ECE 49595 Senior Design Proposal

Team Member Names

Nathan Bitner: nsbitner@purdue.edu
Atin Dewan: dewan2@purdue.edu
Ishan Kumar: kumar488@purdue.edu

Name: YourPalate
Idea: Meal and Shopping Planner
Repository: https://github.com/bitNathan/YourPalate

Abstract

Meal planning and grocery shopping can be time-consuming and overwhelming tasks, particularly for college students. To address this, we propose YourPalate. Our ECE 49595O senior design project is an open-source web application that automates the process of meal planning and shopping list creation by utilizing machine learning algorithms, k-nearest neighbors, and public APIs, YourPalate generates personalized weekly meal plans based on user preferences, dietary restrictions, culinary skills, and time constraints. Unlike traditional recipe platforms that focus on individual meals, YourPalate integrates user data to produce a balanced and diverse meal schedule that encourages culinary exploration while accommodating time and budget limitations. Additionally, the application offers a seamless user experience with features like ADA-compliant design, downloadable meal plans, and dynamically updated shopping lists. We will implement YourPalate using Django, TensorFlow, and SQL as the primary technologies, demonstrating its potential as a valuable tool for enhancing dietary habits and simplifying meal preparation.

Introduction and System Features

In this project, we will develop a user-centric web platform where individuals can complete a short “cuisine test” and input their weekly schedule along with choosing from a list of dietary restrictions we can account for. Based on this data, the system will generate a weekly meal plan that balances familiar

dishes with new culinary challenges, tailored to the user's time constraints and dietary restrictions. Each recipe will either match the user’s favorite recipes or be selected by our predictive model to suit their preferences and abilities. An auto-generated shopping list will accompany the meal plan. Users will also have the flexibility to replace or adjust meals, with corresponding updates to the shopping list. Both the meal plan and shopping list can be exported in formats like CSV or PDF for easy sharing and printing.

To begin using the program, the user inputs key information. This includes their schedule, specifying how many meals they typically eat and how much prep time they prefer for each. Next, the user completes our “cuisine test”. This is a short survey of how much they would like to prepare and eat a variety of foods over the next week. The user also selects any dietary restrictions from a list we provide, this could include veganism, religious observances, or something as simple as disliking fish. Budgetary constraints could also be considered here, although the program focuses on the practical aspects rather than the reasons behind these choices. Another crucial piece of information is the user’s daily caloric intake goal, which is especially important if they are cooking for others, such as roommates or family. Finally, this data is stored in a file, such as a CSV or JSON, which the user can update and re-upload later. This allows users to gradually discover and incorporate new recipes into their preferences.

In addition to user inputs, the program considers several baseline considerations. While not specifically marketed as a fitness product, the meal plans will prioritize macronutrient breakdowns that align with government recommendations and ensure a regular inclusion of fruits and vegetables. Furthermore, perishable items, such as salad mixes, fruits, and certain vegetables, will be scheduled earlier in the week to prevent spoilage and reduce waste.

Gap Analysis

A simple Google search for a recipe recommendation returns many web-based applications that claim to give new recipes to try based on available ingredients. Others aim to recommend new recipes based on your preferences.

AllRecipes and Yummly are popular platforms that offer a large number of recipes, but they serve their users differently. AllRecipes provides a community-based approach where users can search and filter through a large database of recipes submitted by users. It includes reviews and images from users, which help gauge the quality and appeal of recipes. It mainly focuses on individual recipes rather than personalized meal planning. Yummly takes a more personalized approach by offering recipe recommendations based on users' dietary preferences, allergies, and favorite cuisines. It uses these inputs to filter recipes, making it easier for users to find a meal that suits their preferences. Yummly also focuses on individual recipes with more refined searching and filtering for users.

Contrastingly, our application will automate the decisions and planning that many people feel take too much time or outside knowledge to make. YourPalate is designed to offer recipe suggestions and incorporate them into a meal plan for the week that can be adjusted to fit dietary restrictions, culinary skills, and time. Unlike AllRecipes which focuses on community-based recipes and lacks personalized meal scheduling, or Yummly, which personalizes recipes but does not incorporate meal planning, our tool helps to automate the process of meal planning and grocery shopping. This takes the hassle out of deciding what to cook and buy. By guiding users in making healthier meal choices and optimizing their grocery lists, we help form sustainable habits, reduce food waste, and save money. We also remove the daily worries of "What's for dinner?" and "What do I need from the store?" We offer value to students or busy individuals who want to enhance their diet and culinary skills without overwhelming their schedules while gradually improving their culinary skills and making home cooking more viable.

Use Cases

- Use Case 1: Learning to Cook
    - Users can be given meals they have not made before so that they can improve their abilities and become more comfortable cooking
- Use Case 2: Losing Weight

    - A restriction that a user can give is their calorie goals. Users will be given a meal plan that will help them attain their weekly goals to help them lose weight. They will be able to track the calories they eat on a daily basis to know if they are on track with their weight goals.
- Use Case 3: Saving time
    - Creating a meal plan by hand and creating a shopping list can take a large amount of time away. By automating the process users will save time from cooking each day and also planning their meals.

Requirements

- Users will be able to input dietary restrictions, time restrictions, and caloric needs into the website. This will allow us to filter recipes from the database that do not meet the users preferences.
- The website will ask questions about food preferences and restrictions to create initial user data which the model will use to give recommendations. This initial data helps to give more accurate recommendations for a new user.
- The model will generate a recommended plan and a shopping list that the user can download from the website. This allows the user to access their meal plan without having to go to the website.
- The model should output a meal plan based on the user's input requirements within 60 seconds.
- Users will be able to regenerate meals if they do not like the meal recommended to them so the user can have control of the meal plan and select a plan they are happy with.

System Context Diagram

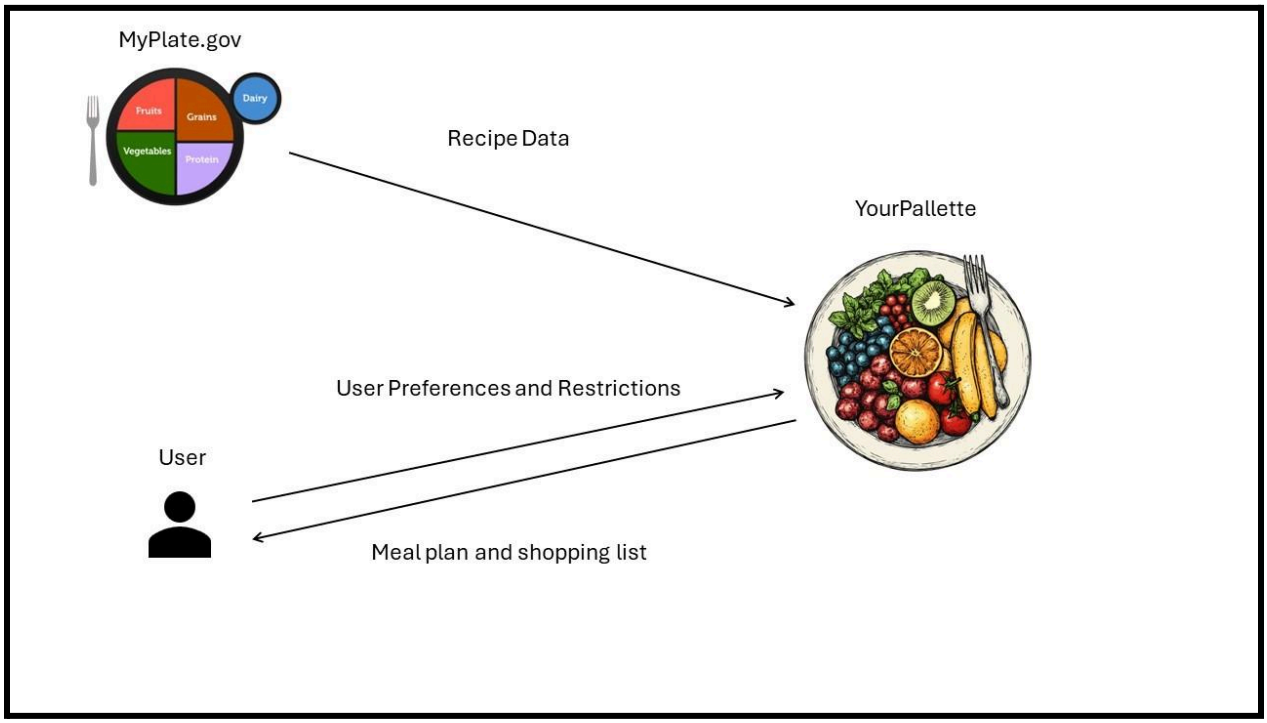

Fig. 1. Context of the system in relation to data source and user.

Software Component Diagram

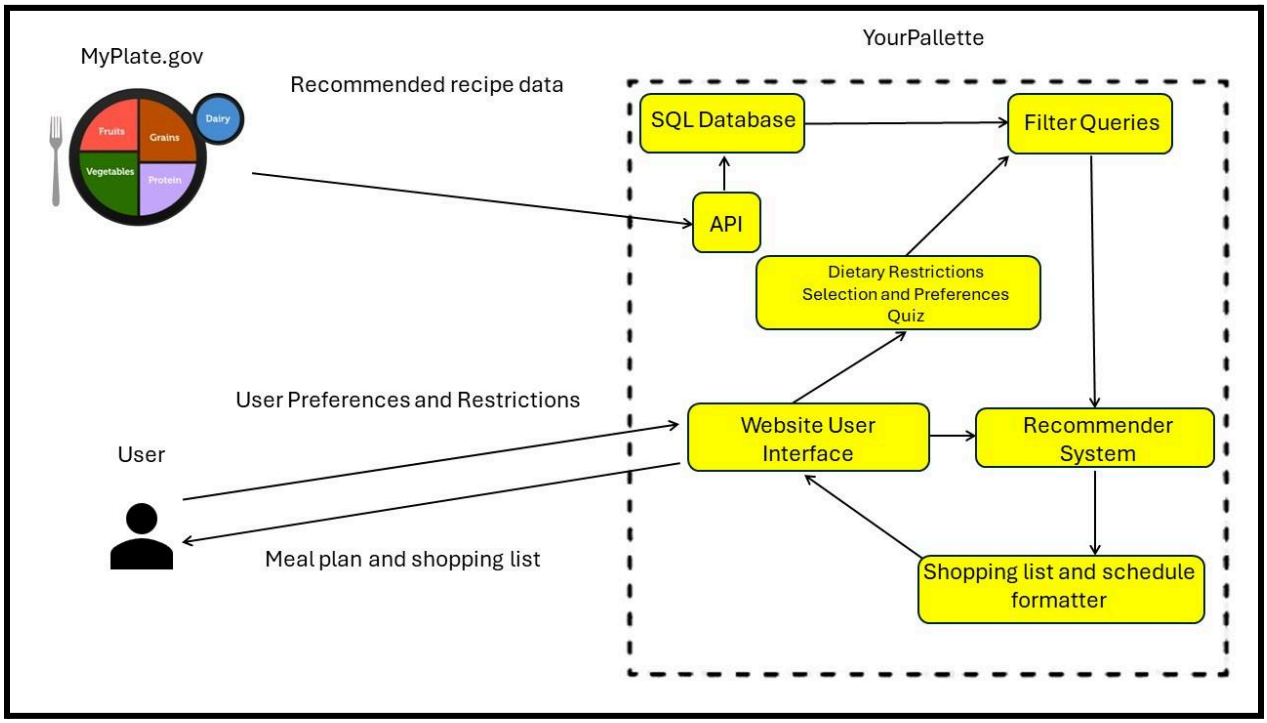

Fig. 2. High level system design components and dataflow.

Open Source Libraries

- **Tensor Flow GNN or Pytorch Geometric**: Used to develop a recommender system, primarily a k-nearest neighbor algorithm, to find the most similar recipes that meet our requirements.
- **Django**: Very common high-level web framework, also includes REST API framework to pull our data with.
- **Scikit-Learn:** Useful for any machine-learning training and includes functions for data splitting and processing, though the bulk of our analysis may be too complex for many of their algorithms.
- **Python Pandas**: Essential for data handling and manipulation, enabling us to manage and analyze large datasets effectively.
- **SQL:** To be used for our local database where we store recipes that will be recommended to the user. Our API will scrape data from MyPlate.gov to populate the database.

- **Beautiful Soup**: Since our API will be scraping data from MyPlate.gov, we'll be using this Python library for web scraping, allowing for parsing HTML documents.
- **Requests**: Useful for sending HTTP requests to access and download web pages when scraping data.