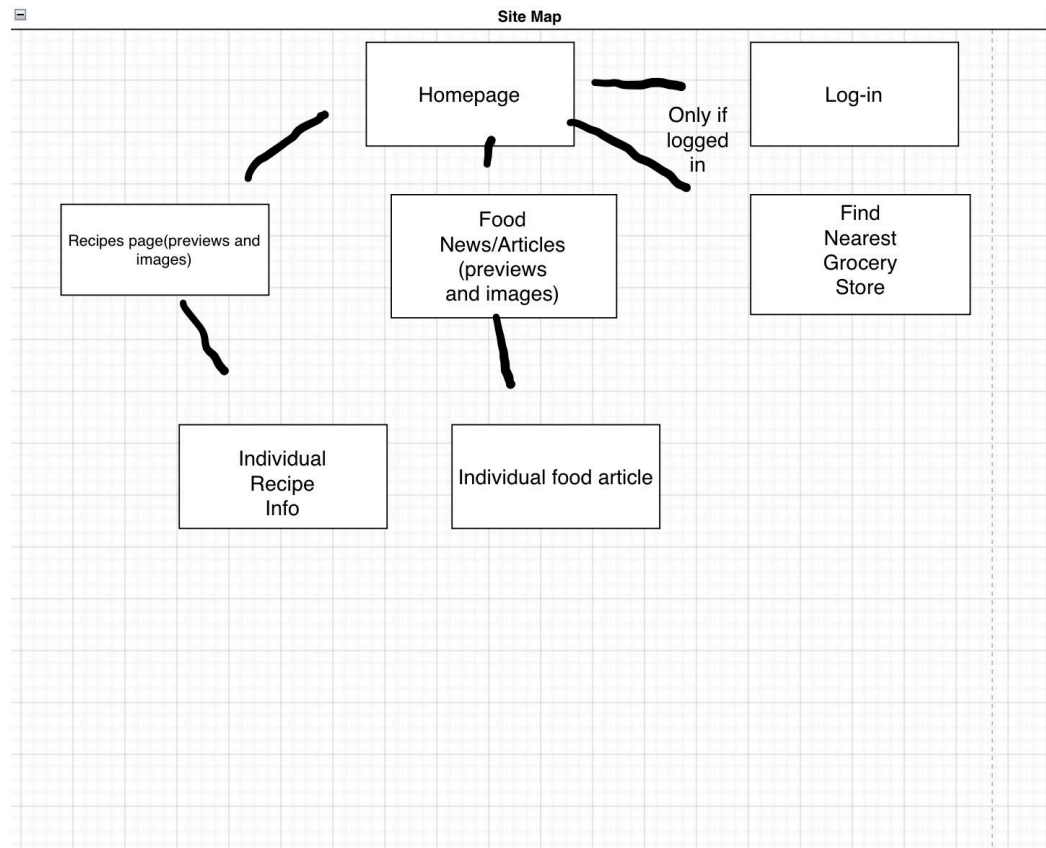   a. Uses links to bring to each page

C. Program Components

1. Flask/Python:
   a. Serves the websites, handles HTTP requests, manages user sessions and authentication (login), controls logic surrounding permissions
   b. Interacts with various tables within the database file to fetch and store user and API data
   c. Coordinates Flask routes (between HTML pages) and middleware functions (ex. Module for storing data into tables, creating user sessions, etc) to control application flow
2. SQLite3 (Database):
   a. Stores user information (usernames, passwords, permissions, etc)
   b. Stores data about each food page in a unique table

        c.   Data used by middleware functions and HTML templates
3. HTML: Provides user interface with links to access different pages
      a.   Renders content passed by Flask routes/database
      b.   Displays form for user interactions and works with Python to create the various actions
4. Jinja: Create templates for pages with room for customization
      a.   Used by Flask routes to have pages with default formatting (keeps things consistent and makes it easy to create pages for new content)
5. CSS: Create a consistent theme for pages across the board
      a.   Works with HTML documents

HTML

Templates

Geolocation API

CSS

Front End

User info

Catalogs and Catalog notes

Grocery stores near user

Flask/Python Back-end

Flask app - run webpage and catalogs

Able to access and process APIs

Able to access/modify Databases

Databases

User information

Food information

Handles in between data

D. Site Map

Site Map

Homepage

Only if logged in

Log-in

Recipes page(previews and images)

Food News/Articles (previews and images)

Find Nearest Grocery Store

Individual Recipe Info

Individual food article

E. Database Organization

1. User Table
   a. User Id/Username
   b. Password
2. API Info Tables
   a. Page ID
   b. Name
   c. Image
   d. Recipes

F. APIs

1. Spoonacular
   a. This food API that provides information on ingredients, recipes, nutrition, allergens, and more.
2. New York Times API
   a. This API provides information such as monthly statistics on stories, NYT article database, and bestselling books.

  b. We plan to use the 'Top Stories' section and filter for the food category to access recent food-related news.

 3. [Location API](#)

  a. This API returns information about the location and details of various Kroger grocery stores, including department and store hours.

 4. [Geolocation API](#)

  a. The Geolocation API provides the user's location for web application use.

  b. We will use this API in combination with the Location API to find the user's nearby grocery stores.

G. Front-end Framework

1. We will use Bootstrap as our front-end framework because of its aesthetics and simplicity in achieving desired appearance. It is intuitive to use.
2. We plan to utilize Bootstrap's responsive grids to easily format articles, recipes, and information. We are also interested in customizing Boostrap's default colors and fonts to suit our design.

H. Task Breakdown

1. Amanda Tan: Project Lead
   a. Manage overall system/file structure: Work with backend and frontend to make sure everything works smoothly
   b. Pick up any work that isn't explicitly delegated/falling behind schedule
   c. Assist with roadblocks in any area
2. Naomi Lai:
   a. Make sure routing flows smoothly and web page reflects accurate API information
   b. Organize individual favorites page (database/page)
3. Jacob Lukose:
   a. Create databases from API info
   b. Login page and database.
4. Kishi Wijaya: HTML & CSS
   a. Create HTML templates
   b. Create CSS templates
   c. Work with the Geolocation API and the Location APId