Recipe Recommendation System

A MINI-PROJECT REPORT

Submitted by

DEEBAK N 220701503

In partial fulfillment of the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI
ENGINEERING COLLEGE
An AUTONOMOUS Institution
Affiliated to ANNA UNIVERSITY, Chennai

RAJALAKSHMI ENGINEERING COLLEGE

AUTONOMOUS, CHENNAI

NOV/DEC, 2024

BONAFIDE CERTIFICATE

Certified that this mini project **"Recipe Recommendation System"** is the bonafide work of **"DEEBAK N (220701503)"** who carried out the project work under my supervision.

SIGNATURE

Mrs. JANANEE V,
Assistant Professor,
Computer Science and Engineering,
Rajalakshmi Engineering College,
Thandalam, Chennai - 602105

Submitted for the End Semester Practical examination to be held on _____

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

I express my sincere thanks to my beloved and honorable chairman **Mr. S. MEGANATHAN** and the chairperson **Dr. M. THANGAM MEGANATHAN** for their timely support and encouragement.

I am greatly indebted to my respected and honourable principal **Dr. S.N. MURUGESAN** for his able support and guidance.

No words of gratitude will suffice for the unquestioning support extended to us by my head of the department, **Dr. P. KUMAR,** and my Academic Head **Dr. R.SABITHA,** for being every supporting force during my project work.

I also extend my sincere and hearty thanks to my internal guide **Mrs. JANANEE V** for her valuable guidance and motivation during the completion of this project.

My sincere thanks to my family members, friends and other staff members of Computer Science and Engineering.

ABSTRACT

The Recipe Recommendation System, built using Flask, MongoDB, and machine learning, provides a tailored platform for users to explore personalized recipe suggestions. Users can sign up, log in, and access a custom dashboard that recommends recipes based on their preferences, cooking history, and ratings. Secure user authentication is managed with bcrypt for password hashing. Admins have exclusive access to manage recipes, including adding, updating, and removing entries, while users can browse, save, and rate recipes to improve the relevance of their recommendations. Session tracking ensures a smooth, personalized experience.

Machine learning models are integrated to enhance recipe recommendations. Using a collaborative filtering algorithm, the system analyzes user behavior and preferences to suggest recipes that align with individual tastes. MongoDB stores user credentials, recipe data, and user interactions, enabling the machine learning model to continually refine recommendations based on new ratings and interactions. The user dashboard updates dynamically to reflect personalized recommendations, allowing users to discover new recipes they’re likely to enjoy. User sessions are managed for a seamless experience across logins, creating an intuitive, secure recipe discovery platform driven by machine learning.

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

1.1 INTRODUCTION

With the rise in demand for personalized cooking content and the shift towards digital platforms, recipe recommendation systems have become increasingly popular. Traditional recipe collections often fail to meet users' unique preferences and dietary restrictions, while a digital, data-driven platform can provide highly personalized suggestions. This Recipe Recommendation System leverages machine learning to analyze user preferences, past interactions, and ratings, enabling users to discover new recipes that suit their tastes. The platform allows users to register, log in, browse recommended recipes, and save their favorites. Additionally, administrators can manage recipe content, ensuring an updated and relevant database for users.

1.2 SCOPE OF WORK

This project involves developing a Recipe Recommendation System that includes user registration, login, personalized recipe suggestions, and administrative functionalities. Designed with Flask and MongoDB, the platform is scalable and efficient, storing and retrieving user and recipe data with ease. Core functionalities include user authentication, recipe recommendation, and session management, while the machine learning component improves recommendation accuracy based on collaborative filtering techniques. Future extensions may include options for dietary preferences, ingredient-based filtering, and interactive cooking tutorials to enhance user engagement.

1.3 AIM AND OBJECTIVES OF THE PROJECT

The primary aim of this project is to build a secure, scalable, and user-centered Recipe Recommendation System that personalizes content for each user. The objectives are as follows:

1. **Develop a secure user authentication system**: Ensure users can register, log in, and log out securely with password encryption.
2. **Provide a dynamic, personalized recipe dashboard**: Allow users to view recommended recipes, rate them, and save their favorites for future access.
3. **Create an administrative dashboard**: Enable administrators to manage recipe content, including adding, updating, and deleting recipes as needed.
4. **Implement session management**: Track user interactions and preferences to maintain personalized experiences, such as welcoming users by name and displaying saved recipes.
5. **Integrate machine learning for recommendation accuracy**: Use collaborative filtering to improve recipe suggestions based on user data and interactions.
6. **Ensure scalability and flexibility**: Build a platform capable of handling a growing user base and recipe database while maintaining performance and relevance.

CHAPTER 2

SYSTEM SPECIFICATIONS

2.1 HARDWARE SPECIFICATIONS

To run the application smoothly, the following hardware is recommended:

- **Processor**: Intel Core i5 or higher (or equivalent AMD Ryzen processors).
- **RAM**: A minimum of 8 GB to handle model inference and web requests efficiently.
- **Storage**: 500 GB hard drive or SSD for storage of required application files and datasets.

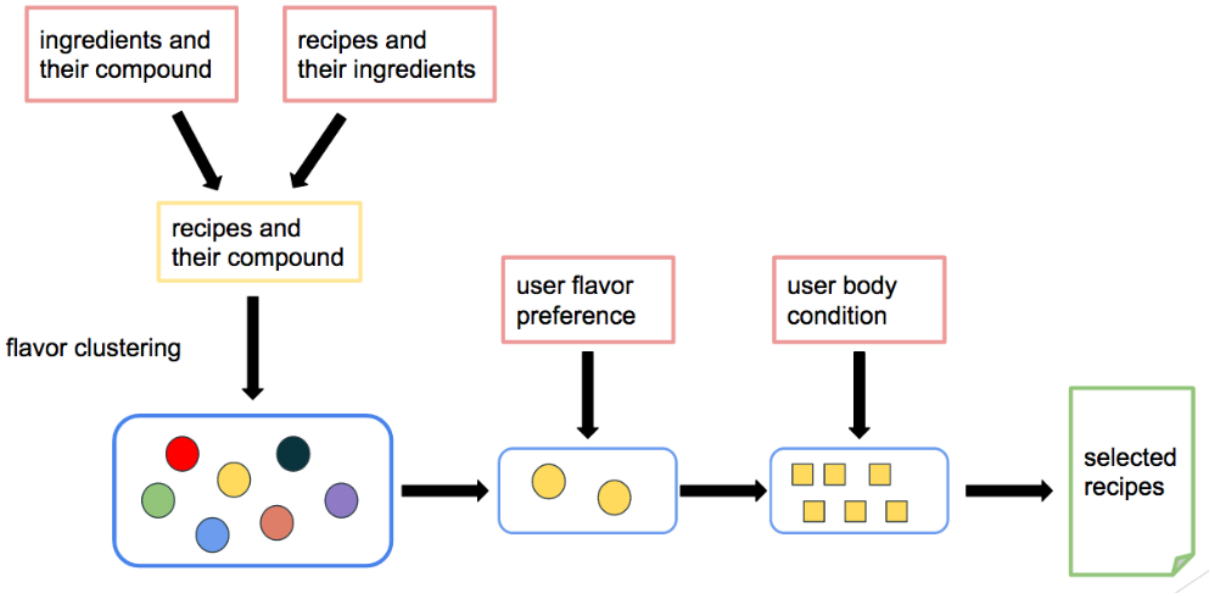
2.2 SOFTWARE SPECIFICATIONS

The project relies on a set of modern software tools and technologies to build, test, and deploy the system:

- **Front-End:** HTML, CSS, Bootstrap, and JavaScript
- **Framework:** Flask
- **Backend:** Python3, MongoDB

CHAPTER 3

ARCHITECTURE DIAGRAM



CHAPTER 4
MODULE DESCRIPTION

4.1. User Authentication Module

This module manages the secure registration, login, and logout processes for users. Passwords are hashed and stored securely using bcrypt to ensure user data protection. This module checks credentials, manages login sessions, and restricts access to certain functionalities based on the user's authentication status. It also handles the account recovery process if a user forgets their password.

4.2. User Profile and Dashboard Module

This module provides each user with a personalized dashboard that displays recipe recommendations based on their past interactions, ratings, and preferences. The dashboard enables users to view, rate, and save recipes, as well as filter recommendations based on dietary restrictions, cuisines, and other preferences. The profile section allows users to manage their saved recipes and account details.

4.3. Recipe Recommendation Engine

Powered by machine learning, this module uses collaborative filtering techniques to generate personalized recipe suggestions. The recommendation engine analyzes user data, such as ratings and previously saved recipes, to predict recipes that align with a user’s tastes. Over time, the model improves by learning from user feedback and interactions, enhancing the accuracy of recommendations.

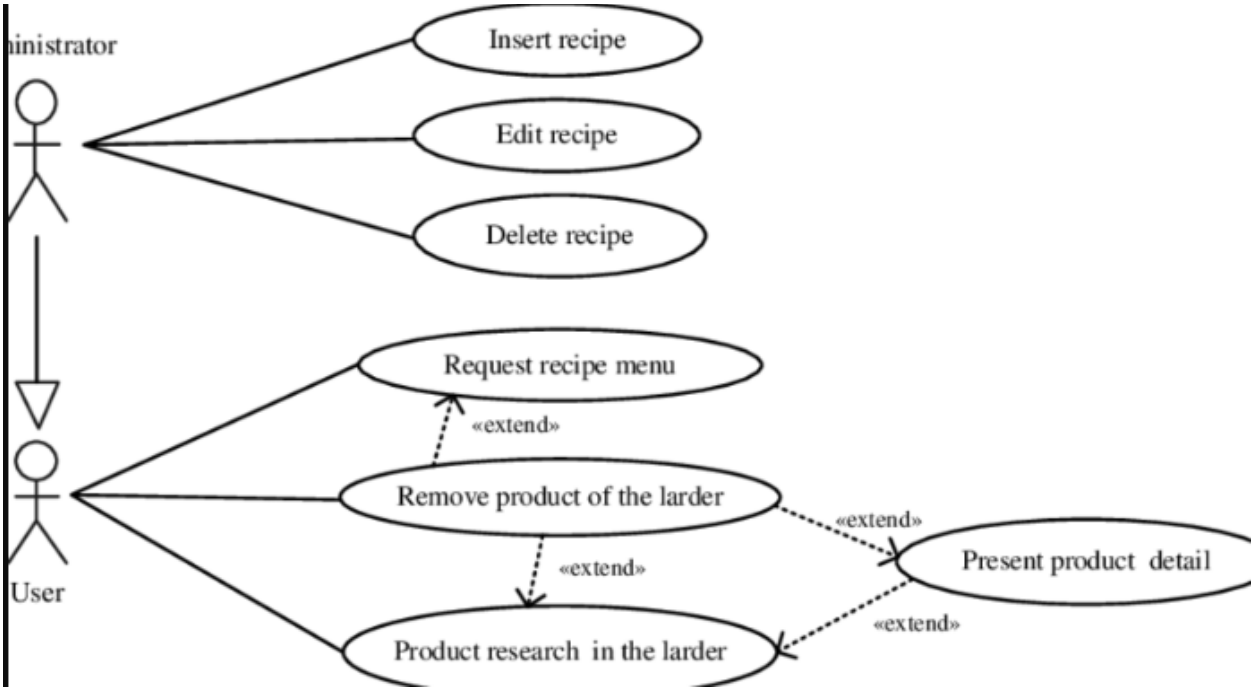
4.4. Recipe Management Module (Admin)

Accessible only to administrators, this module enables the management of recipe content within the system. Admins can add new recipes, update existing ones, and delete outdated or irrelevant recipes from the database. This module ensures that the recipe collection is diverse, up-to-date, and relevant to users. Admins can also categorize recipes by cuisine, meal type, or dietary preferences for easier user filtering.

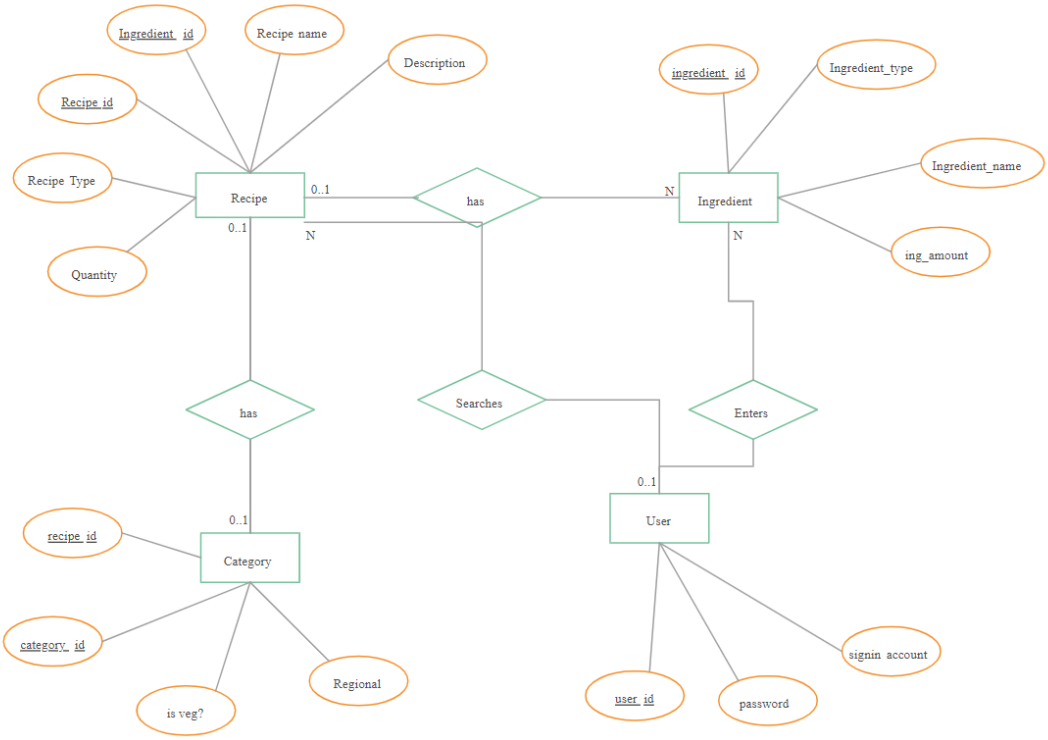
CHAPTER 5

SYSTEM DESIGN

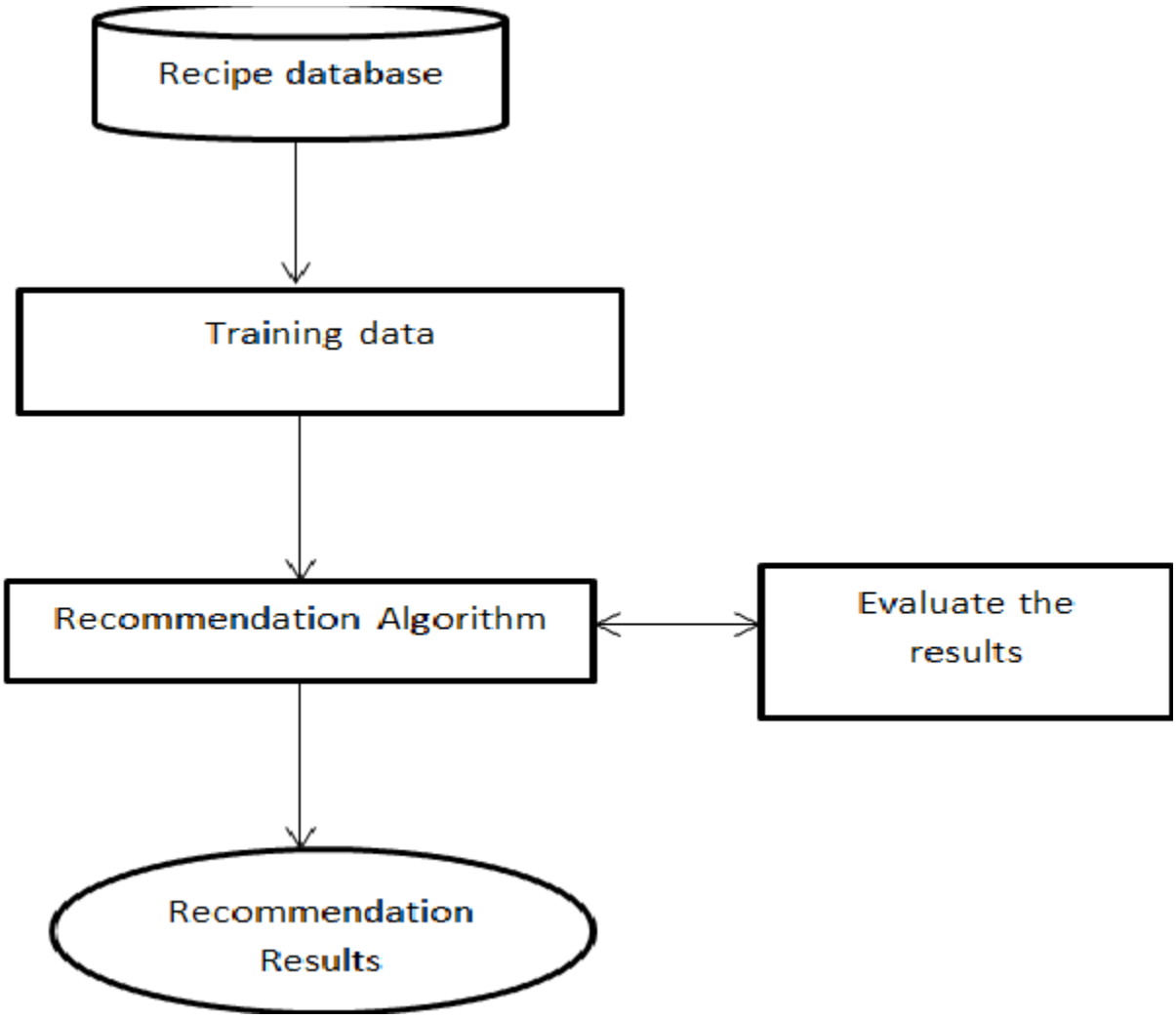
5.1 USE CASE DIAGRAM



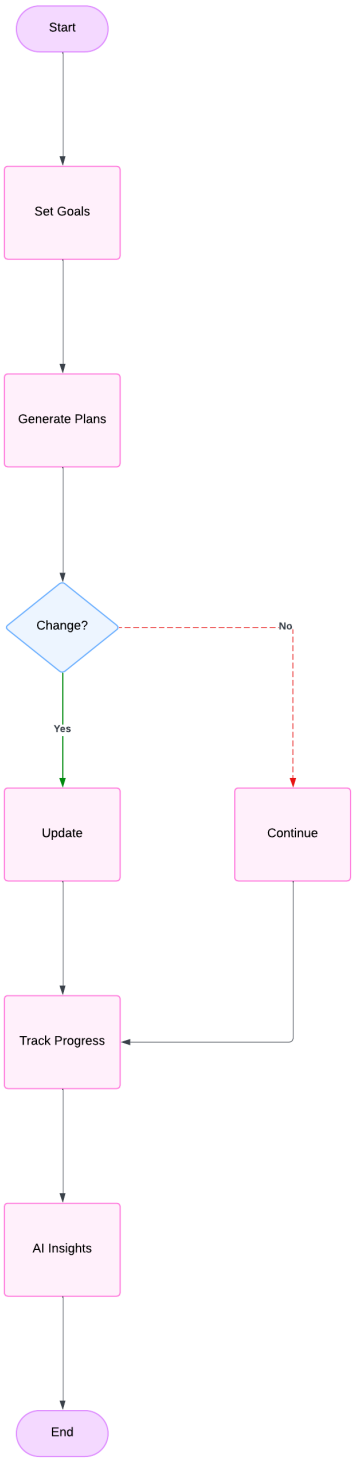
5.2 ER DIAGRAM



5.3 DATA FLOW DIAGRAM



5.3 ACTIVITY DIAGRAM



CHAPTER 6

SAMPLE CODING

```
<!DOCTYPE html>
<html lang="en">
<head>
   <title>WELLIFY</title>
   <link rel="stylesheet" href="basic.css">
</head>
<body>

   <div class="main">
      <div class="navbar">
         <div class="icon">
            <h2 class="logo">Wellify</h2>
         </div>

         <div class="menu">
            <ul>
               <li><a href="#">HOME</a></li>
               <li><a href="#" onclick="showAboutIdea()">IDEA</a></li>
               <li><a href="#" onclick="showAboutAlert()">ABOUT</a></li>
               <li><a href="#"
onclick="showContactAlert()">CONTACT</a></li>
            </ul>
         </div>

         <!-- Calorie Finder and Welcome Username -->
         <div class="action-items">
            <div class="calorie-finder">
               <button onclick="redirectToCalorieCalculator()"
class="btn">Calorie Finder</button>
            </div>
```

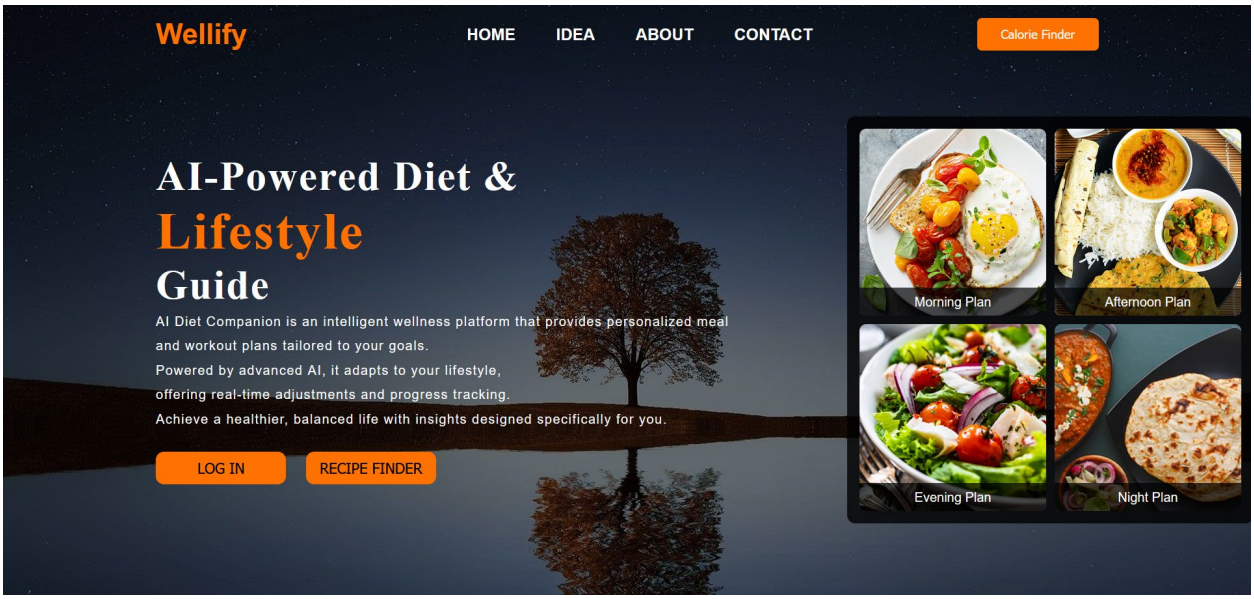
```
            <div id="user-display" class="user-display"></div> <!-- User
display area -->
        </div>
    </div>

    <div class="content">
        <h1>AI-Powered Diet &<br><span>Lifestyle</span>
<br>Guide</h1>
        <p class="par">AI Diet Companion is an intelligent wellness
platform that provides personalized meal <br>and workout plans tailored to
your goals.<br> Powered by advanced AI, it adapts to your lifestyle,<br>
offering real-time adjustments and progress tracking.
            <br> Achieve a healthier, balanced life with insights designed
specifically for you.</p>

        <button class="cn"><a href="login.html">LOG IN</a></button>
        <button class="cn"><a href="recipe.html">RECIPE
FINDER</a></button> <!-- Updated Recipe Finder button -->

        <!-- 2x2 Plan Grid -->
        <div class="plan-grid">
            <!-- Morning Plan box with link to morning_plan.html -->
            <div class="plan-box" onclick="redirectToMorningPlan()">
                <img src="morning.jpg" alt="Morning Plan">
                <div class="plan-box-footer">Morning Plan</div>
            </div>
            <div class="plan-box">
                <img src="lunch.jpg" alt="Afternoon Plan">
                <div class="plan-box-footer">Afternoon Plan</div>
            </div>
            <div class="plan-box">
                <img src="eve.jpg" alt="Evening Plan">
```

```
            <div class="plan-box-footer">Evening Plan</div>
          </div>
          <div class="plan-box">
            <img src="dinner.jpg" alt="Night Plan">
            <div class="plan-box-footer">Night Plan</div>
          </div>
        </div>
      </div>
    </div>

    <script src="https://unpkg.com/ionicons@5.4.0/dist/ionicons.js"></script>

    <script>
      // Redirects to calorie_calculator.html
      function redirectToCalorieCalculator() {
        window.location.href = "calorie_calculator.html"; // Redirect to the
calorie calculator page
      }

      // Redirects to morning_plan.html
      function redirectToMorningPlan() {
        window.location.href = "morning_plan.html"; // Redirect to the
morning plan page
      }

      // Display the username if available in sessionStorage
      window.onload = function() {
        const username = sessionStorage.getItem('userName');
        if (username) {
          document.getElementById('user-display').innerText = `Welcome,
${username}`;
        }
```

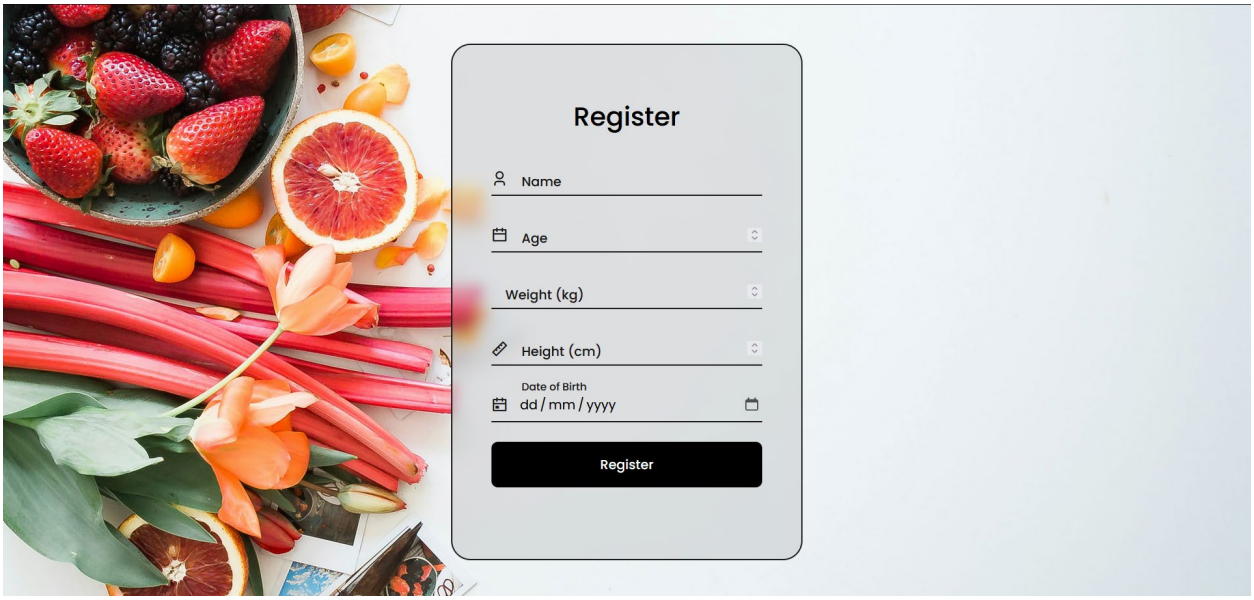
```
        };

        function showContactAlert() {
            alert("CONTACT: MOHNISH - 220701171.");
        }

        function showAboutIdea() {
            confirm("Wellify is an AI-powered platform that offers personalized diet and workout plans. Our goal is to help you achieve a healthier, balanced lifestyle through tailored insights.");
        }

        function showAboutAlert() {
            confirm("WELLIFY - YOUR AI COMPANION\nHelping you achieve a balanced and healthier lifestyle through tailored AI insights.");
        }
    </script>
</body>
</html>

# models.py
from flask_sqlalchemy import SQLAlchemy
from flask_login import UserMixin

db = SQLAlchemy()

class User(db.Model, UserMixin):
    id = db.Column(db.Integer, primary_key=True)
    username = db.Column(db.String(150), unique=True, nullable=False)
    password = db.Column(db.String(150), nullable=False)
    name = db.Column(db.String(150))
    age = db.Column(db.Integer)
```

```
    weight = db.Column(db.Float)
    height = db.Column(db.Float)
    profile_complete = db.Column(db.Boolean, default=False)
    routine = db.relationship('Routine', backref='user', lazy=True)

class Routine(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    time_of_day = db.Column(db.String(50))  # e.g., "morning", "afternoon"
    workout_plan = db.Column(db.String(250))
    meal_plan = db.Column(db.String(250))
    user_id = db.Column(db.Integer, db.ForeignKey('user.id'), nullable=False)
```

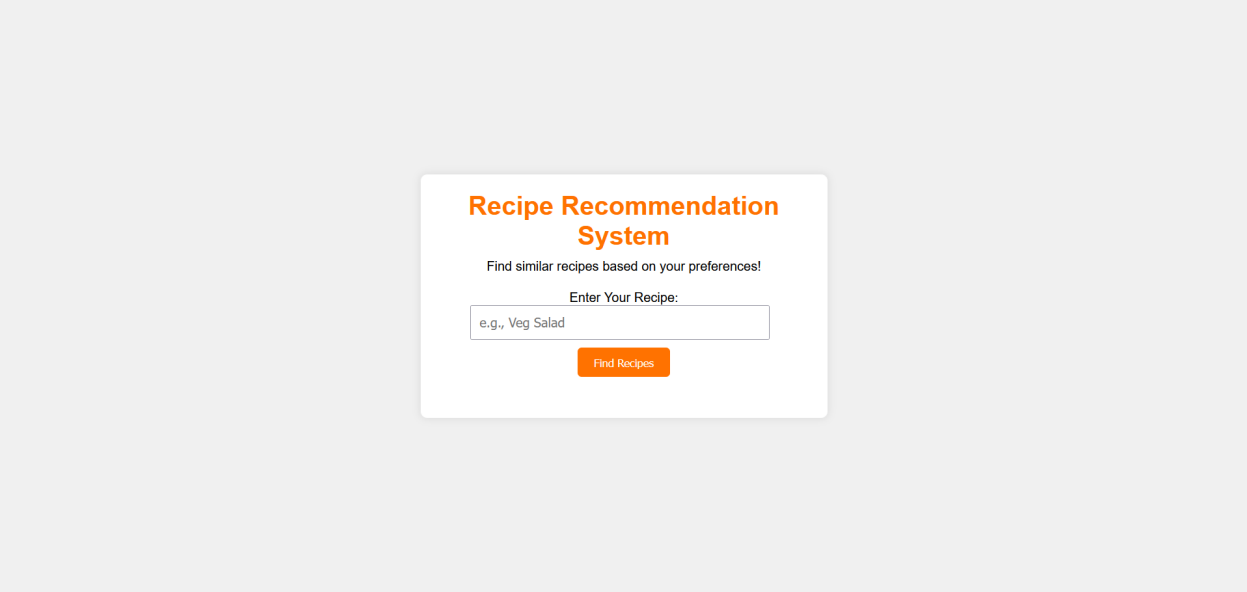
Chapter 7
SCREENSHOTS



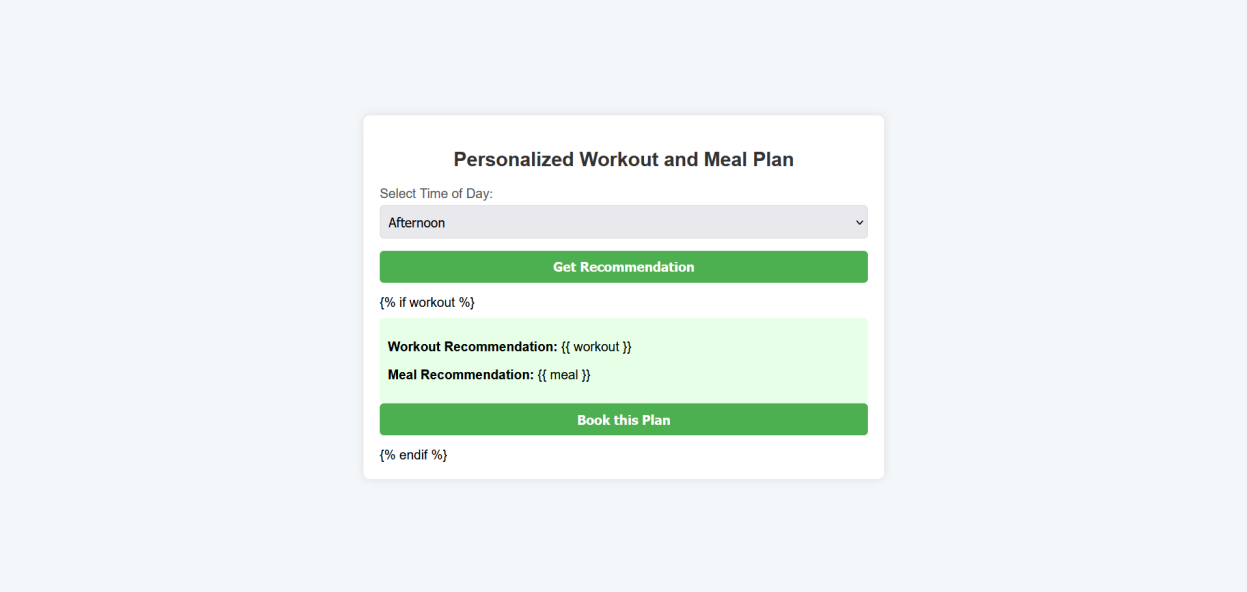
7.1 Landing Page



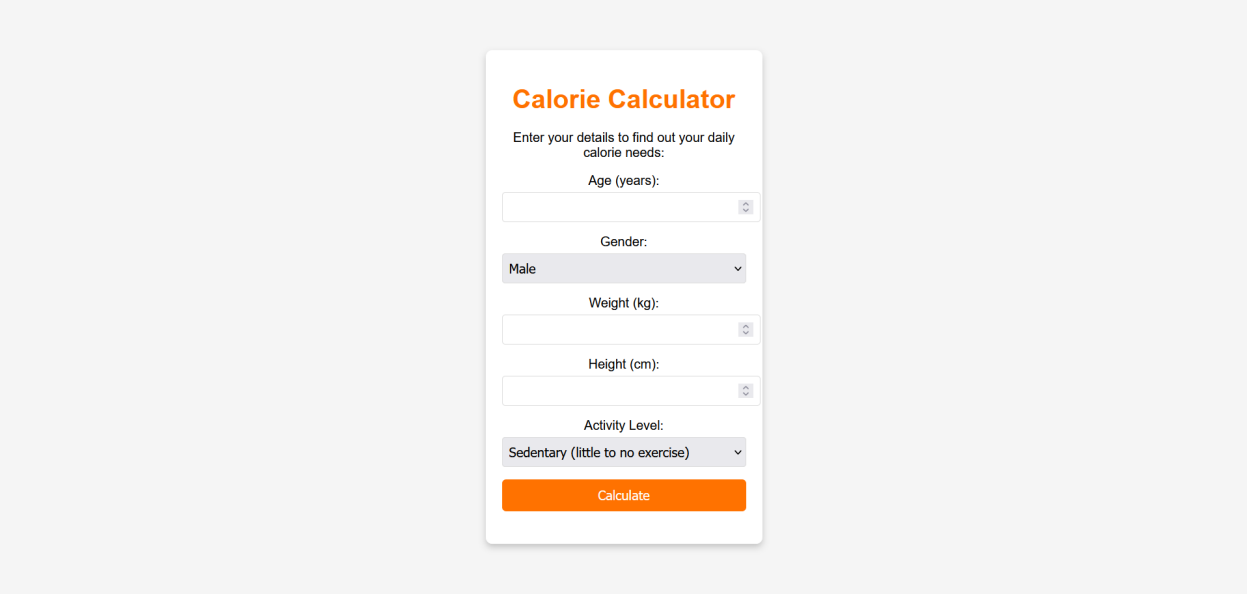
7.2 Registration Page



7.3 RECIPE FINDER



7.4 PLAN BOOKING PAGE



7.5 CALORIE CALCULATOR PAGE

CHAPTER 8
CONCLUSION

The Recipe Management System offers a robust, user-centric platform that enables users to discover, save, and enjoy personalized recipe recommendations tailored to their tastes. By integrating essential features such as secure user authentication, a dynamic recommendation engine powered by machine learning, and an easy-to-navigate recipe catalog, the system enhances the cooking experience for users of all skill levels. The administrative capabilities allow for effective management of recipe content, ensuring that the platform remains relevant, up-to-date, and diverse.

This modular, scalable architecture allows for continuous enhancements, such as ingredient-based filtering, dietary preferences, and expanded user analytics. Ultimately, the Recipe Management System is a powerful digital resource that adapts to the evolving needs of its users, supporting a personalized, enjoyable, and engaging recipe discovery experience. Through its innovative combination of machine learning and user-centered design, the platform is well-positioned to become a valuable tool for culinary exploration and meal planning.

REFERENCES

1. Agrawal, R., & Srikant, R. (1994). "Fast algorithms for mining association rules." Proceedings of the 20th International Conference on Very Large Data Bases (VLDB).

2. Koren, Y., Bell, R., & Volinsky, C. (2009). "Matrix factorization techniques for recommender systems." Computer, 42(8), 30-37.

3. Cheng, C., Yang, H., & Lyu, M. R. (2014). "Where you like to eat: Exploiting multiple information sources in location-based recommender systems." Proceedings of the 7th ACM Conference on Recommender Systems (RecSys).

4. Chambers, E., & Smith, E. A. (1998). "The Sensory Evaluation of Foods: Traditional and Contemporary Approaches." Critical Reviews in Food Science and Nutrition, 38(4), 345-358.

5. Pasquale, F. (2015). The Black Box Society: The Secret Algorithms That Control Money and Information. Harvard University Press.

6. Bengfort, B., Bilbro, R., & Ojeda, T. (2018). Applied Text Analysis with Python: Enabling Language-Aware Data Products with Machine Learning. O'Reilly Media.

7. Russell, S. J., & Norvig, P. (2016). Artificial Intelligence: A Modern Approach (3rd ed.). Pearson Education.

8. Ricci, F., Rokach, L., & Shapira, B. (2011). Introduction to Recommender Systems Handbook. Springer.

9. MongoDB, Inc. (n.d.). MongoDB Documentation. Retrieved from https://www.mongodb.com/docs/.