A

Mini Project Report

on

SmartPlate: ML Based Mealkit Delivery System

Submitted in partial fulfillment of the requirements for the degree

Third Year Engineering – Information Technology

by

Akshata Nalavade **21104003**

Manjiri Gole **21104006**

Shweta Bhutada **21104007**

Under the guidance of

Ms. Geetanjali Kalme

DEPARTMENT OF INFORMATION TECHNOLOGY

A.P. SHAH INSTITUTE OF TECHNOLOGY

G.B. Road, Kasarvadavali, Thane (W)-400615

UNIVERSITY OF MUMBAI

Academic year: 2023-24

CERTIFICATE

This to certify that the Mini Project report on **SmartPlate: ML Based Meal delivery System** has been submitted by Akshata Nalavade (21104003), Manjiri Gole (21104006) and Shweta Bhutada (21104007) who are bonafide students of A. P. Shah Institute of Technology, Thane as a partial fulfillment of the requirement for the degree in **Information Technology**, during the academic year **2023-2024** in the satisfactory manner as per the curriculum laid down by University of Mumbai.

Ms. Geetanjali Kalme
Guide

Dr. Kiran Deshpande
HOD, Information Technology

Dr. Uttam D. Kolekar
Principal

External Examiner:
1.

Internal Examiner:
1.

Place: A. P. Shah Institute of Technology, Thane
Date:

ACKNOWLEDGEMENT

This project would not have come to fruition without the invaluable help of our guide **Geetanjali Kalme**. Expressing gratitude towards our HoD, **Dr. Kiran Deshpande**, and the Department of Information Technology for providing us with the opportunity as well as the support required to pursue this project. We would also like to thank our project coordinator **Ms. Sonal Jain** who gave us her valuable suggestions and ideas when we were in need of them. We would also like to thank our peers for their helpful suggestions.

1. Shweta Bhutada 21104007
2. Manjiri Gole 21104006
3. Akshata Nalavade 21104003

TABLE OF CONTENTS

Abstract

1. Introduction…. 1

1.1. Purpose 1

1.2. Problem Statement…. 2

1.3. Objectives. 2

1.4. Scope 3

2. Literature Review…. 4

3. Proposed System…. 6

3.1. Features and Functionality 6

4. Requirements Analysis 7

5. Project Design 9

5.1. Use Case diagram… 9

5.2. DFD (Data Flow Diagram) 10

5.3. System Architecture 12

5.4. Implementation… 13

6. Technical Specification 16

7. Project Scheduling 18

8. Results…. 19

9. Conclusion 24

10. Future Scope 25

References

ABSTRACT

The website SmartPlate is a dynamic online platform that seeks to reinvent how we discover, produce, and enjoy food. Our website's primary goals are to improve culinary experiences and make meal planning less difficult, even for those who find it difficult. It serves a wide range of customers, including health-conscious people committed to nutritious eating, enthusiastic home cooks thirsty for originality, and busy professionals in need of convenience.

Our website is more than simply a recipe book; it's a cooking partner and an interactive environment where cooking is turned into an adventure. Users can access a plethora of categorized recipe collections, become part of a lively community of like-minded food enthusiasts, and delve into a treasure trove of culinary expertise here. It's a place to explore flavors, try out new recipes, and celebrate culinary achievements with pride.

We've packed our website full of features that put the user in a position of control so that this experience is truly customized to users need. You can anticipate easy serving size adjustments, customized recipe recommendations based on your preferences, and the extra convenience of ordering ingredients for recipes you can't wait to test.

Chapter 1

Introduction

SmartPlate is user-friendly website for an online recipe box ordering service that enables customers to easily browse, customize, and order recipe boxes with pre-portioned ingredients and step-by-step cooking instructions. It is a recipe platform that provides users with both recipes and the corresponding ingredient lists for those recipes. The website aims to make cooking at home easier and more enjoyable by offering a comprehensive solution for meal planning and preparation.

The common challenges that many individuals face when it comes to meal planning and cooking. Busy schedules, limited resources, and repetitive cooking routines can hinder their ability to enjoy the benefits of cooking at home, such as variety in their meals and the satisfaction of preparing their own food. Solutions such as meal kit delivery services aim to address these challenges by providing convenience, diverse recipes, and pre-portioned ingredients to make home cooking more accessible and enjoyable. The website caters to the needs of busy individuals and families, aiming to streamline the meal planning process, promote healthier eating habits, and provide a delightful cooking experience at home.

1.1 Purpose:

i. Individuals with hectic schedules who seek convenient meal solutions without compromising on quality. The website serves as a comprehensive meal solution provider. It offers users a convenient and time-saving way to plan meals.

ii. Users can access a wide variety of recipes along with pre-portioned ingredients, eliminating the need of extensive meal planning and grocery shopping.

iii. It achieves this by offering detailed, step-by-step recipes with instructions and accompanying visuals. Users can learn new cooking techniques and explore diverse cuisines. This multifaceted approach aims to make home cooking enjoyable, educational and socially engaging.

1.2 Problem Statement:

i. In today's hectic world, juggling work, family, and personal commitments leave little time for meal planning and cooking. This leads to unhealthy eating habits and reliance on expensive convenience foods. The solution lies in streamlining meal prep without sacrificing nutrition.

ii. Additionally, the monotony of cooking the same dishes can diminish the joy of cooking. Innovative solutions are needed to diversify culinary options while catering to busy lifestyles.

Solution proposed:

1. Creating a website that caters to busy individuals and families, streamlining meal planning with a wide variety of recipes for users to choose from.
2. Delivering fresh pre-portioned ingredients and recipes to your door, saving time.

1.3 Objectives:

i. To **offer a convenient solution** for individuals who struggle with meal planning and offering ready-to-cook recipe boxes with pre-portioned ingredients and clear cooking instructions, providing a time-saving solution for busy lifestyles.

ii. To implement a user-friendly interface for **easy navigation and recipe selection.**

iii. To provide a variety of recipes so users can explore **various cuisines and recommend recipes** to users.

iv. To integrate a **dynamic ingredient selection** that adjusts ingredient quantities based on user preferences, and substitute ingredients according to their dietary restrictions, minimizing waste and optimizing ingredient usage

1.4 Scope:

i. Can be the platform that **suggests and recommends recipes** to users based on feedback from other users. By analyzing the preferences and cooking experiences of other users, the platform can suggest recipes that align with individual tastes and dietary preferences.

ii. Can be the platform that customizes recipes to their **dietary requirements and ingredient preferences**, providing a tailored cooking experience. By enabling customization, users can adapt recipes to suit their individual tastes and nutritional requirements, enhancing satisfaction.

iii. Can be the platform incorporates a **chatbot feature** to address basic user queries effectively. Users can interact with the chatbot to receive instant assistance and guidance, enhancing their overall experience with the platform.

Chapter 2

Literature Survey

Sr. No	Title of Paper	Year of Publication	Technology used	Drawbacks
1	NLP and ML approaches for food categorization and nutrition quality prediction compared with traditional methods (Guanlan Hu, Mavra Ahmed, Mary R.)	2023	Natural Language Processing (NLP), Word Embeddings, Machine Learning (ML)	Inadequate feature selection can harm model performance. Deploying complex ML models often demands substantial computational resources.
2	Food Demand Prediction using Statistical and Machine Learning Models (Sasikumar Jayapal)	2022	multiple linear regression, lasso, ridge, Bayesian ridge regression, SVR, decision tree, random forest, and Gradient boosting regression models such as Gradient Boosting, XG-Boosting,	Inaccurate or incomplete data can lead to poor prediction performance. May suffer from overfitting, where the model captures noise in the training data instead of

				underlying patterns
3	Raspberry Pi based Nutritional Health Kits for Diabetic Patients (Preethi Mannepally, Krishna Chaithanya, Bhaskar Bhanavath)	2022	Raspberry Pi, Sensors, Web Development, Data Analysis Tools	While Raspberry Pi is generally considered affordable, the cost of additional components. Personal health information, may raise concerns about privacy and security.
4	Calorie Estimation of Food and Beverages using Deep Learning (IEEE)	2021	Convolutional NeuralNetworks (CNN)	Face difficulties in accurately estimating the portion sizes, Limited dataset,
5	Content-Based Recommendation Systems	2020	Recommendation Algorithms, CDNs	Implementation Complexity, recommendation algorithms may exhibit biases that impact the fairness and diversity of content recommendation.

Chapter 3

Proposed System

1.1 Features and Functionality

1) **Feature 1:** Categorized Recipes with Detailed Pages, Images, and Nutritional Information

Each recipe would have its own detailed page, including a list of ingredients, step-by-step cooking instructions, and high-quality images showcasing the finished dish. Additionally, nutritional information such as calorie count, macronutrient breakdown, and potential allergens would be provided to help users make informed choices about their meals.

2) **Feature 2:** User Login for Personalized Experience

The login system would also allow for a seamless experience, ensuring that users can access their recipes and preferences.

3) **Feature 3:** Recommendation System

The recommendation system provides a variety of dishes for the users. It recommends on the basis of each users input on what they should try out. With the help of this recommendation system we are able to create a much enhanced user experience for those users who want to try out new recipes.

4) **Feature 4:** Chatbot

The implementation of a chatbot helps us in answering frequently asked questions or if the users have some query or they want to give some feedback.

5) **Feature 5:** Ratings and reviews

This feature enables users to give ratings and reviews for the recipe they have tried and encourages other users to try out the recipes

Chapter 4

Requirement Analysis

4.1. Requirement Gathering

4.1.1 User Requirements:

1. Users need fast, simple recipes that fit their busy schedules, requiring minimal prep time and complexity for efficient meal preparation.

2. Users may want convenient ingredient sourcing to save time and streamline shopping, ensuring they have what they need when they need it.

3. Users are provided with a variety of diverse recipes to keep cooking exciting and broaden cooking horizons.

4. Users require a search function that enables them to find recipes based on ingredient availability.

4.2 Software Requirement Specification:

4.2.1 Functional Requirements:

1. **Quick Recipes:** A user's requirement is quick and easy-to-follow recipes that align with their busy schedule. They may need recipes that don't demand extensive prep time or complicated techniques, allowing them to prepare meals efficiently.

2. **Ingredient Ordering:** User may seek convenience in ingredient procurement. Integration with local grocery stores or suppliers for ingredient ordering and delivery is crucial. This feature would save them valuable time and streamline the shopping process, ensuring they have the necessary ingredients when they need them.

3. **Ingredient Information:** A user requires access to information about the ingredients like the number of calories or the nutrients provided from the ingredients.

4. **Diverse Recipes:** To keep their cooking exciting and varied, we seek a wide range of diverse and international recipes. We want to explore different cuisines and cooking styles, which can broaden their culinary horizons.

5. **Efficient Search:** A user needs the ability to search for recipes based on ingredient availability, preparation time, and dietary preferences. These features help them discover recipes that align with their specific needs and interests.

4.2.2 Non-Functional Requirements:

1. **Performance:** The system should consistently deliver fast response times and efficient processing.
2. **Data Backup**: Regular automated backups of user data should be implemented to prevent loss.
3. **Usability:** The system interface should be intuitive and user-friendly for easy navigation and interaction.

Chapter 5

Project Design

5.1 Use Case diagram

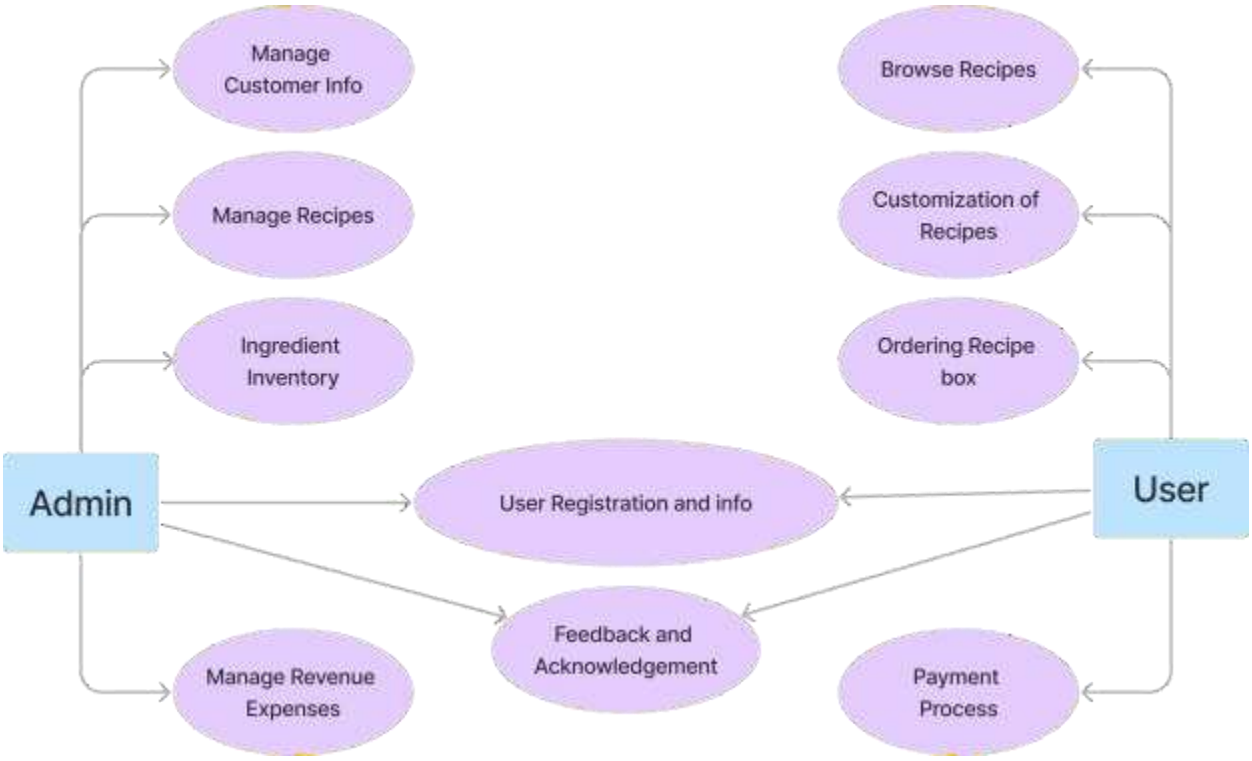


Figure 5.1.1 Use Case Diagram

Figure 5.1.1 is a general use case diagram for the SmartPlate defining two main actors Admin and User. As an admin, you have access to a wide range of use cases, including managing customer information and overseeing delivery services. As an admin, you can manage user information, update recipes, and keep track of ingredient records, including stock levels. You are also responsible for managing overall expenses and tracking the delivery service by maintaining delivery records. On the user's side, user can browse what they want and customize their orders before purchase. Furthermore, we highly value their feedback and encourage them to share their thoughts on how the recipe worked for them. Here also we are trying to implement the tracking system to check the status of order.

5.2 DFD(Data Flow Diagram)

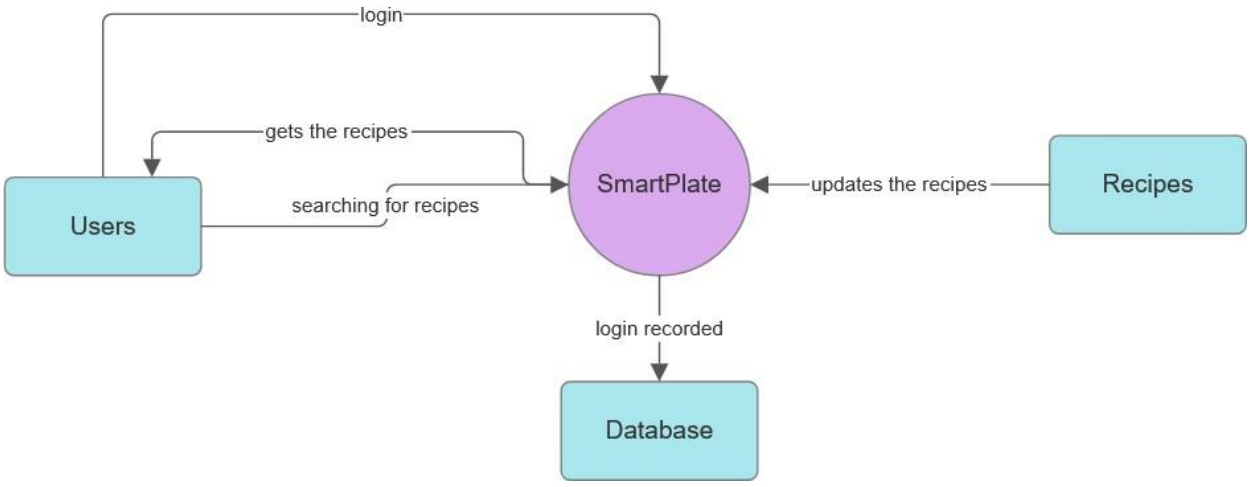


Fig 5.2.1 Level – 0 DFD

Fig 5.2.1, we have users who interact with the "Smart Plate" website. Users can perform several key actions, such as accessing and obtaining recipes, searching for specific recipes, and logging into their accounts. The system maintains a record of user logins, ensuring a secure and personalized experience. This diagram outlines the fundamental flow of activities within the system, where users engage with recipes and the platform.

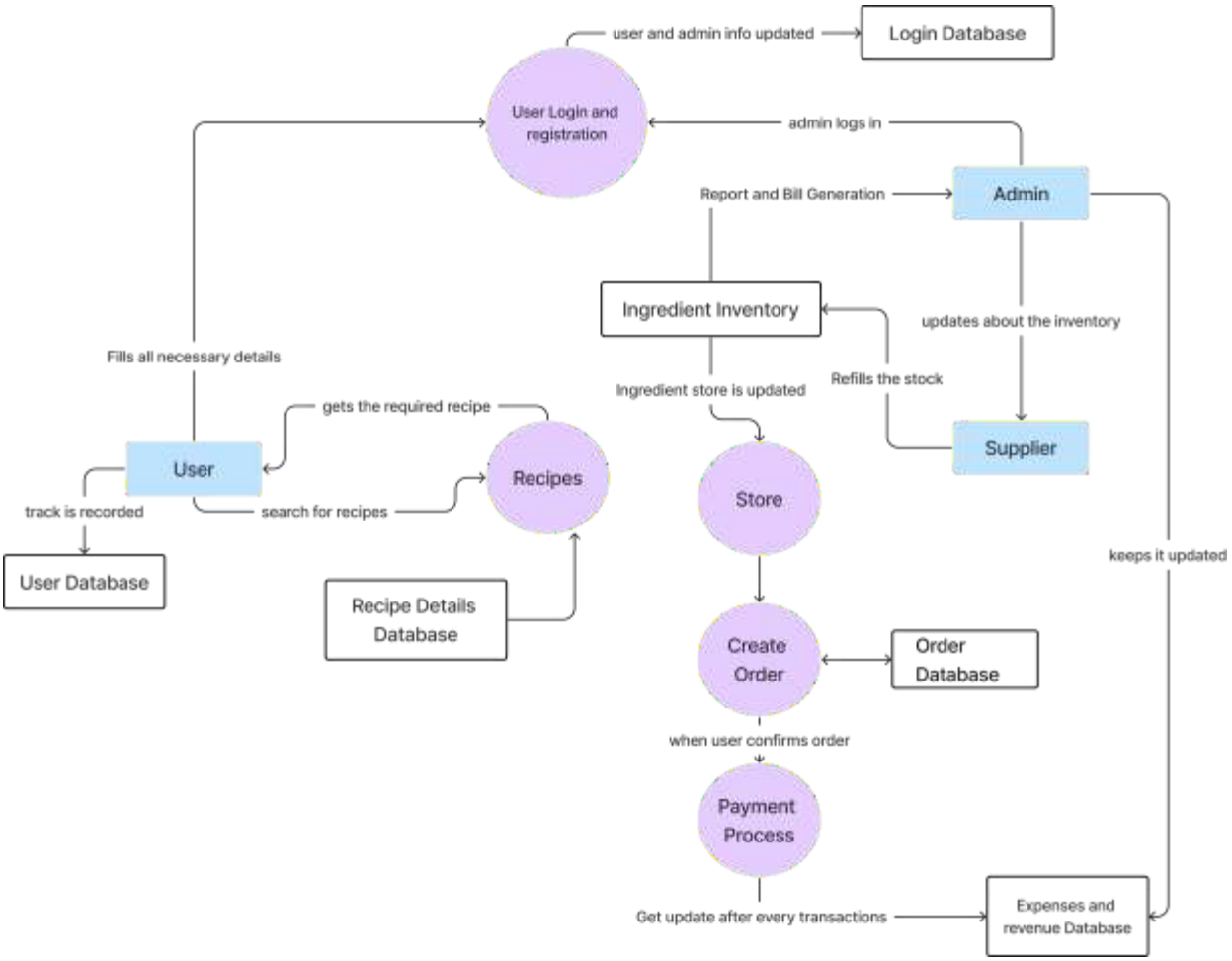


Fig 5.2.2 Level – 1 DFD

Figure 5.2.2 is a DFD diagram which gives us more detail information about the process. A user may start with filling their personal details and complete the registration process. All this information is gets stored into login database. As the user we can browse or search different recipes, these recipes are retrieved from the recipe database, allowing the user to access and view them. The admin is responsible for receiving reports and bills for the ingredients supplied to the store. Users then create the order which is recorded into order database. As the user confirms the order, they are headed towards the payment process. The receipt then gets generated which us forwarded to user as well the admin. Transactions are updated into the expenses database which is monitored by the admin.

5.3 System Architecture

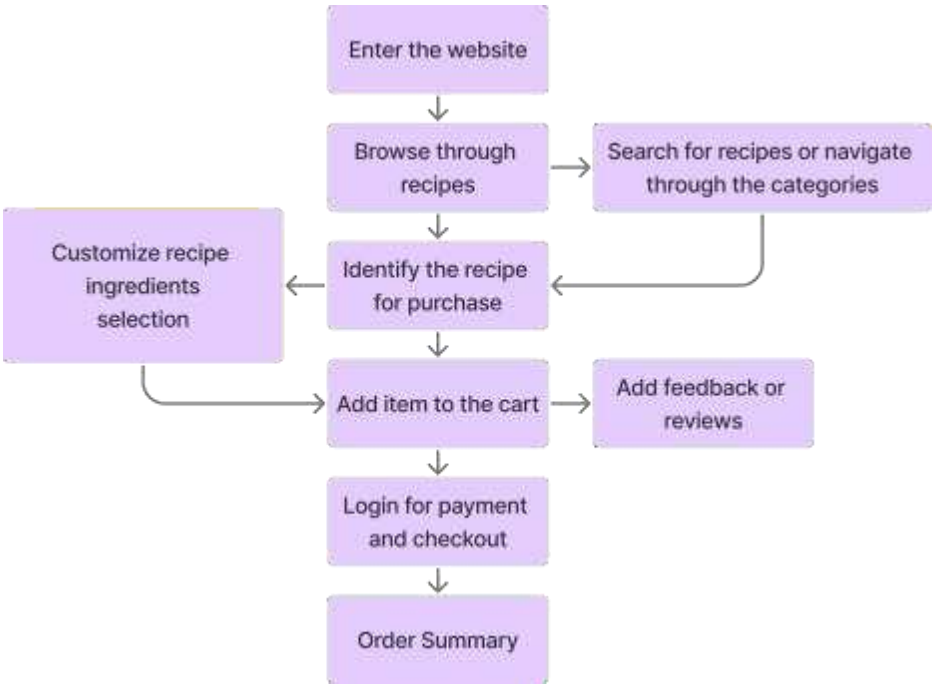


Fig 5.3.1 System Architecture

Figure 5.3.1 is the system architecture of the website which offers a seamless and intuitive user experience for meal planning and recipe selection. Starting from the homepage, users can easily access the recipe list page, where a variety of culinary options await. They can then explore recommended recipes, tailored to their tastes through the website's recommendation engine. If users prefer to browse by category, they can simply use the menu 14 button to navigate and find recipes of their choice. For those with specific preferences, a search feature is available for easy recipe discovery. Upon identifying a recipe they'd like to prepare, users can add the required ingredients to their cart for a convenient shopping experience. Should users wish to explore more recipes, the website offers a continuous selection. After finalizing their recipe choices, the seamless checkout process begins, allowing users to review their orders and make any necessary adjustments. If not already logged in, users can sign up or log in at this stage. Once confirmed, users receive an order confirmation, ensuring they have successfully completed their culinary journey on the website.

5.4 Implementation

```
38  app.use(express.static('images'));
39  app.get('/', checkAuthenticated, (req, res) => {
40    res.render('/views/login.ejs', { name: req.user.name })
41  })
42
43  app.get('/login', checkNotAuthenticated, (req, res) => {
44    res.render('login.ejs')
45  })
46
47  app.post('/login', checkNotAuthenticated, passport.authenticate('local', {
48    successRedirect: '/index',
49    failureRedirect: '/login',
50    failureFlash: true
51  }))
52
53  app.get('/register', checkNotAuthenticated, (req, res) => {
54    res.render('register.ejs')
55  })
56
57  app.post('/register', checkNotAuthenticated, async (req, res) => {
58    try {
59      const hashedPassword = await bcrypt.hash(req.body.password, 10)
60      users.push({
61        id: Date.now().toString(),
62        name: req.body.name,
63        email: req.body.email,
64        password: hashedPassword
65      })
66      res.redirect('/login')
67    } catch {
68      res.redirect('/register')
69    }
```

Fig. 5.4.1 Server.js

Fig. 5.4.1 is a code snippet from our server.js file. This snippet includes the code for authentication of user login. An authenticated user is a user who has successfully registered for the website. This code snippet checks if a user is authenticated or no if they are they will be redirected to the login page if they are registered and if they are not authenticated they are redirected to the register page.

```
function sendMessage() {
    const userMessage = messageInput.value;
    const chatBox = document.createElement("div");
    chatBox.className = "chat-message";

    // Display user message
    const userDiv = document.createElement("div");
    userDiv.className = "user-message";
    userDiv.innerText = "You: " + userMessage;
    chatBox.appendChild(userDiv);
    chatMessages.appendChild(chatBox);

    // Check user message and respond accordingly
    if (userMessage.toLowerCase() === "hello") {
        const botDiv = document.createElement("div");
        botDiv.className = "chat-message bot";
        botDiv.innerText = "SmartPlate: Hello there! Please select a query from the menu:";
        chatMessages.appendChild(botDiv);

        // Display menu options
        const menuOptions = ["Order Status", "Customer Care", "Feedback"];
        menuOptions.forEach(function(option) {
            const optionDiv = document.createElement("div");
            optionDiv.className = "chat-message bot";
            optionDiv.innerText = "SmartPlate: " + option;
            chatMessages.appendChild(optionDiv);
        });
    } else if (userMessage === "Order Status") {
        const botDiv = document.createElement("div");
```

Fig. 5.4.2 Chat-bot System

Fig. 5.4.2 shows the chat-bot system for our project. This is a basic Chabot which works on basic If and Else statements. It takes the user input and checks whether the condition matches any of the its if statements if it does it replies with the code output.

```
[ ] similarity = cosine_similarity(vectors)

[ ] sorted(list(enumerate(similarity[0])),reverse=True, key=lambda x:x[1])[1:6]

    [(1283, 0.6562044563236563),
     (3408, 0.6143190654944868),
     (5178, 0.5995012171811742),
     (331, 0.5942321724388199),
     (881, 0.582794237108619 4)]

    def recommend(recipe_name):
        recipe_index = recipes[recipes['title'] == recipe_name].index[0]
        distances = similarity[recipe_index]
        recipes_list = sorted(list(enumerate(distances)), reverse=True, key=lambda x: x[1])[1:6]

        for i in recipes_list:
            print(recipes.iloc[i[0]].title)

[ ] recommend('Italian Sausage and Bread Stuffing')

    Challah, Sausage, and Dried Cherry Stuffing
    Sourdough, Italian Sausage, and Chestnut Stuffing
    Sage Stuffing
    Cornbread Stuffing with Sausage and Collard Greens
    Stuffed Artichokes
```

Fig. 5.4.3 Recommendation system

Fig. 5.4.3 shows our main function for the recommendation system. In this recommendation system we have implemented it using vectorization and then finding out the cosine similarity between each vector. Cosine similarity ranges from 0 to 1 where 1 being the highest similarity. In this we have applied the cosine similarity on the 0^{th} index of our dataset and found out the 5 most similar indexes from the dataset. In the recommend function we have taken a user input for the recipe name and we have printed the names of the most similar recipes.

Chapter 6

Technical Specifications

1. **HTML & CSS:** HTML provides the structural framework for your website's content. It defines the layout of pages, including headings, paragraphs, lists, and links. HTML provides the foundation for integrating JavaScript for interactive features. It's also used as the basis for rendering content from a database or generating dynamic web pages. CSS enables responsive web design, ensuring that your website adapts to various screen sizes and devices. This can improve user engagement and make the site more dynamic.

2. **JavaScript:** JavaScript can be used to validate user inputs in forms. This ensures that users provide accurate and complete information when registering, logging in, or submitting orders, preventing errors and improving data quality. Can be used to display user feedback messages, such as success messages after placing an order or error messages when submitting a form with incomplete data. This provides a more user-friendly experience.

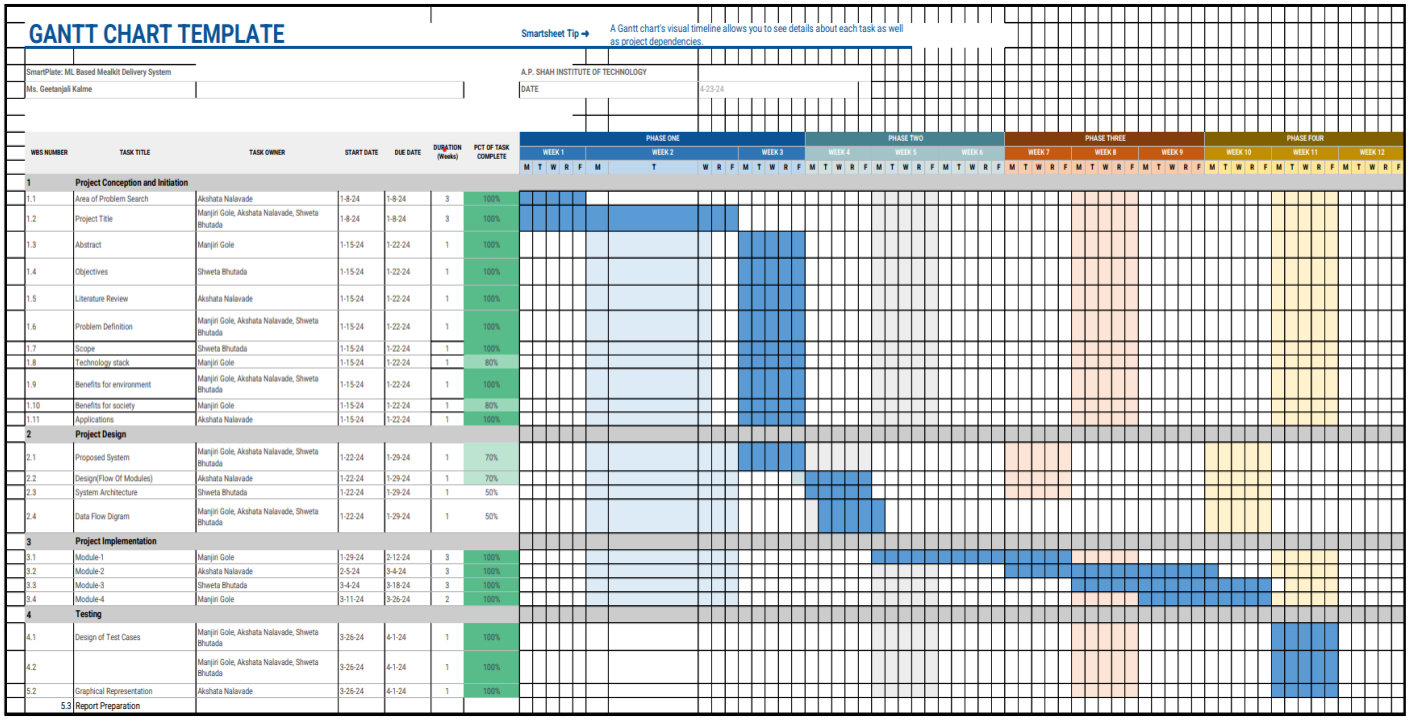
3. **MongoDB:** MongoDB uses a flexible schema-less data model, allowing you to store recipe data without a predefined structure. This flexibility accommodates changes in the types of data you want to store, making it easier to adapt to evolving recipe formats or features. As we are using Node.js for our website's backend, MongoDB offers excellent compatibility and performance when paired with Node.js applications. Recipes often have various components like ingredients, measurements, steps, and additional notes. MongoDB allows for the storage of complex data structures like arrays and nested documents, making it suitable for representing detailed recipes. MongoDB's flexibility, scalability, and efficient handling of complex data structures make it a strong candidate for serving as the database for a recipe website.

4. **NodeJS:** Node.js uses JavaScript on both the client and server sides. This means that it can use the same language and data structures throughout the application, making it easier to manage and maintain. Node.js allows us to use JavaScript on both the frontend and backend, providing a unified development environment. This means that developers can seamlessly work with JSON data, which is the native format for both JavaScript and MongoDB

5. **Python:** Python is used for recommendation system in the project. It has been implemented in google colab. Python has a vast variety of libraries which helps in integrating various projects with python. We have imported the pandas library and the numpy library for fetching and reading the dataset and for using the dataset to create a recommendation system. We have also imported Flask for connecting the frontend of the recommendation system.

Chapter 7

Project Scheduling



GANTT CHART TEMPLATE

Smartsheet Tip → A Gantt chart's visual timeline allows you to see details about each task as well as project dependencies.

SmartPlate: ML Based Mealkit Delivery System

Ms. Geetanjali Kalme

A.P. SHAH INSTITUTE OF TECHNOLOGY

DATE 4-23-24

WBS NUMBER	TASK TITLE	TASK OWNER	START DATE	DUE DATE	DURATION (Weeks)	PCT OF TASK COMPLETE
1	**Project Conception and Initiation**					
1.1	Area of Problem Search	Akshata Nalavade	1-8-24	1-8-24	3	100%
1.2	Project Title	Manjiri Gole, Akshata Nalavade, Shweta Bhutada	1-8-24	1-8-24	3	100%
1.3	Abstract	Manjiri Gole	1-15-24	1-22-24	1	100%
1.4	Objectives	Shweta Bhutada	1-15-24	1-22-24	1	100%
1.5	Literature Review	Akshata Nalavade	1-15-24	1-22-24	1	100%
1.6	Problem Definition	Manjiri Gole, Akshata Nalavade, Shweta Bhutada	1-15-24	1-22-24	1	100%
1.7	Scope	Shweta Bhutada	1-15-24	1-22-24	1	100%
1.8	Technology stack	Manjiri Gole	1-15-24	1-22-24	1	80%
1.9	Benefits for environment	Manjiri Gole, Akshata Nalavade, Shweta Bhutada	1-15-24	1-22-24	1	100%
1.10	Benefits for society	Manjiri Gole	1-15-24	1-22-24	1	80%
1.11	Applications	Akshata Nalavade	1-15-24	1-22-24	1	100%
2	**Project Design**					
2.1	Proposed System	Manjiri Gole, Akshata Nalavade, Shweta Bhutada	1-22-24	1-29-24	1	70%
2.2	Design(Flow Of Modules)	Akshata Nalavade	1-22-24	1-29-24	1	70%
2.3	System Architecture	Shweta Bhutada	1-22-24	1-29-24	1	50%
2.4	Data Flow Digram	Manjiri Gole, Akshata Nalavade, Shweta Bhutada	1-22-24	1-29-24	1	50%
3	**Project Implementation**					
3.1	Module-1	Manjiri Gole	1-29-24	2-12-24	3	100%
3.2	Module-2	Akshata Nalavade	2-5-24	3-4-24	3	100%
3.3	Module-3	Shweta Bhutada	3-4-24	3-18-24	3	100%
3.4	Module-4	Manjiri Gole	3-11-24	3-26-24	2	100%
4	**Testing**					
4.1	Design of Test Cases	Manjiri Gole, Akshata Nalavade, Shweta Bhutada	3-26-24	4-1-24	1	100%
4.2		Manjiri Gole, Akshata Nalavade, Shweta Bhutada	3-26-24	4-1-24	1	100%
5.2	Graphical Representation	Akshata Nalavade	3-26-24	4-1-24	1	100%
5.3	Report Preparation					

PHASE ONE: WEEK 1, WEEK 2, WEEK 3
PHASE TWO: WEEK 4, WEEK 5, WEEK 6
PHASE THREE: WEEK 7, WEEK 8, WEEK 9
PHASE FOUR: WEEK 10, WEEK 11, WEEK 12

Chapter 8

Results

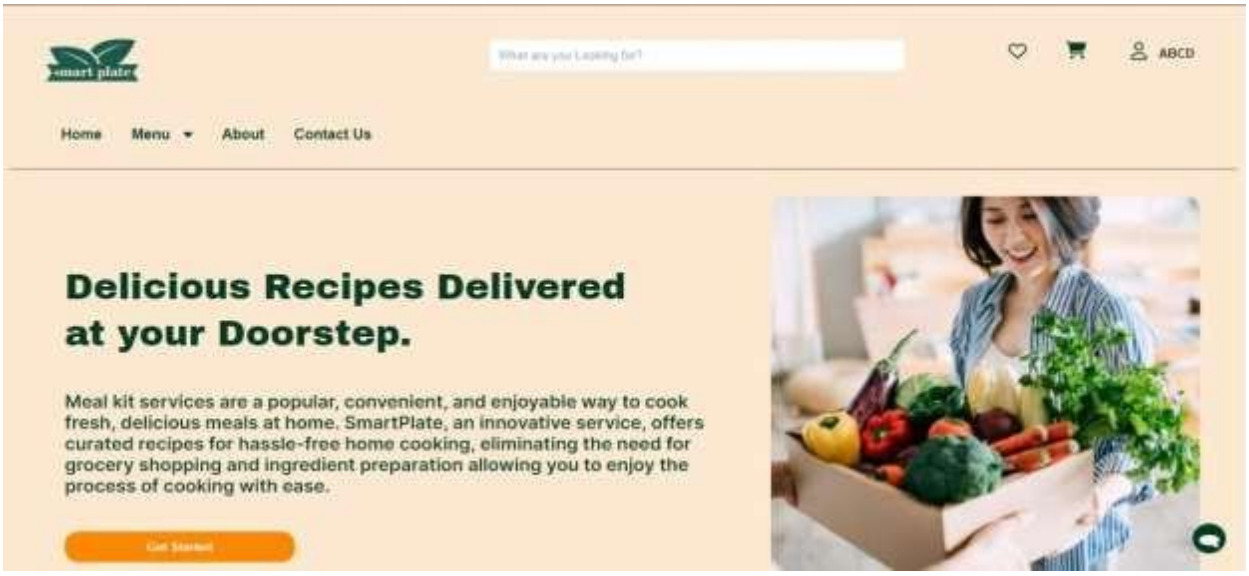


Fig. 8.1 Home page part 1

Fig. 8.2 Home page part 2

Fig. 8.1. and Fig. 8.2 show our home page or the landing page for our project.

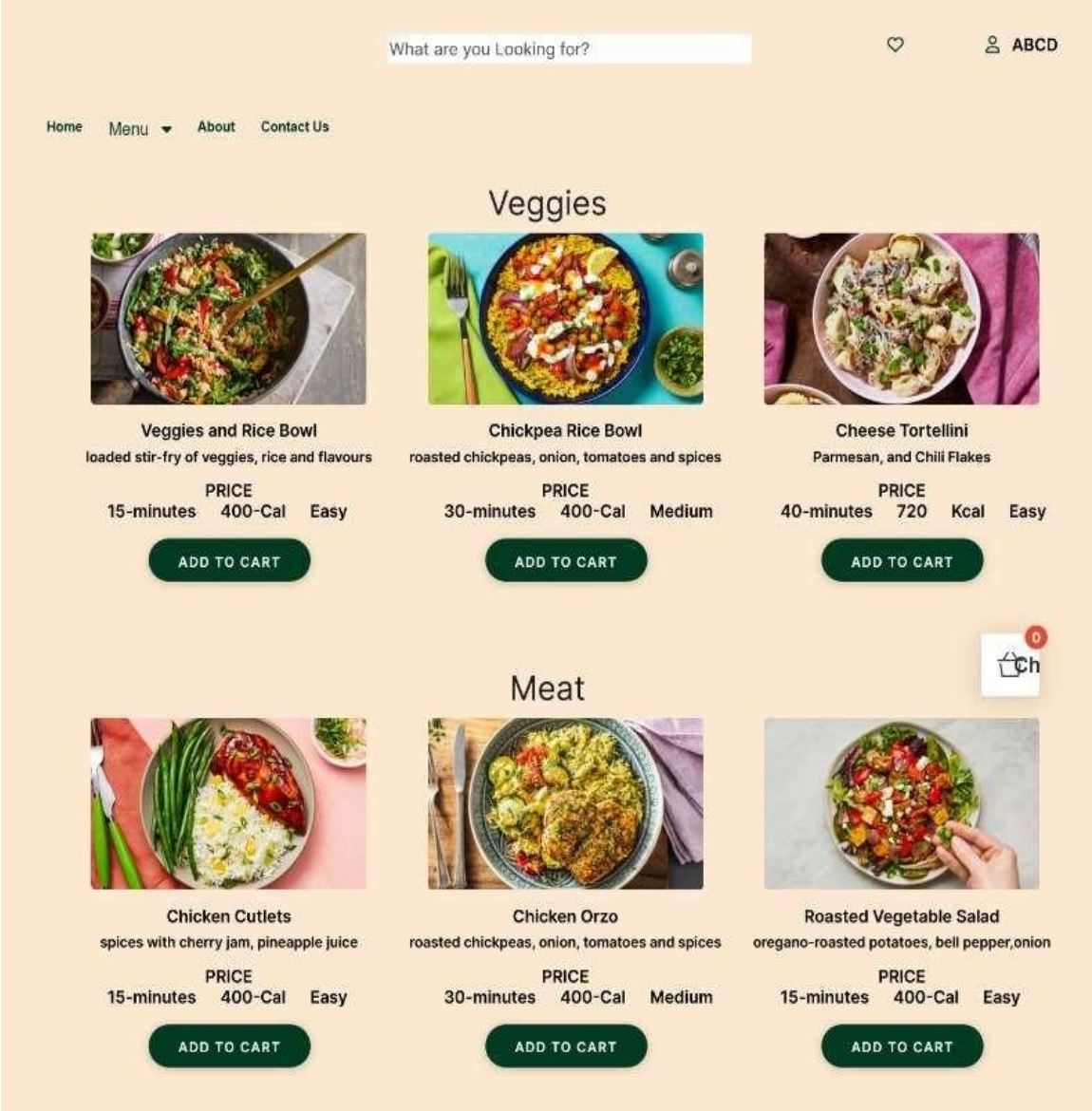


Fig. 8.3 Categories page

Fig. 8.3 shows our menu and categories page. Here you can browse through a variety of dishes. Then you can select any recipe that spikes your interest and look whether it suits your tastes and whether you want to buy it.

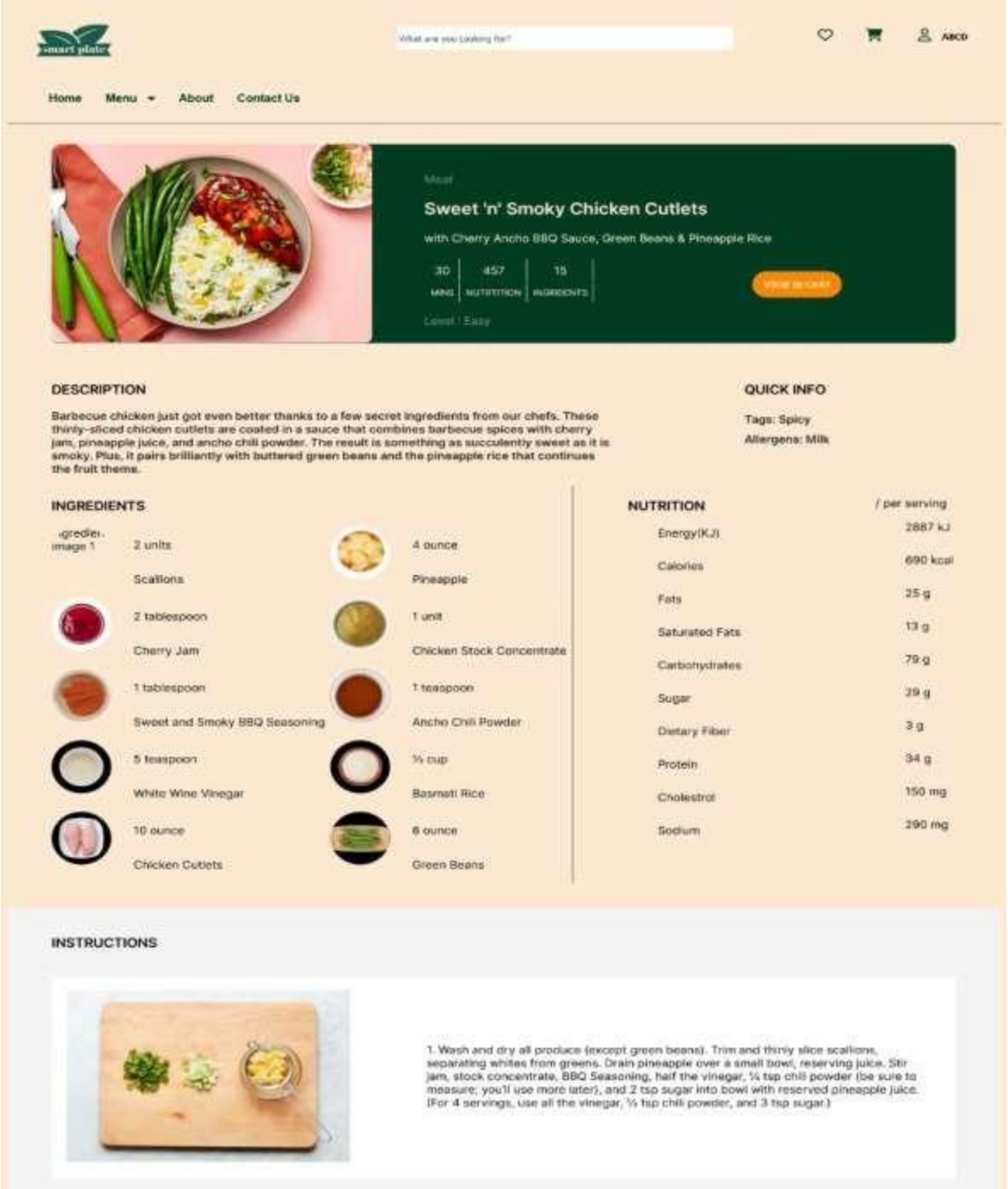


Fig. 8.4. Recipe page

Fig.8.4. shows a recipe page. Here you can find the ingredients required for the recipe along with that you can also find the nutrients provided by the recipe. You can also find step-by-step instructions along with pictures.

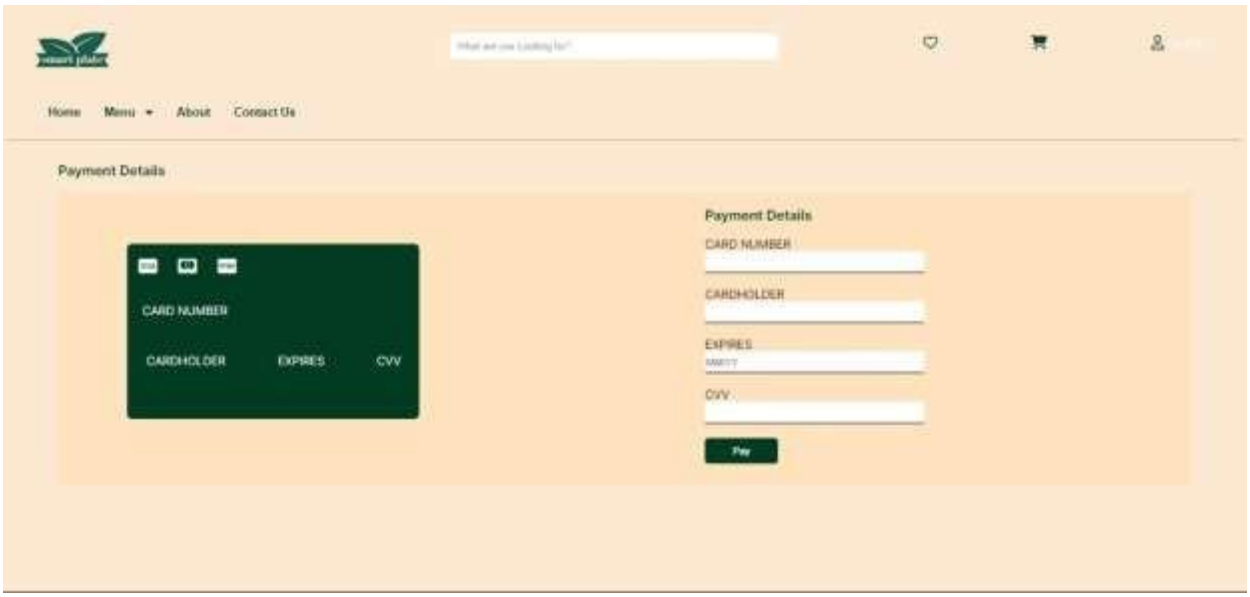


Fig.8.5. Card page

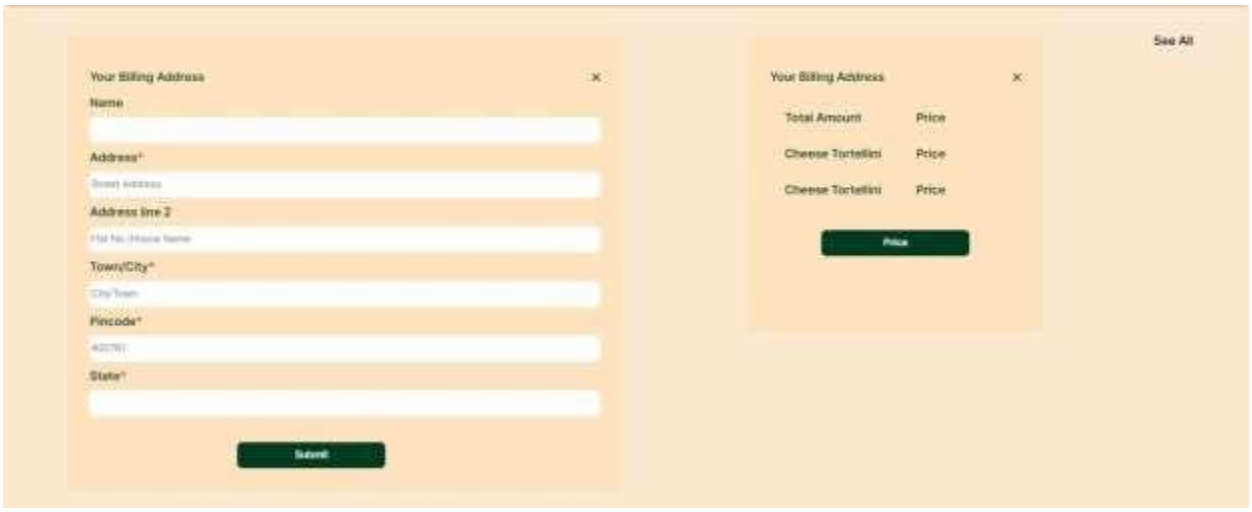


Fig. 8.6. Checkout and Billing page

Fig.8.5 and Fig. 8.6 show the checkout procedure for our project. After choosing your recipe you will be directed here for the billing and cart process where you can check ad review your order.

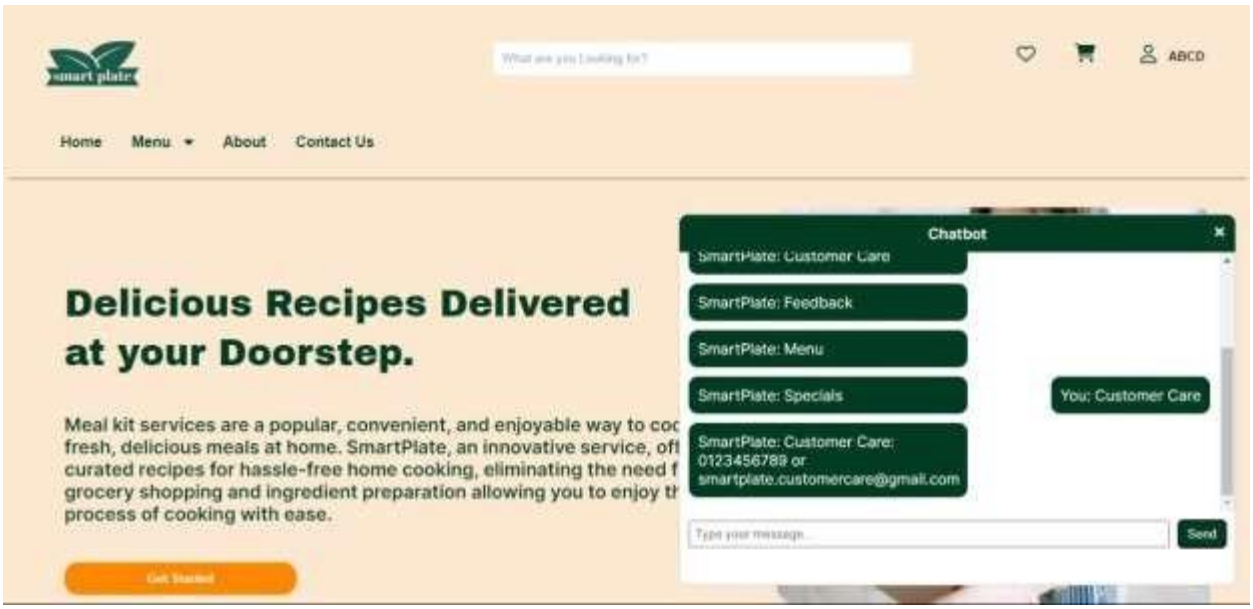


Fig. 8.7 Chat-bot

Fig.8.7. shows the implementation of the chat-bot system where you can feed some queries and get the responses from the chat-bot.

Fig. 8.8 Recommendation system

Fig 8.8 shows our recommendation system page. Here when you select a recipe from the drop down you are shown five other similar recipes which we think you might like.

Chapter 9

Conclusion

The recipe website, following extensive research and a series of well-thought-out implementations, stands ready to provide a comprehensive and enriching culinary experience for its users. The journey began with a clear understanding of the challenges faced by individuals with busy lifestyles and a passion for cooking, ultimately leading to the development of a user-centric platform. These features, informed by user feedback and preferences, are instrumental in shaping a dynamic and user-responsive environment. The website's design, while already visually appealing, is continually refined to ensure that the concept is both intuitive and accessible. Users can easily navigate through a diverse range of recipes, and enjoy a seamless ordering experience.

Chapter 10

Future Scope

i. Enable Delivery Tracking: Implement a delivery tracking system that provides users with real-time updates on the status of their orders. Users should be able to see the estimated delivery time and track the delivery's location as it progresses.

ii. Serving Adjustments: Users to adjust portion sizes to match the number of servings they require, ensuring meals are perfectly portioned for their needs. This empowers users to create dishes that suit their needs.

iii. Cooking Tutorials and tips: Tutorial videos can demonstrate step-by-step instructions for preparing recipes, while cooking tips provide helpful insights and shortcuts to improve efficiency and quality. Chef recommendations can offer expert advice on flavor pairings, cooking methods, and recipe variations, empowering users to experiment and innovate in the kitchen.

References

[1] https://www.sciencedirect.com/science/article/pii/S0002916522105526

[2] https://norma.ncirl.ie/6594/1/sasikumarjayapal.pdf

[3] https://ieeexplore.ieee.org/document/10104654

[4] https://dl.acm.org/doi/abs/10.1145/2500098.2500109

[5]https://medium.com/bithubph/payment-integration-with-node-js-express-request-and-paystack-api-8cebf51c1f52