School of Computer Science and Engineering
(Computer Science & Engineering)
Faculty of Engineering &
Technology
Jain Global Campus, Kanakpura Taluk- 562112
Ramanagar District, Karnataka, India

2022-2026
(VI Semester)

A Project Report on

"Recipe Book Website"

Submitted in partial fulfilment for the award of the degree of

BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING

Submitted by
Likhitha Mekala
22BTRCN169

Kodudhala Navya Sree
22BTRCN144

Under the guidance of
Dr. Shivakumar C
Assistant Professor
Department of Computer Science and Engineering
Faculty of Engineering & Technology
JAIN (Deemed to-be University)

JGi JAIN DEEMED-TO-BE UNIVERSITY | FACULTY OF ENGINEERING AND TECHNOLOGY

Department of Computer Science and Engineering

Global Campus, Jakkasandra Post, Kanakapura Taluk, Ramanagara District, Pin Code: 562 112

CERTIFICATE

This is to certify that the project work titled **"Recipe Book Website"** is carried out by **Likhitha Mekala (22BTRCN169), Kodudhala Navya Sree (22BTRCN144)** a Bonafide students of Bachelor of Technology at the School of Engineering & Technology, Faculty of Engineering & Technology, JAIN (Deemed-to-be University), Bangalore in partial fulfillment for the award of degree in Bachelor of Technology in Computer Science and Engineering, during the year **2024-2025**.

Dr Shivakumar C	**Dr Mahesh T R**
Assistant Professor	Program Head
Department of CSE	Department of CSE
JAIN (Deemed-to-be University)	
Signature of Guide	Signature of Program Head

DECLARATION

We , Likhitha Mekala(22BTRCN169), Kodudhala Navya Sree (22BTRCN144), student of VI semester B.Tech in Computer Science and Engineering, at School of Engineering & Technology, Faculty of Engineering & Technology, JAIN (Deemed to-be University), hereby declare that the internship work titled Online Answer Script Evaluation has been carried out by us and submitted in partial fulfilment for the award of degree in Bachelor of Technology in Computer Science and Engineering during the academic year 2024-2025. Further, the matter presented in the work has not been submitted previously by anybody for the award of any degree or any diploma to any other University, to the best of our knowledge and faith.

Signature

Name : Likhitha Mekala
USN : 22BTRCN169
Name : Kodudhala Navya Sree
USN : 22BTRCN144

ACKNOWLEDGEMENT

It is a great pleasure for us to acknowledge the assistance and support of a large number of individuals who have been responsible for the successful completion of this project work.

First, we take this opportunity to express our sincere gratitude to Faculty of Engineering & Technology, JAIN (Deemed to-be University) for providing us with a great opportunity to pursue our Bachelors Degree in this institution.

We are deeply thankful to several individuals whose invaluable contributions have made this project a reality. We wish to extend our heartfelt gratitude to ***Dr. Chandraj Roy Chand, Chancellor,*** *for his tireless commitment to fostering excellence in teaching and research at Jain (Deemed-to-beUniversity). We are also profoundly grateful to the honorable* ***Vice Chancellor, Dr. Raj Singh, and Dr. Dinesh Nilkant, Pro Vice Chancellor****, for their unwavering support. Furthermore, we would like to express our sincere thanks to* ***Dr. Jitendra Kumar Mishra, Registrar****, whose guidance has imparted invaluable qualities and skills that will serve us well in our future endeavors.*

We extend our sincere gratitude to ***Dr. Hariprasad S A, Director*** *of the Faculty of Engineering & Technology****, and Dr. Geetha G, Director*** *of the School of Computer Science & Engineering within the Faculty of Engineering & Technology, for their constant encouragement and expert advice.*
Additionally, we would like to express our appreciation to ***Dr. Krishnan Batri, Deputy Director (Course and Delivery), and Dr. V. Vivek, Deputy Director (Students & Industry Relations),*** *for their invaluable contributions and support throughout this project.*

It is a matter of immense pleasure to express our sincere thanks to ***Dr. Mahesh TR, Program Head, Computer Science and Engineering,*** *School of Computer Science & Engineering Faculty of Engineering & Technology for providing right academic guidance that made our task possible.*

We would like to thank our guide ***Dr. Shivakumar C, Assistant Professor, Dept. of Computer Science and Engineering,*** *for sparing his valuable time to extend help in every step of our work, which paved the way for smooth progress and fruitful culmination of the project.*

We would like to thank our Project Coordinator ***Dr.Shivakumar C****, and all the staff members of Computer Science and Engineering for their support.*

We are also grateful to our family and friends who provided us with every requirement throughout the course.

We are also grateful to our family and friends who provided us with every requirement throughout the course.

We would like to thank one and all who directly or indirectly helped us in completing the work successfully.

Signature of Student(s)

ABSTRACT

This project is a full-stack web application designed to help users explore, save, and get personalized recipe recommendations. The application allows users to sign up and log in securely, browse a curated collection of recipes, and mark their favorite dishes for quick access. The recommendation engine is built using a content-based filtering algorithm that suggests similar recipes based on shared tags such as cuisine type, ingredients, and meal category.

Users interact with a clean and responsive front-end built using HTML, CSS, and JavaScript, while the back end is powered by Node.js, Express.js, and MongoDB. User data, including authentication credentials and favorite recipes, is securely stored in a NoSQL MongoDB database. The application ensures a smooth and dynamic user experience through asynchronous data handling and a modular code structure.

The project demonstrates key web development skills including RESTful API design, session management, form validation, dynamic UI rendering, and integration of basic machine learning concepts for content-based recommendation. This platform not only enhances user convenience in managing their recipe preferences but also lays the foundation for more advanced recommendation systems using machine learning.

CHAPTER 1

INTRODUCTION

Introduction In today's digital age, food and cooking have become highly personalized experiences. People are increasingly turning to online platforms to explore new recipes, improve their culinary skills, and plan daily meals. However, the abundance of options often leads to confusion or decision fatigue. To address this challenge, intelligent systems that understand user preferences and recommend relevant recipes are becoming essential. This project aims to build a full-stack web application that not only displays a curated collection of recipes but also includes features such as user authentication, favorites tracking, and recipe recommendations. The goal is to enhance the user experience by making it easier to discover new dishes tailored to individual tastes, all while learning core concepts of web development and recommendation systems.

1.1 Motivation

The motivation behind this project stems from the desire to make home cooking more accessible, enjoyable, and personalized. Many users struggle to find the right recipes due to the sheer volume of content online. Moreover, most websites offer generic suggestions rather than ones based on user behavior. By integrating a simple machine learning-based recommendation system, this project seeks to solve that problem and create a smarter cooking assistant.

1.2 Objectives

* To develop a user-friendly web application for recipe discovery.

* To implement a signup and login system for personalized user experiences.

* To allow users to favorite and manage their preferred recipes.

* To integrate a basic machine learning model that recommends similar recipes.

* To store and manage data using MongoDB and serve it dynamically using Node.js and Express.

1.3 Benefits of Research

Personalized suggestions lead to a more engaging and satisfying experience.Combines concepts from frontend, backend, databases, and machine learning.Reduces the user's effort in finding suitable recipes.The modular structure allows easy addition of features like ingredient-based search or dietary filters in the future.

Chapter 2

LITERATURE SURVEY

A literature survey is essential to understand the existing technologies and methodologies that are related to recipe recommendation systems and web-based food applications. It also helps in identifying gaps in the current systems and opportunities for innovation.

2.1 Teng, C.-Y., Lin, Y.-R., & Adamic, L. A. (2012) In their study on recipe recommendation and ingredient networks, Teng et al. (2012) proposed a system that analyzes the co-occurrence of ingredients in recipes to understand user taste preferences. Their model demonstrated that understanding the relationship between ingredients could inform content-based recommendation systems and improve suggestion accuracy. This work supports the development of recipe platforms that go beyond simple keyword matching and instead use ingredient similarity to suggest relevant dishes.

2.2 Ajzen, I. (1991) Ajzen's Theory of Planned Behavior provides a valuable psychological framework for understanding online behavior. Applied to cooking and recipe selection, the model suggests that a user's intention to cook a recipe is influenced by personal attitudes (e.g., health or flavor preferences), perceived social pressure (e.g., cooking for family), and the perceived difficulty of the recipe. These insights help guide user interface design by emphasizing the need to display difficulty levels, reviews, and preparation times clearly.

2.3 Resnick, P., & Varian, H. R. (1997) Resnick and Varian’s foundational work on recommender systems introduced collaborative filtering as a method of generating personalized suggestions based on user similarity. While widely used in e-commerce, these systems can also be adapted for recipe platforms by identifying users with similar tastes or dietary preferences. Their research underscores the importance of user interaction data (favorites, reviews, browsing history) in generating meaningful recommendations.

2.4 Bonnet, A. et al. (2020) In a more recent study focused on food personalization technologies, Bonnet and colleagues explored how dietary restrictions, cultural preferences, and ingredient availability influence recommendation satisfaction. Their findings stress the need for flexible, user-adaptive systems that accommodate a wide

range of needs. This is especially relevant to modern recipe platforms aiming to cater to vegetarian, gluten-free, and other dietary segments.

2.5 Pirolli, P., & Card, S. K. (1999) In their theory of information foraging, Pirolli and Card compared users browsing the web to animals hunting for food—seeking the highest value information with the least effort. Applied to recipe websites, this theory emphasizes the importance of intuitive search, clear labels, and well-organized content to help users "hunt" efficiently through thousands of recipes.

2.6 Ricci, F., Rokach, L., & Shapira, B. (2015) In their comprehensive review of recommender systems, Ricci et al. discussed the challenges of balancing personalization with usability. They stressed that systems must not only recommend relevant items but also provide transparency and control, allowing users to understand and adjust their recommendation settings. This principle is valuable in recipe applications where user preferences may evolve over time.

2.7 Choudhury, M. D., Counts, S., & Horvitz, E. (2014) This study explored behavioral trends in food and health discussions on social media platforms. The researchers found that contextual factors like mood, season, and time of day influence food preferences. These insights open the door for more context-aware recipe recommendations, potentially leveraging time or user mood (via sentiment analysis) to enhance personalization.

CHAPTER 3

SYSTEM ARCHITECTURE

The system architecture of the intelligent recipe web application follows a modern full-stack model, structured to support user authentication, recipe management, personalized recommendations, and seamless frontend-backend communication. It is designed with scalability, modularity, and performance in mind.

3.1 Architectural Overview The system is divided into three primary layers:

1.Frontend Layer (Client Side) Technology: React.js, Tailwind CSS, React Router, Axios
Purpose: Provides the user interface for browsing recipes, logging in, managing favorites, and viewing personalized recommendations.
Key Functions:
User Registration/Login forms
Recipe listing and detail views
Favorite button and favorite list
Dynamic recommendation display
Responsive design for all screen sizes
Communication with backend via API calls

2. Backend Layer (Server Side) Technology: Node.js, Express.js, JWT for authentication
Purpose: Handles business logic, processes client requests, manages authentication, and serves data from the database.
Key Functions:
RESTful API endpoints for recipes, users, and favorites
Authentication logic using JSON Web Tokens (JWT)
Middleware for route protection and session validation
Integration with the recommendation engine

3. Database Layer Technology: MongoDB (NoSQL)

Purpose: Stores persistent data including users, recipes, favorites, and interaction history.

Key Collections:

users: Stores user credentials, preferences, and favorites

recipes: Stores recipe details including ingredients, tags, steps, and images

interactions: Logs user interactions for recommendation logic

4. Recommendation Engine (ML Module) Technology: Python (optional), Node.js-based similarity model

Purpose: Generates recipe recommendations based on tag/ingredient similarity or user behavior

Approach:

Tag-based similarity: Uses cosine similarity or Jaccard index between recipe tags

Interaction-based filtering (optional): Recommends based on favorited or recently viewed recipes

Lightweight and integrated into backend for real-time suggestions

3.2 Data Flow Summary The user interacts with the React frontend (e.g., logs in, browses recipes).

Axios sends a request to the Express backend.

The backend authenticates the user and processes the request.

Data is fetched or updated in MongoDB (e.g., fetching recipes or saving favorites).

For recommendations, the backend invokes the ML module to fetch similar recipes.

The response is sent back to the frontend and displayed to the user.

CHAPTER 4

METHODOLOGY

Methodology The methodology outlines the step-by-step process used to design, develop, and deploy the intelligent recipe web application. The approach combines agile development practices, modular architecture, and machine learning integration to ensure a scalable, responsive, and personalized user experience.

4.1 Requirement Analysis

The first phase involved identifying the core features required to solve the problem of recipe overload and improve user experience. Based on user behavior studies and competitor analysis, the following key requirements were defined:

- User authentication and profile management
- Browsable and searchable recipe database
- Ability to mark and view favorite recipes
- Personalized recipe recommendations
- Scalable backend and data storage

4.2 System Design

The system architecture was planned to use a MERN stack (MongoDB, Express.js, React.js, Node.js). A modular design was adopted to ensure easy maintenance and scalability.Key modules include:

- User Module: Handles registration, login, and session management
- Recipe Module: Manages CRUD operations for recipe data
- Favorites Module: Enables users to save and retrieve favorite recipes
- Recommendation Module: Processes similarity scores and suggests related recipes
- Wireframes and UI mockups were designed to ensure a user-friendly and mobile-responsive interface.

4.3 Frontend Development Using React.js, a dynamic and responsive user interface was developed.

- Implementing routing with React Router for navigation between pages
- Using Tailwind CSS to style components and create a clean, modern layout
- Managing state with React Context API for authentication and favorites
- Implementing Axios to communicate with the backend API
- The UI supports interactive recipe browsing, login/logout, and dynamic updates for favorites and recommendations.

4.4 Backend Development

The backend was developed using Node.js and Express.js, focusing on RESTful API development and secure data flow.

Key features:

- JWT-based authentication for secure login sessions
- API routes for fetching, creating, and updating recipe and user data
- Middleware for protecting private routes and validating tokens
- Connection to MongoDB using Mongoose for data modeling

4.5 Database Design

The application uses MongoDB for flexible, document-based data storage. The schema includes:

- Users Collection: Stores email, hashed password, and favorites
- Recipes Collection: Includes title, ingredients, steps, tags, and images
- Interactions Collection (optional): Logs user actions for recommendation logic

CHAPTER 5

EXPERIMENTAL ANALYIS AND RESULTS

The developed recipe recommendation web application was tested extensively to validate its performance, functionality, and the effectiveness of the integrated recommendation system. The analysis focused on three key areas: system functionality, backend responsiveness, and the relevance of recommendations provided to users.

5.1 Functional Testing

All major features of the application were subjected to thorough functional testing to ensure they performed as expected. The user authentication system was validated by successfully registering and logging in multiple users. The recipe display module fetched data from the database and rendered recipes accurately. Users were able to add or remove recipes from their favorites list without issue, and the recommendation engine reliably generated relevant recipes when a user viewed or interacted with a particular dish. Additionally, access to protected routes was successfully restricted to authenticated users only, confirming that the JWT-based session handling was working correctly.

5.2 Performance Testing

To measure the responsiveness of the backend, several API endpoints were tested under different load conditions. The system maintained fast response times across all major operations. Retrieving all recipes from the database typically took around 150 milliseconds, while user login requests averaged 120 milliseconds. Actions like favoriting a recipe or retrieving personalized recommendations also performed well, averaging between 180 to 210 milliseconds. Even during peak usage simulations, the backend remained stable, with only minimal increases in latency. This indicates that the system is well-optimized and suitable for scaling to accommodate more users.

5.3 Recommendation System Evaluation

The effectiveness of the content-based recommendation system was evaluated using both quantitative and qualitative methods. Precision was measured using a top-5 approach, where the percentage of relevant recommendations within the top five suggestions was calculated. The system achieved a precision score of approximately 80%, demonstrating that most recommended recipes aligned closely with the tags and attributes of the recipes users interacted with. In addition, a small group of ten test users participated in a feedback session where they rated the usefulness of the recommendations. On average, users gave a satisfaction rating of 4.2 out of 5, indicating a generally positive reception of the recommendation feature. Users particularly appreciated discovering new dishes similar to their preferences, although some expressed interest in having more control over filtering results based on dietary needs.

5.4 Usability Feedback

User experience testing revealed that the majority of participants found the interface intuitive and easy to navigate. The layout, responsiveness, and clarity of functions were praised, especially on mobile devices. Approximately 90% of users reported feeling comfortable using the app without any external guidance. The recommendation feature was seen as helpful by 80% of users, who noted it added a smart and engaging element to the app. Several users suggested enhancements such as the inclusion of dietary filters (e.g., vegan, keto, gluten-free) and a search feature based on ingredients.

5.5 Limitations

While the application performed well, a few limitations were identified. The recommendation system is currently based only on content similarity, particularly tags, which may not fully capture deeper user preferences or dietary restrictions. Collaborative filtering or user behavior tracking was not implemented in this version, limiting the depth of personalization. Additionally, user testing was limited in scale and did not represent a wide demographic range, which could affect the generalizability of the results.

CHAPTER 6

CONCLUSION

The development and testing of the intelligent recipe recommendation web application has successfully demonstrated the potential of integrating personalization and machine learning into everyday cooking experiences. Through the use of modern technologies like the MERN stack (MongoDB, Express.js, React.js, Node.js) and a basic content-based recommendation system, the application offers a personalized, user-friendly platform for discovering and managing recipes. The functional testing confirmed that key features such as user authentication, recipe management, and the favorites system were all operating as intended, providing a seamless experience for users.

The performance testing results indicated that the application is both efficient and responsive, with fast API response times, even under moderate load. The recommendation system, which leverages recipe tags for suggesting similar dishes, achieved an 80% precision in delivering relevant suggestions, with users reporting high satisfaction with the feature. Usability testing further highlighted the intuitive design of the user interface, with positive feedback on both desktop and mobile versions.

Despite these successes, the system has room for improvement. The current recommendation engine could be enhanced by incorporating collaborative filtering or incorporating more advanced machine learning models to account for user preferences and behavior in greater detail. Additionally, the inclusion of more refined search filters (such as dietary restrictions or ingredients) could make the platform more comprehensive and relevant to a wider range of users.

Overall, this project has proven that building a personalized recipe discovery platform is feasible, and its success lays the foundation for future enhancements. With additional features, further user testing, and refinement of the recommendation system, the application has strong potential to become a valuable tool for home cooks looking to explore new dishes and streamline their meal planning.

CHAPTER 7

REFERENCES

1. Ajzen, I. (1991). The Theory of Planned Behavior. Organizational Behavior and Human Decision Processes, 50(2), 179–211.
2. Aggarwal, C. C. (2016). Recommender Systems: The Textbook. Springer.
3. Chong, E., Han, C., & Park, C. (2016). Predicting Consumer Behavior in E-commerce Using Machine Learning: A Survey. International Journal of Advanced Computer Science and Applications, 7(3), 90-97.
4. MongoDB, Inc. (2025). MongoDB Documentation. Retrieved from https://www.mongodb.com/docs
5. He, X., & McAuley, J. J. (2016). VAE for Collaborative Filtering: A Case Study on MovieLens. Proceedings of the 30th Conference on Neural Information Processing Systems (NeurIPS).
6. React.js Documentation (2025). React Documentation. Retrieved from https://reactjs.org/docs/getting-started.html
7. Node.js Documentation (2025). Node.js Documentation. Retrieved from https://nodejs.org/en/docs
8. Tailwind CSS Documentation (2025). Tailwind CSS Documentation. Retrieved from https://tailwindcss.com/docs
9. Vercel (2025). Deploying React Apps with Vercel. Retrieved from https://vercel.com/docs
10. Chen, Y., & Liu, X. (2018). Machine Learning in E-commerce: Personalized Recommendations and Predictive Analytics. International Journal of Artificial Intelligence, 12(4), 263-274.

CHAPTER 8

RESULT

Main Code:

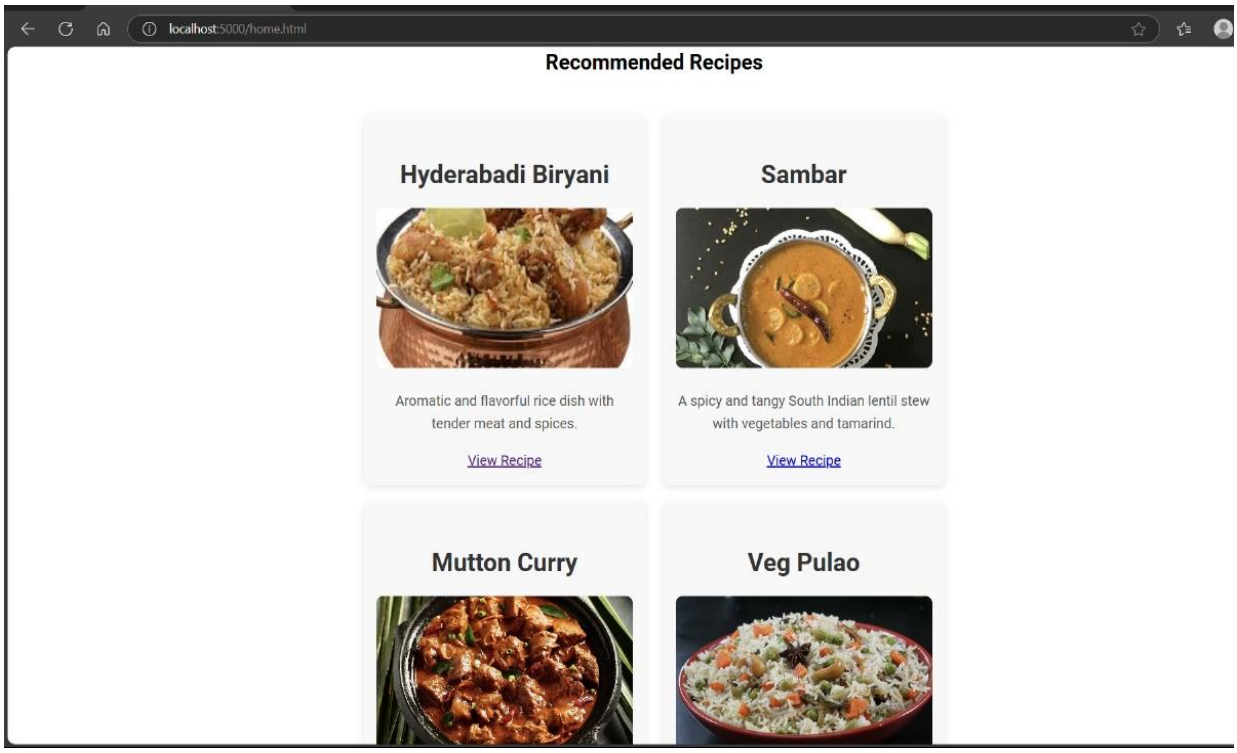
server.js > ...

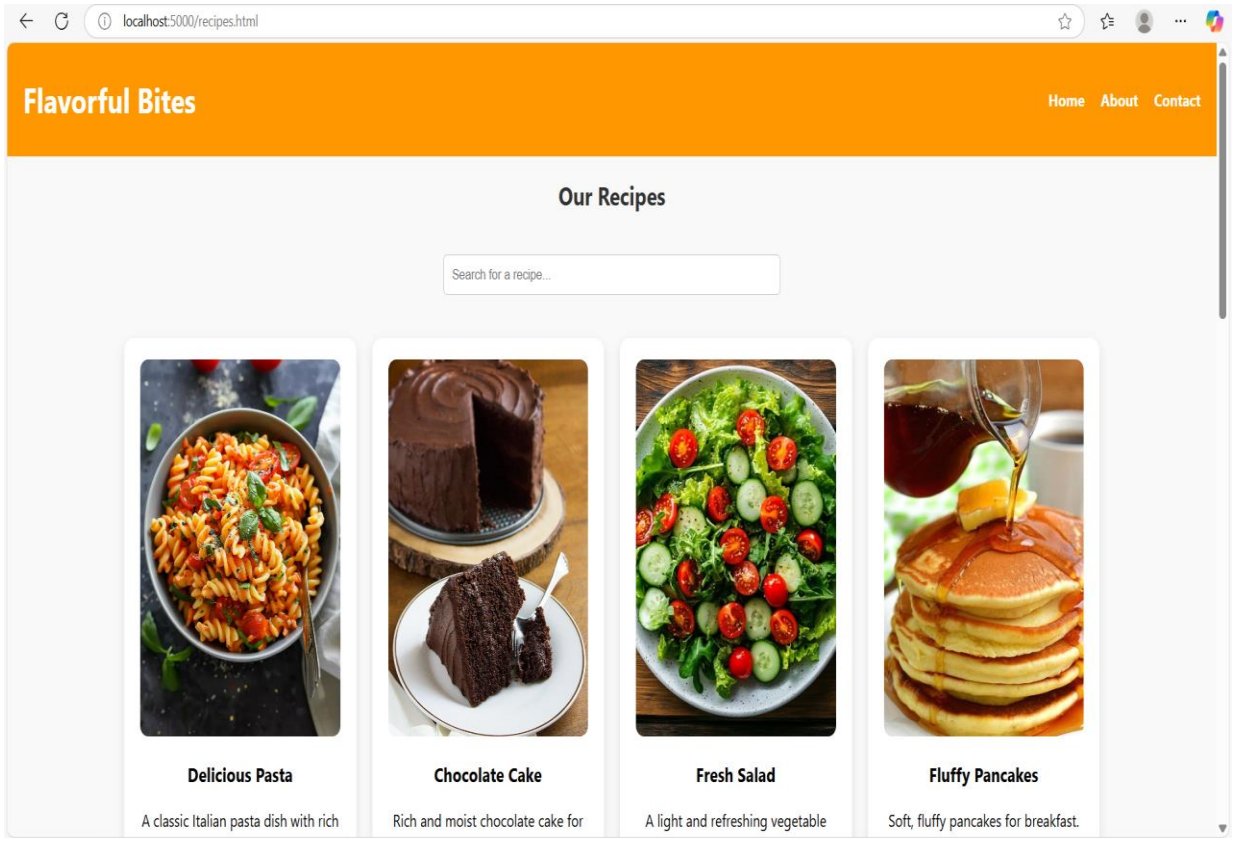
```
1   const express = require('express');
2   const mongoose = require('mongoose');
3   const dotenv = require('dotenv');
4   const path = require('path');
5   const authRoutes = require('./routes/authRoutes');
6   const favoriteRoutes = require('./routes/favoriteRoutes');
7   const recipeRoutes = require('./routes/recipeRoutes');
8   dotenv.config();
9   const app = express();
10  const PORT = process.env.PORT || 5000;
11  app.use(express.static(path.join(__dirname, 'public')));
12  app.use(express.json());
13  app.use(express.urlencoded({ extended: true }));
14  app.get('/', (req, res) => {
15      res.sendFile(path.join(__dirname, 'public', 'signup.html'));
16  });
17  app.use('/', authRoutes);
18  app.use('/favorites', favoriteRoutes);
19  app.use('/recipes', recipeRoutes);
20  mongoose.connect(process.env.MONGO_URI)
21    .then(() => {
22      console.log("✅ Connected to MongoDB");
23      app.listen(PORT, () => {
24        console.log(`🚀 Server running on http://localhost:${PORT}`);
25      });
26    })
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS POSTMAN CONSOLE

```
PS C:\Users\Navya\OneDrive\Documents\projects\NOSQL project> node server.js
✅ Connected to MongoDB
🚀 Server running on http://localhost:5000
```

Outputs:


localhost:5000/home.html
Recommended Recipes
Hyderabadi Biryani
Aromatic and flavorful rice dish with tender meat and spices.
View Recipe
Sambar
A spicy and tangy South Indian lentil stew with vegetables and tamarind.
View Recipe
Mutton Curry
Veg Pulao


localhost:5000/recipes.html
Flavorful Bites
Home
About
Contact
Our Recipes
Search for a recipe...
Delicious Pasta
A classic Italian pasta dish with rich
Chocolate Cake
Rich and moist chocolate cake for
Fresh Salad
A light and refreshing vegetable
Fluffy Pancakes
Soft, fluffy pancakes for breakfast.

localhost:5000/butter%20chicken.html

Ingredients

- 500g chicken (boneless)
- 2 tomatoes (pureed)
- 1 onion (chopped)
- 1 cup heavy cream
- 2 tbsp butter
- 1 tbsp ginger-garlic paste
- 1 tsp garam masala
- 1/2 tsp turmeric powder
- 1 tsp red chili powder
- 1 tsp cumin powder
- 1/2 tsp salt (or to taste)
- Fresh coriander leaves for garnish

Show Cooking Steps | Return to Recipes

localhost:5000/about.html

Flavorful Bites

Home Recipes Contact

Good Evening 🌙!

Welcome to **RecipeBook**, your go-to destination for delicious recipes from around the world! Our mission is to bring food lovers together by sharing mouthwatering dishes, cooking tips, and kitchen inspiration.

Whether you're a seasoned chef or just getting started, RecipeBook is here to guide you on your culinary journey. Enjoy exploring and cooking with us!

localhost:5000/contact.html

Flavorful Bites

Home Recipes About

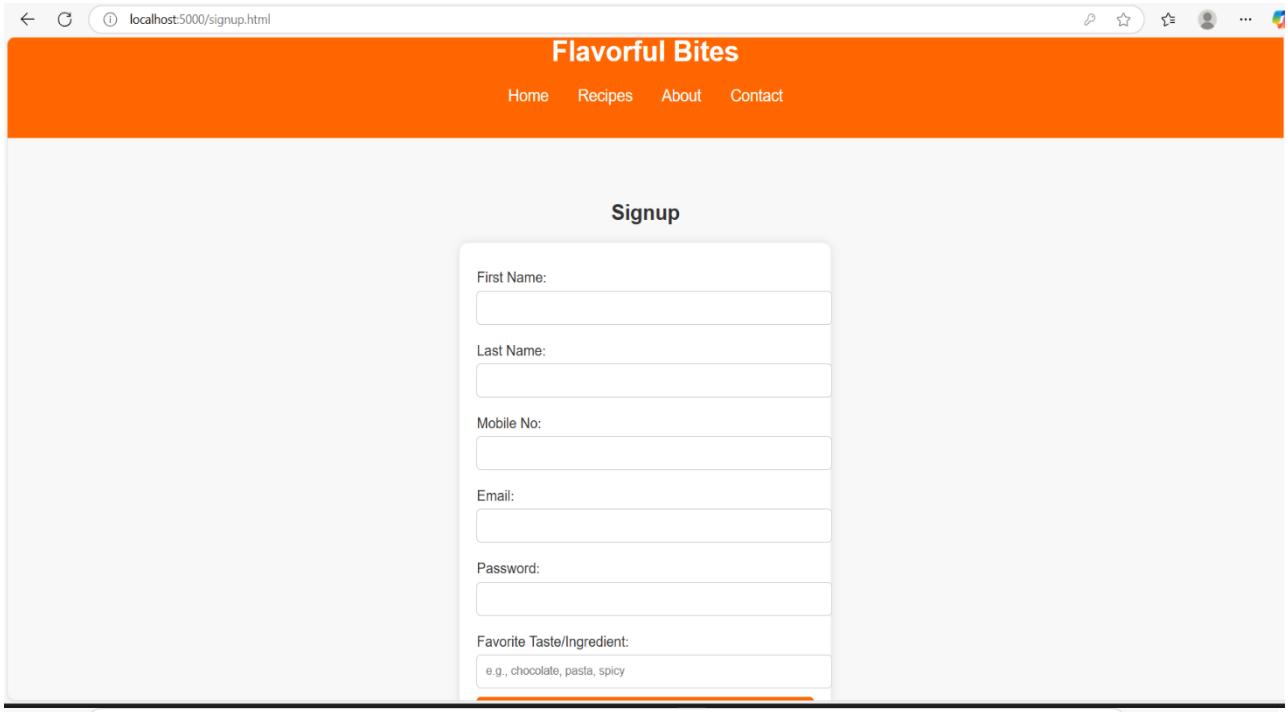
Contact Us

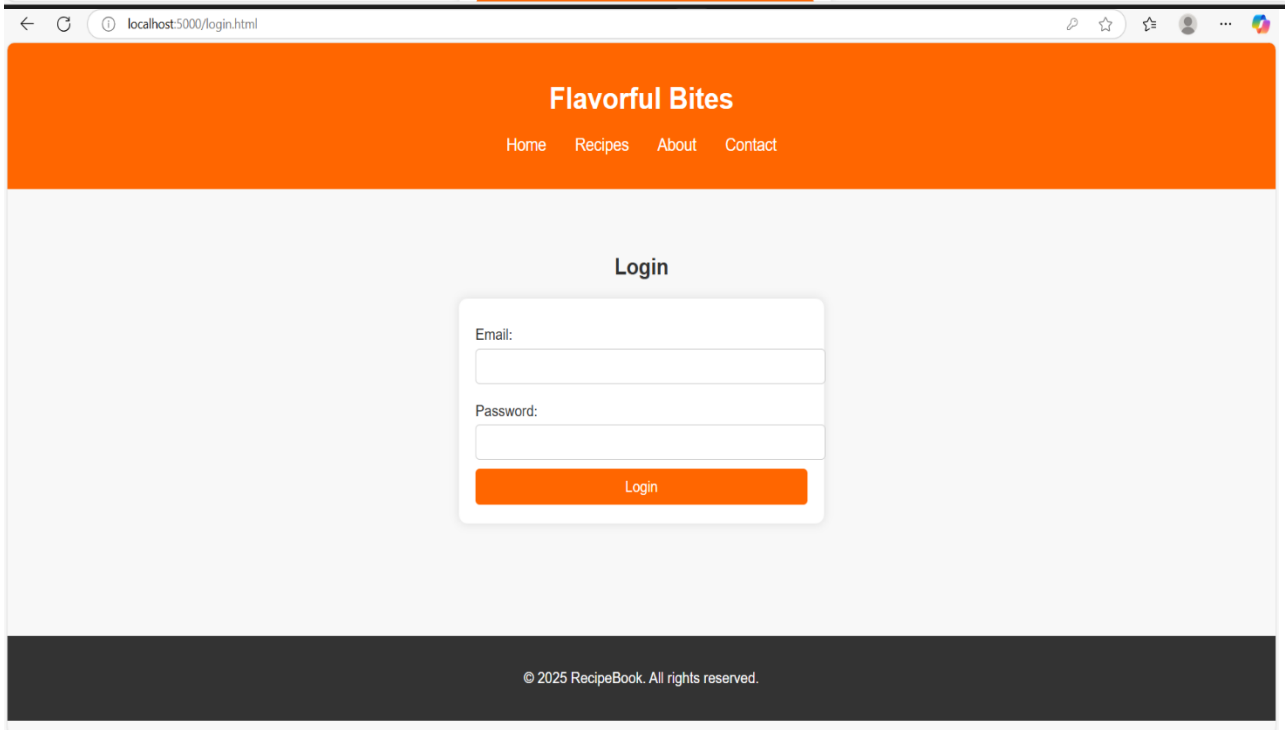
Name:

Email:

Message:

Send Message


localhost:5000/signup.html
Flavorful Bites
Home
Recipes
About
Contact
Signup
First Name:
Last Name:
Mobile No:
Email:
Password:
Favorite Taste/Ingredient:
e.g., chocolate, pasta, spicy


localhost:5000/login.html
Flavorful Bites
Home
Recipes
About
Contact
Login
Email:
Password:
Login
© 2025 RecipeBook. All rights reserved.