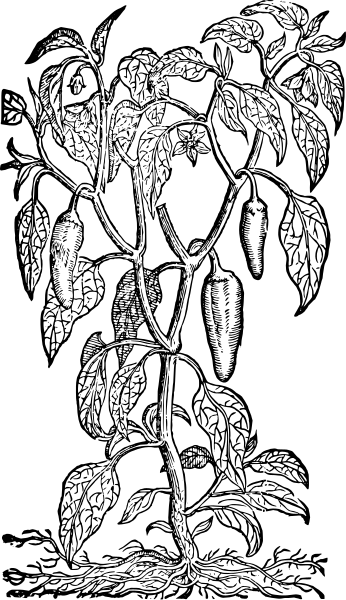
The Recspicy Index



Concept Document

Team #1

James McGuigan

Steven Forester

SUNY College at Old Westbury

# **Table of Contents:**

[**Table of Contents: 2**](#_3t3ipg64oss0)

[**Problem Statement 3**](#_huxvvteqgfuv)

[Our Goals: 3](#_spwhe2y42yyw)

[**Mission Statement 3**](#_oc6asimnxpw0)

[**Project description 4**](#_pao0ionzzib9)

[Overview: 4](#_qfr1ssuviyoz)

[Project Structure: 6](#_ipq2mfhj1hc3)

[**Use Cases 8**](#_e1adhgvyre30)

[Use Case 1: Welcome Page 8](#_49ojohimlrdq)

[Use Case 2: Profile Accessibility, Features 9](#_lo8nafvudqnx)

[Use Case 3: Profile Management 10](#_l3nwjujrbhjv)

[Use Case 4: Recipe management 11](#_n1wawitkwcmu)

[Use Case 5: Home Page 12](#_hy2pwl4jkevs)

[Use Case 6: Data Management 14](#_kp3wug7ohy5r)

[Use Case 7: Meal Planner 16](#_mzaaxghlog1d)

[Use Case 8: Admin Page and Controls 17](#_c95vv5qupc2d)

[**Flow of Events Diagrams 18**](#_c4n4apt01xav)

[Flow 1: Landing, Registration, Sign-in, Sign-Up, Footer 18](#_44zhx0d1ts0i)

[Flow 2: Homepage, Header Navigation, Profile 19](#_2zpftjlf4ppy)

[Flow 3: Feedback, Contact, and Support 21](#_8p80y2630rjy)

[Search Engine 23](#_6di5ptgytumw)

[Class Diagram: 24](#_93ccy9fs4yqh)

[Front End 24](#_cyx7ur1uoj6w)

[Back end overview 25](#_dmu2bz78gxwq)

[Site Map: Recspicy 26](#_u8r7407vmy9b)

[**Appendix 27**](#_sxd1su9xdafq)

[Key Terms: 27](#_3phzn73ex0et)

[References: 28](#_vl8oglxs5pp)

# Problem Statement

Existing recipe management platforms lack essential features such as meal planning tools, advanced search capabilities, and easy-to-use interfaces that include dietary needs. This limits users' abilities to explore recipes while creating balanced meal plans, leading to wasted time and frustration. Platforms developed to post and share recipes do not provide advanced filtering options based on dietary preferences, cooking time, or ingredient availability, this makes it difficult for users to find suitable recipes quickly. On competitor websites, the lack of integration between meal planning and recipe exploration tools complicates the process, leaving users with no easy way to organize their meals effectively when dieting or in need of recipe planing, additional tools without focus creates difficulties for many users in this regard.

## Our Goals:

The goal of this project is to outperform competitors and to enhance user experience, provide comprehensive tools, and improve accessibility. To achieve these goals, we decided to include meal planning tools with features like multiple search criteria and seamless recipe integration. We will allow users to efficiently find and organize recipes that fit their unique dietary needs and lifestyle preferences. Unlike competitor platforms, The Recspicy Index will offer an extensive library of recipes with detailed nutritional information and superior search/filtering options, ensuring a more efficient and enjoyable user experience.

# Mission Statement

The Recspicy index is a recipe web system with goals of enhancing user experience through meal planning features. Our priority is simplicity, ease of navigational design, and well-structured filtering tools for the end user. We aim to create a platform that is convenient to all users. Our system design is aimed toward dietary needs and preferences, offering search capabilities within our recipe databases with added features of integration from additional nutritional information.

# Project description

## Overview:

We are developing a recipe management platform designed to enhance user experience through an easy to use structured interface. The site will include a landing page, sign-in, and sign-up functionalities. Users will have the option to sign in using Google, providing an alternative method for registration. After signing up, users will create a profile where they can upload images, edit their personal information, including email, bio, username, first name, and last name. An edit profile page will allow users to save changes, while a profile page will display this information, showcasing user stats such as created recipe posts, favorite recipes, and ratings on a scale of 1 to 5 stars for recipes within the database.

Each user's profile will consistently follow this structure. Users can create new recipes and add them to the database, either making them public or keeping them private. Private recipes will only be visible to the user who created them, else a sign-out option that returns users to the landing page.

The main page of the website will include a navigation bar present on every page except the landing page. This bar will provide easy access to:

* **The Home Page**: which is the main page of this web project allowing the user to explore new recipes.
* **Top 10 Recipes**: A list of popular recipes featured on our website.
* **A Meal Planner** : Tools to help users create meal plans based on their preferences.
* **A pre-filtered browse by list**: An index of recipes on our website with filtering capabilities organized by ingredients such as chicken, pasta, beef, fish, etc. by meal for breakfast, lunch, dinner, snack, or by dietary restriction.
* **A Random Recipe Generator**: To allow users to find a random recipe as a suggestion from our database.
* **A Full Recipe Database List**: A list of all available recipes within our accessible database.

The main page will display a grid of the top 5 recipes, each presented in a card format, a horizontal scroll will showcase 20 randomly selected recipe cards.

The footer will be accessible from every page the footer has accessibility to:

* The Landing page
* Contact Us
* Frequently Asked Questions (FAQ)
* Feedback submission
* Copyright (T&S)

For administrative access, a dedicated login will be available for admin users. This interface will provide admins with the necessary tools to monitor, edit, track, and assist website users as needed. Admins will have full control over various aspects of the platform, ensuring smooth operation and support.

When selected, each recipe will load to the view page with detailed information that displays saved recipe images, titles, ingredients, measurements, instructions, and notes. This page will also include features like favoriting recipes to a personal collection, rating recipes on a scale of 1 to 5 stars, and printing the recipe for offline use.

This project will include a search function accessible from multiple pages, including the main page, navigation bar, and individual recipe view pages. This feature will allow users to perform quick and efficient searches, finding recipes based filtering, dietary restrictions and other included filters as we develop further.

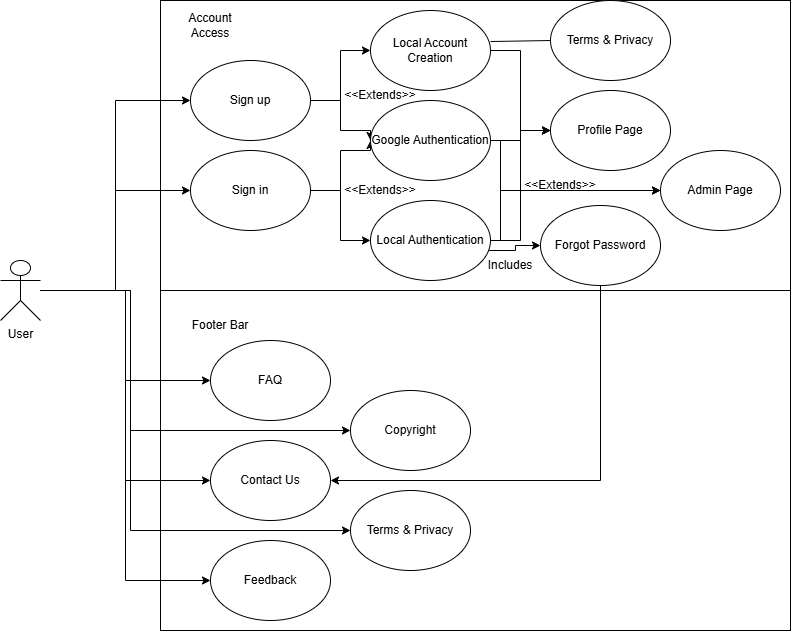
Our website is designed with simplicity, ease of use, and comprehensive functionality in mind, aiming to be a go-to resource for users interested in baking, cooking, and additional culinary interests, we are aiming to create an organized and efficient recipe management system.

## Project Structure:

1. Landing Page
2. User accounts
   1. Sign in
   2. Sign up
   3. Include google account login functionality
   4. Account
      1. Profile
         1. Edit profile
      2. Create recipe (accessibility toggle)
         1. Edit Recipe
      3. Meal Plan
         1. Active toggle
      4. Sign out > return to landing
3. Admin Page
   1. User and Recipe Statistics Dashboard
   2. User Management
   3. Recipe Management
4. Key Features
   1. Title Bar
   2. Navi Bar
   3. Search Bar
   4. Footer Fix to page locked
      1. About us
      2. FAQ
      3. Support
         1. Feedback
      4. Contact Us
      5. Copyright
      6. T&C
5. Home page
   1. Top 5 today
   2. 12 random recipes
   3. Our recommended (top 10 by rating)
6. Top 10
7. Random
8. Full Index
   1. Alphabetical Hyperlinks that return to view page
9. Meal Planner
   1. Create Meal plans, saved to profile
10. Search Bar
11. Search Page
    1. Filters
       1. Cuisine
       2. Meal
       3. Dietary Restrictions
       4. Cooking Time
       5. Allergens
12. Recipe Viewer
    1. Photo
    2. Rating
    3. Favorite Toggle
    4. Print Option
    5. Ingredients
    6. Instructions
    7. Authors Notes
    8. Nutritional Facts
    9. Tags

# Use Cases

## Use Case 1: Welcome Page



1. **Who is going to be using this page?**

Both end-users and admins may access the welcome page, but it is primarily designed for end-users.

1. **What is the user's reason for accessing the page?**

We want the end user to learn about the platform.

To encourage joining to explore the site's features and offerings..

Have the user sign up for an account or log in to an existing one.

1. **What is the purpose of the page?**

To provide a welcoming introduction to the website.

To inform users about the platform's main features and value proposition.

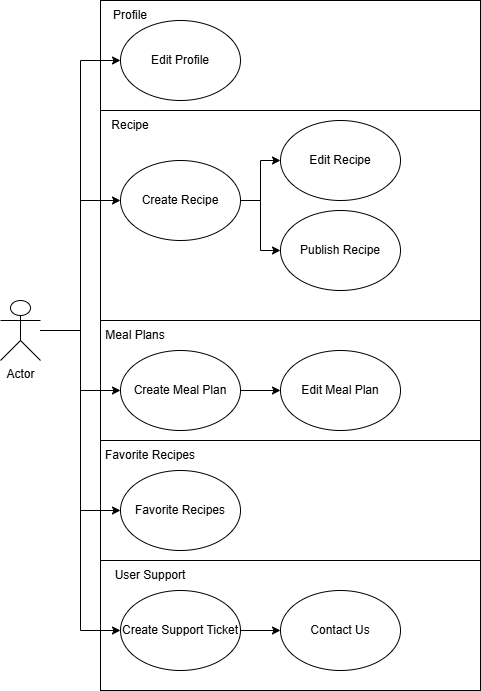
To guide users on how to navigate the site, including sign-up, login, or exploring the recipe database.

1. **Why is the user here?**

The user is new to the platform and wants to understand what the site has to offer.

The user may be interested in joining or exploring the learning about us before deciding to create an account.

## Use Case 2: Profile Accessibility, Features



1. **What can the user do on this page?**

Users can create new accounts by signing up.

Existing users can log in to access their accounts.

1. **What can the admin do on this page?**

Admins can assist in managing user accounts, including editing or deleting accounts.

Admins can assist with password recovery for forgotten credentials.

1. **What is the purpose of this page?**

Offering multiple login options enhances user convenience.

Users could prefer the ease of using their Google accounts for streamlined log-in, while others may prefer logging in with their own credentials or email addresses if google is not preferred.

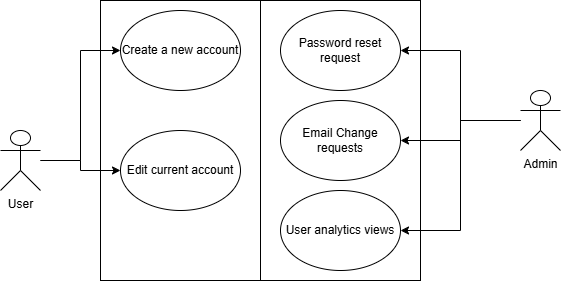
1. **How are we recording our data?**

An NoSQL database will be used to store user data efficiently.

The database will handle non-relational data such as user profiles, login records, and preferences.

Using a NoSQL like MongoDB will keep our project in range for scalability and flexibility.

## Use Case 3: Profile Management



1. **How does the user manage their own profile?**

Users can update their bio, add or change profile pictures, modify their email address, and adjust privacy settings for their profile.

They can manage their favorite recipes and ratings within their profile.

1. **How can the admin assist the user?**

Admins can assist users by resetting passwords upon request, unlocking accounts in case of too many failed attempts, and reviewing user profiles as needed.

There is a feature allowing users to flag their profiles for admin review if they suspect issues or need help.

1. **What support do we have?**

The profile page includes a help section with FAQs and a contact form for users facing difficulties.

A reporting system allows users to inform admins about any problems with the profile management process.

1. **What is saved?**

User profiles store information such as name, email address, bio, and profile picture.

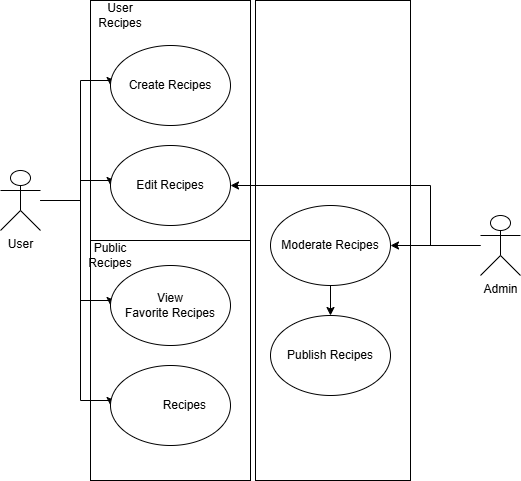
Favorite recipes and ratings are also saved within the user's profile.

1. **What can change?**

Users can edit or update their bios, email addresses, privacy settings, names, and manage their favorite recipes and ratings.

The system stores all personal data using appropriate measures for access by users and by admins.

## Use Case 4: Recipe management



1. **Purpose of the view page?**

Display detailed information about a selected recipe, allowing users to easily view and interact with the recipe's content.

1. **What can the user visualize?**

Recipe title and an image or thumbnail.

Ingredients list, including quantities and measurements.

Detailed step-by-step instructions for cooking or preparation.

Nutritional information and dietary information vegetarian, low-carb etc.

User ratings for the recipe and option to favorite the recipe to profile.

1. **What can the user do?**

Users can click a "Print" button to generate a PDF version of the recipe for easy reference.

They can save the recipe to their favorites list or add it as a private note within their profile.

Users can rate the recipe on a scale of 1 to 5 stars.

1. **What features will be included?**

Detailed recipe instructions with images.

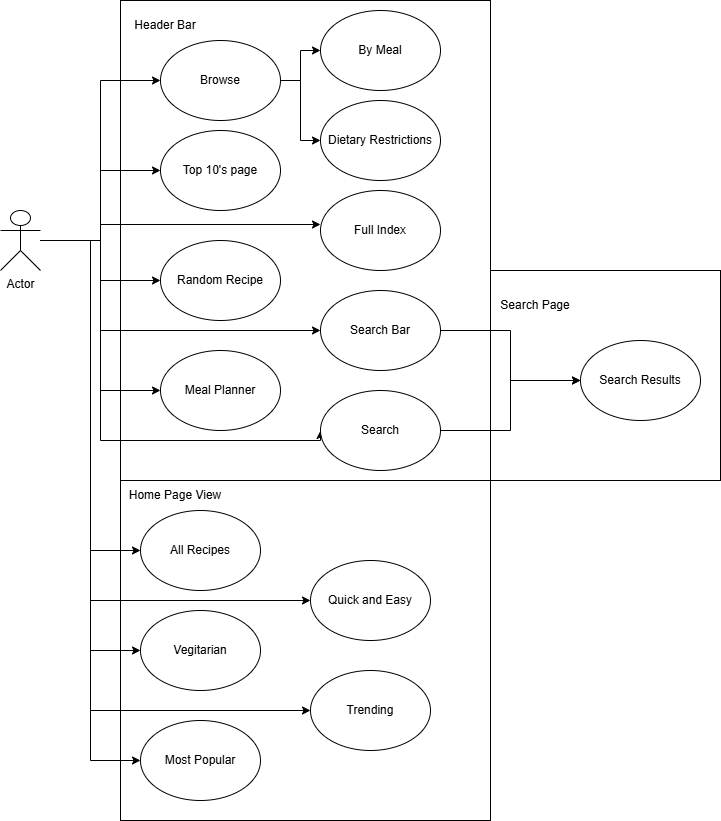
Possible implementation of embedded videos through youtube or other services.

1. **Why will our page be designed in a specific way/ design rationale?**

The design prioritizes readability and ease of navigation.

Print-friendly formatting ensures the recipe is easily accessible offline.

## Use Case 5: Home Page



1. **What pages will be accessible?**

Recipe of the Day, Popular Recipes, Recent Recipes, Categories such as by meal for, Breakfast, Lunch, Dinner, Desserts. Categories for dietary restrictions such as nt allergies, dairy, gluten, The users profile, top 10’s, Random Recipes, Meal Planner, Full index,

1. **What kind of filtering will we want to have?**

Users can filter recipes by category, dietary restrictions, preference, vegetarian or low carb, cook time or rating.

1. **Why do we have a welcome and a home page?**

The welcome page is intended as an incentive for exploration and introduction to the webpage for first-time visitors, providing an introduction to the site's features.

The home page includes selections of recipes, categories, and featured content, users are always greeted with relevant and new options each page load.

The purpose of the separation is to provide “exclusivity and inclusivity” within our project.

1. **Purpose of the navigation menu?**

To provide easy access to key pages of the website, including recipes, categories, user profiles, search, full indexes, and a random recipe generator.

The menu is designed for navigation without requiring users to scroll through multiple pages while keeping the pages uniform and consistent.

1. **Purpose of the search functionality vs. full index?**

Search allows users to input keywords or phrases to find specific recipes based on title, ingredients, or tags.

The search page provides filtered and refined results without needing to browse through categories.

The full index lists of all available recipes, organized by title and category.

Users can explore to discover new recipes from our index vs finding recipes from our search functionality.

## Use Case 6: Data Management



1. **What kind of database will we have?**

The project will use a NoSQL database, we will either use MongoDB or for key-values Redis. We chose this because it handles unstructured data, recipe, and user profile information.

1. **How many databases will we have?**

We will have three separate databases:

A Recipes Database for title, ingredients, steps, serving size, etc.

This will include a quarantine edit for admin purposes.

User Profiles Database for username name, email, password, etc.

Logs Database for activities like login attempts, user actions, or system events for auditing purposes.

1. **What is the information that will be CRUD in the database?**

**Recipes**:

* Create: Submitting a new recipe.
* Read: Searching or retrieving existing recipes.
* Update: Editing an existing recipe.
* Delete: Removing a recipe from the system.

**User Profiles**:

* Create: Registering a new user.
* Read: Retrieving user profiles.
* Update: Allowing users to edit their profiles.
* Delete: Permanently deleting a user account.

**Logs**:

* Create: Recording events like successful login attempts.
* Read: Analyzing logs for troubleshooting.
* Update: Log entries are recorded.
* Delete: Allowing manual cleanup of old logs.

1. **What will the user have access to?**

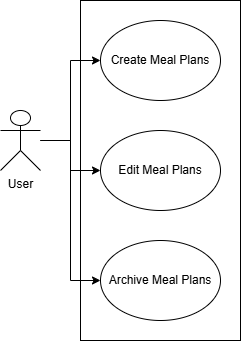
Users will have read-only access to most information. They will have accessibility to change their own profiles and recipes they create.

Users will have limited write access to update their profile information or add new recipes.

1. **What will the admin have access to?**

Admins will have full read and write access to the limited information of the databases, allowing them to view and modify any data within the system.

## Use Case 7: Meal Planner



1. **How will we fill out the meal planner?**

We will pre define and provide options for selecting pre-defined meal templates or allowing free-form inputs by users.

1. **How will we regulate our nutrition facts and guide?**

Have an API for Nutrition information to fetch info for selected ingredients or recipes.

Store this data in a way that allows quick access during meal plan creation.

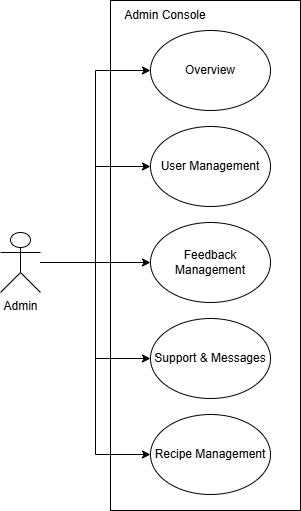
1. **How will we retrieve our information from the database?**

We will use a structured database where each user's meal plan is stored separately, linked to the user's profile.

1. **What will we include for our plans?**

Allow users to mark meal plans as "Active" currently being used, "Inactive" archived for 30 days, or delete them entirely.

## Use Case 8: Admin Page and Controls



1. **What is our admin doing?**

Admins monitors all activity, maintains control over user accounts, and supports where necessary.

1. **Why do we need an admin?**

Admins provide assistance to users facing issues, or in need of support.

1. **What capabilities do we have present?**

Admins have access to all recipes, allowing them to edit or quarantine entries as needed.

1. **What support are we recording?**

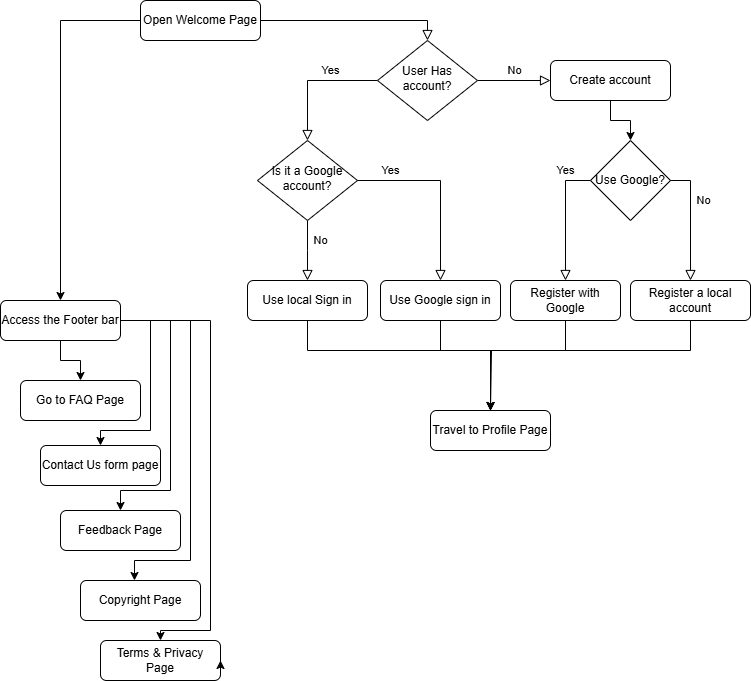
The admin has a view to detailed logs and analytics to understand user behavior, such as popular recipes and frequently visited pages.

1. **How does the admin operate?**

The admin dashboard provides access to system settings, user data, and restricted information.

# Flow of Events Diagrams

## Flow 1: Landing, Registration, Sign-in, Sign-Up, Footer



1. **How does our Landing page work?**

Landing page is intended to greet the users, inform them on what our website is about, it is intended to give a short overview of what is contained in our website, it incentivises registration with us and the footer allows the user to explore further within our project.

1. **How does our registration work?**

Users provide essential information like name, email, and password. Users receive an email confirmation to activate their account and prevent unauthorized sign-ups. else they can register with their google account.

1. **How does our sign-in work?**

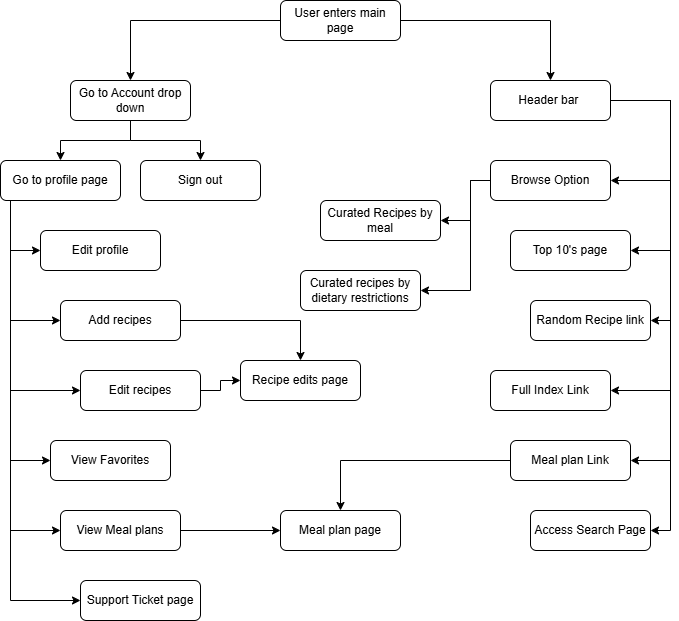
Supports traditional email/password login as well as google OAuth Sign-in

A "Forgot Password" feature allows resetting passwords via email from our contact us page.

1. **What can we access with an account?**

Access to profile details, including username, name, bio, and email. View and manage saved meal plans, created recipes, ratings, and reviews

## Flow 2: Homepage, Header Navigation, Profile



1. After we log-in what capabilities does a user have?

View and manage their profile

Browse saved content

Utilize dietary tools and filters

Taylor searches for meal types, dietary restrictions, or cooking skills.

1. **What pages are accessible now?**

Profile Dashboard

Meal Planner

Recipe Collection

Recipe Randomizer

categorizations of meals by type

categorization of meals by diet restrictions

top 10 recipes by meal selection

a full index of all recipes

a search functionality for recipes with addition refining filters

1. **What is the purpose of this?**

Centralizes access to all user-specific data.

Allows users to plan meals according to their preferences.

recipe options with nutritional details.

Adds to experience by allowing users to contribute feedback.

Offers personalized insights to help users understand their dietary habits.

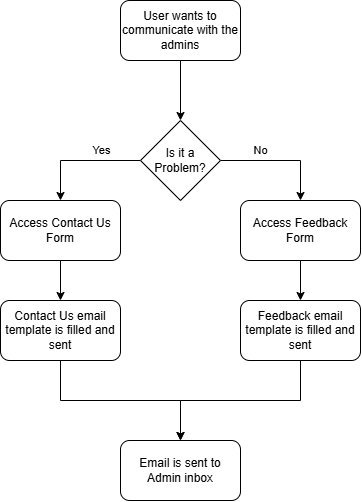
1. **Why is our system better than others?**

Personalization with Tailored meal plans.

Nutritional Tracking with information for each recipe.

easy navigation and customization search options.

## Flow 3: Feedback, Contact, and Support



1. **What is the purpose of feedback?**

To gather user insights and suggestions for improving the project from the end user.

1. **Why do we give this option to the user?**

We are addressing user concerns or enhancing features based on their feedback.

1. **What is our support system for user management?**

We are utilizing a ticketing system where users can submit issues.

1. **What is our system for data management?**

We incorporate noSQL databases and use analytic methods to monitor and enhance user experiences.

1. **How can we CRUD this?**

We enable admins to interact with users' data to manage information.

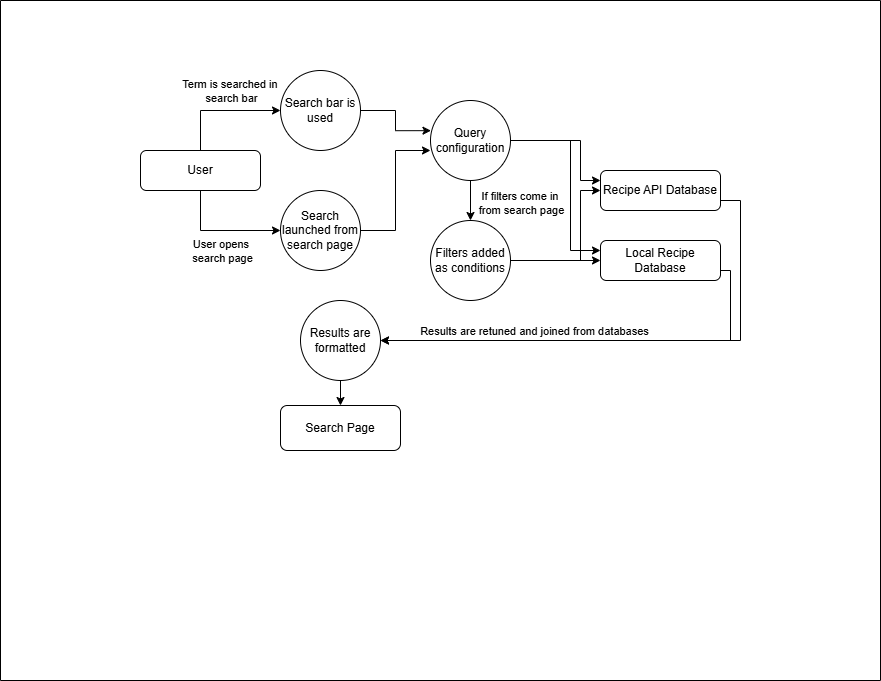
1. **How are we Logging the information and using it?**

We are going to use logs to track user actions and feedback, which admins use for analytics and reporting.

1. **What are we using it for?**

Analytics data is used to determine popular recipes or features the intention is to keep users engaged and informed about what's trending on our website.

## Search Engine



1. **What kind of filtering system will we have for searching for recipes?**

The search will allow users to input keywords, titles, ingredients, dietary preferences or categories.

Users can combine filters for a more refined search

1. **What kind of database will we be interacting with?**

The system will interact with a NoSQL database to store recipe data, user profiles, and other relative information as we develop this project.

We want to create, manage, update and change stored data in our system and the best way of handling flexible information would be to use this method for efficient database operations and data storage.

1. **What is the purpose of this technological stack?**

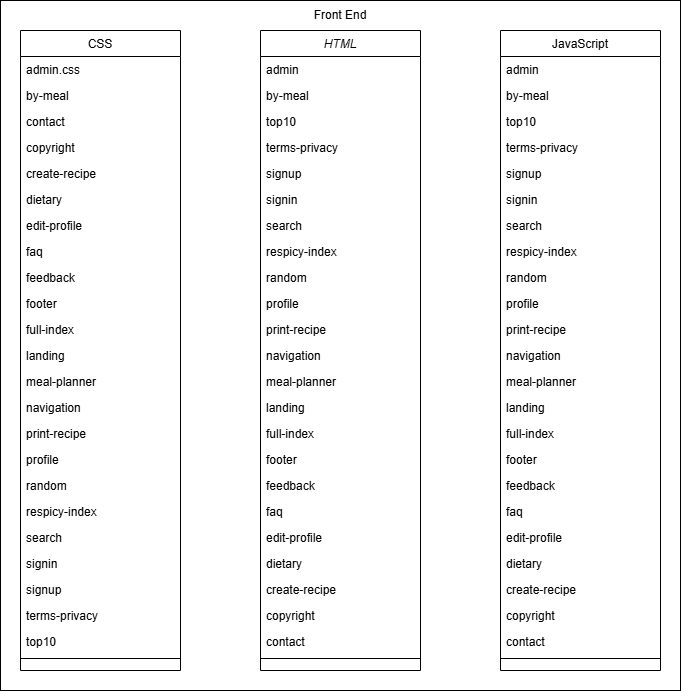
For the frequent updates and changes when handling database operations and storing the correct data.

1. **How will this be implemented throughout the project?**

Back-end will implement API endpoints to interact with the NoSQL database for data retrieval.

## Class Diagram:

### Front End



1. **How is our front end system set up?**

HTML5, CSS3, JavaScript.

1. **Why did we choose this method?**

We are utilizing standard web technologies for creating responsive and interactive user interfaces.

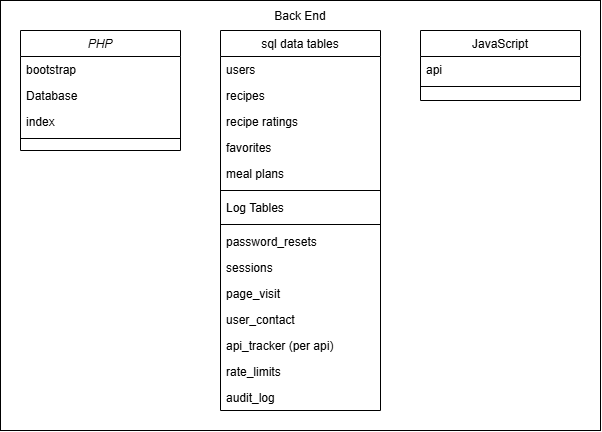
1. **Why is it a web-development system design project?**

To design user-friendly interfaces that communicate with the back end systems quickly and efficiently compared to competitors, encouraging further ease of use to the end user.

1. **How do we communicate with our back end?**

PHP or jQuery to interact with databases like MySQL or MongoDB/Redis.

### Back end overview



1. **How is our backend system set up?**

MongoDB and Redis as our NoSQL databases

1. **Why did we choose this method?**

Flexibility, Scalability, and Real-time Capabilities.

NoSQL databases are more flexible than relational ones.

1. **Why is our database set up in this regard?**

updating recipes or adding new ones is easier, also it allows for ease of use with utilizing the meal planner to the end user.

1. **How does it work for the end user?**

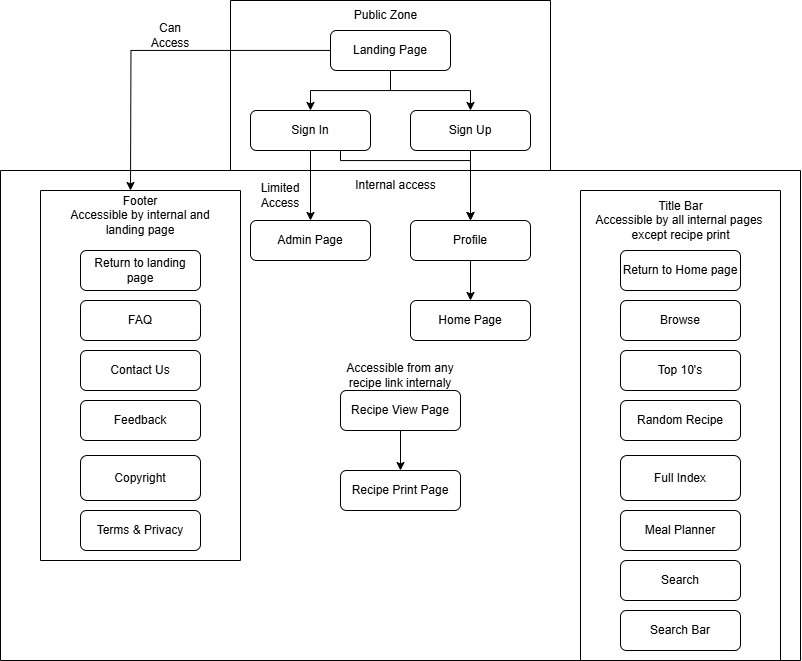
Users can easily navigate through their meal plans and recipes, with efficient searches and filtering.

1. **How do we communicate with our front end?**

Use REST APIs to communicate between our front-end JavaScript application and back-end MongoDB/Redis setup.

AJAX requests in JavaScript will make partial updates, improving user experience and reducing page reloads.

## Site Map: Recspicy



# Appendix

## Key Terms:

1. **What needs to be defined from our Concept document??**

Project purpose and goals - recipe management, meal planning

Audience: home cooks, food enthusiasts

Stack: NoSQL databases, REST APIs, HTML5/CSS3/JavaScript.

Storage methods, Access controls and User authentication.

1. **What key terms are necessary for understanding our system??**

MongoDB or Redis, JavaScript, PHP, or jQuery, HTML5, CSS3,

MySQL, REST APIs, Email/password, OAuth,

AJAX requests for dynamic interactions.

1. **What is needed for understanding our structures??**

**Front-end**: User interface design, responsive layout using HTML, CSS.

**Back-end**: Data storage in NoSQL databases, API endpoints for data retrieval.

**Database layers**: Separate for recipes, user profiles, and logs

**System architecture**: Modular such as recipe DBMS and meal planning

**Data flow**: From front-end to back-end, through API’s.

## 

## 

## References:

Documentation, StackOverflow. “Free HTML5 Book.” Free HTML5 Book, books.goalkicker.com/HTML5Book/. Accessed 26 Feb. 2025.

Documentation, StackOverflow. “Free CSS Book.” Free CSS Book, books.goalkicker.com/CSSBook/. Accessed 26 Feb. 2025.

Documentation, StackOverflow. “Free Javascript Book.” Free JavaScript Programming Book, books.goalkicker.com/JavaScriptBook/. Accessed 26 Feb. 2025.

Documentation, StackOverflow. “Free Jquery Book.” Free jQuery Book, books.goalkicker.com/jQueryBook/. Accessed 26 Feb. 2025.

Documentation, StackOverflow. “Free Mongodb Book.” Free MongoDB Book, books.goalkicker.com/MongoDBBook/. Accessed 26 Feb. 2025.

Documentation, StackOverflow. “Free PHP Programming Book.” Free PHP Programming Book, books.goalkicker.com/PHPBook/. Accessed 26 Feb. 2025.

Documentation, StackOverflow. “Free SQL Database Book.” Free SQL Database Book, books.goalkicker.com/SQLBook/. Accessed 26 Feb. 2025.

Ahmed, Kamran. “Developer Roadmaps.” Roadmap.Sh, 13 Sep. 2024, roadmap.sh/