Project Report for the Subject

DATA ENGINEERING
(UCS677)

Submitted By:
RAVNEET KAUR (102203202)
HARDETYA GILL (102203213)

BE COE- 3C21

FORK & FEAST
Online Recipe Book & Food Blog

Course Instructor:
Dr. MANDEEP KAUR

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)
COMPUTER SCIENCE DEPARTMENT,
THAPAR INSTITUTE OF ENGINEERING AND TECHNOLOGY,
PATIALA

Session: Jan - May 2025
Date of Submission: 6 May 2025

CONTENTS

INTRODUCTION

In today's fast-paced world, where health consciousness and time efficiency are paramount, the need for a personalized, intuitive, and dynamic platform that helps users discover, manage, and track food recipes has become increasingly relevant. The project titled "Fork & Feast" is an intelligent recipe management and discovery platform designed to meet the evolving culinary and health-related needs of modern users. This project aims to bring a holistic experience to users who want to plan their meals based on specific ingredients, nutritional content, or calorie goals, while also allowing them to store and manage their personal recipe collections securely.

The platform is designed to be user-friendly, mobile-responsive, and secure. It provides essential features like user registration, login authentication, recipe discovery based on filters like calories and protein, a personal recipe dashboard ("My Recipes"), and a secure password recovery system. By integrating a robust backend with a clean frontend interface, the project also emphasizes data management using MongoDB and ensures real-time interaction between the user and the database through REST APIs.

The core objective of this project is to create a sustainable and scalable solution for recipe enthusiasts, health-focused individuals, and casual cooks by simplifying the process of meal discovery and personalization. The platform not only aids in achieving health goals but also promotes the use of available ingredients, thereby reducing food waste.

PROBLEM STATEMENT

Many users today rely on multiple sources—ranging from mobile apps to handwritten notes—to manage recipes or search for meals that fit their dietary preferences. However, these systems are often fragmented, lack personalization, or are overly complicated. There exists a significant gap in providing a unified, user-driven system that not only allows for ingredient-based discovery but also lets users curate and manage their own recipes with ease.

Furthermore, with rising security concerns in digital platforms, many recipe apps fail to provide secure and user-centric features such as personalized authentication, robust password recovery, and user-specific recipe storage. Users face difficulty remembering passwords, and most platforms do not verify identity through linked data such as phone numbers.

Another major challenge is the inefficiency in retrieving recipes based on specific nutritional values like protein, carbs, or calorie count. Many platforms do not provide advanced filters or rely solely on generic tags, failing to cater to users with specific dietary goals.

This project aims to address all these concerns by building a centralized, secure, and intelligent recipe management system that not only provides dynamic recipe search features but also enables users to save, edit, and retrieve their own recipes, ensuring both flexibility and security.

SPECIFIC REQUIREMENTS

3.1 FUNTIONAL REQUIREMENTS

- *User Authentication:* Users must be able to register, log in, and manage their credentials.
- *Recipe Search Engine:* Users can search for recipes by entering ingredients or filtering based on nutritional parameters such as calories or protein.
- *My Recipes Section*: Each registered user should be able to add, view, update, and delete their own recipes.
- *Forgot Password Feature:* Users can securely reset their password by providing their registered email and phone number.
- *Recipe Storage in MongoDB*: Recipes are stored only when users save them. If searched again, the system first checks MongoDB before calling an external API.

3.2 NON- FUNTIONAL REQUIREMENTS

- *Security:* All password handling is secured using best practices such as encryption and phone-based identity verification.
- *Scalability*: The architecture supports future enhancements like meal planning, shopping lists, and integration with health APIs.
- *User Interface:* Clean, intuitive, and mobile-friendly UI built with modern HTML and CSS.
- *Performance*: Fast response times for user interactions and recipe queries, with minimal load times even under high traffic.

SYSTEM ARCHITECTURE

4.1 CONTEXT LEVEL DIAGRAM

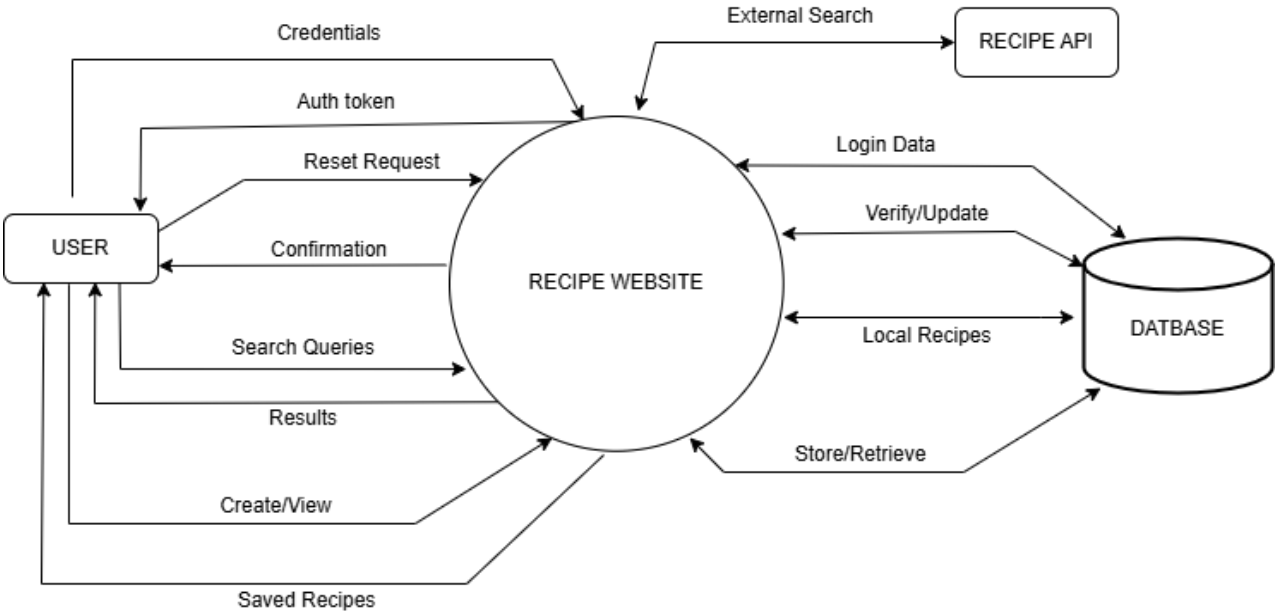


Fig 4.1- Context Level Diagram

4.2 DATA FLOW DIAGRAM

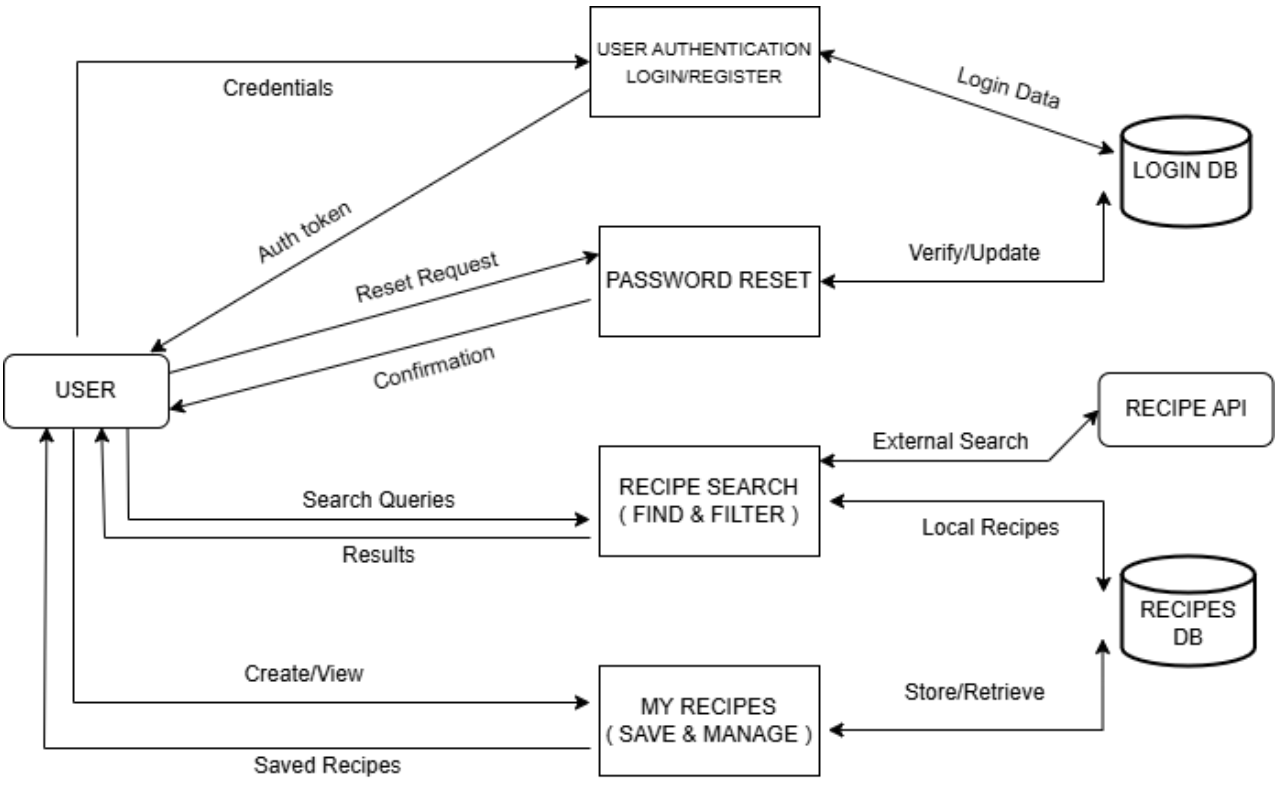


Fig 4.2- Data Flow Diagram

SYSTEM SPECIFICATIONS

5.1 HARDWARE SPECIFICATIONS

- *Processor*: Intel Core i5 or higher
- *RAM*: 8 GB or more
- *Storage*: 256 GB SSD minimum
- *Display:* Full HD screen for UI testing

5.2 SOFTWARE SPECIFICATIONS

- *Frontend*: HTML, CSS, JavaScript (Vanilla or React for future scalability)
- *Backend*: Node.js with Express.js for handling API requests
- *Database:* MongoDB (Cloud or Local)
- *Authentication*: JWT-based session management
- *Development Tools:* Visual Studio Code, Postman, MongoDB Compass
- *Server:* Localhost during development (Node.js), deployable to cloud (Render, Vercel, etc.)

These specs ensure smooth development and deployment while maintaining system performance under expected workloads.

TOOLS USED

6.1 DEVELOPMENT TOOLS

- Visual Studio Code: The primary IDE used for writing frontend and backend code.
- Node.js & Express: Provided a lightweight and fast backend framework for building RESTful APIs.
- MongoDB: NoSQL database used for storing user credentials and saved recipes efficiently.

6.2 UI/UX TOOLS

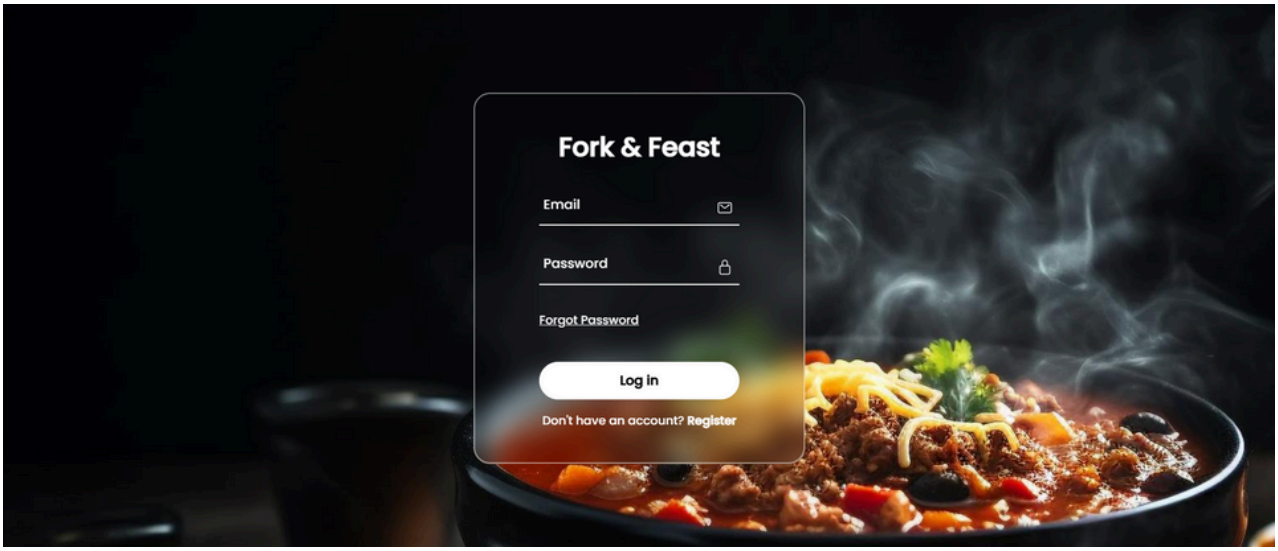
- HTML5/CSS3: For building responsive, accessible, and visually appealing frontends.
- Ionicons: Used for enhancing user interface with visually descriptive icons.
- Google Fonts (Poppins): Provided a clean and modern look to the typography.

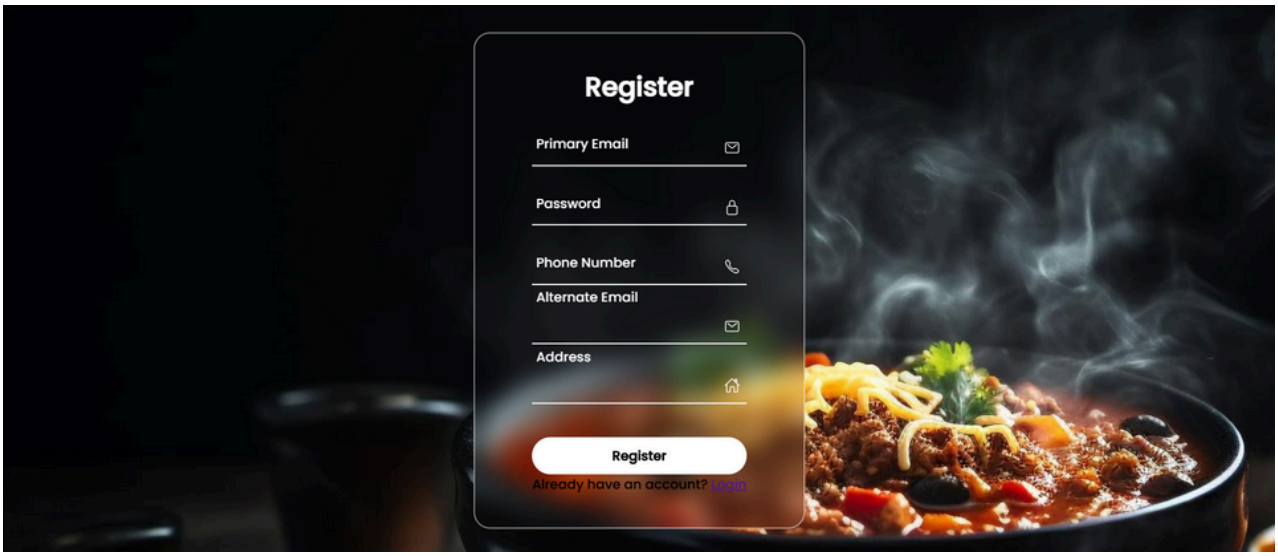
6.3 TESTING & DEBUGGING

- Postman: Used for testing API endpoints.
- MongoDB Compass: GUI tool for inspecting database entries during development.
- Browser Developer Tools: For inspecting UI elements and debugging frontend issues.

These tools helped ensure a reliable development experience and smooth debugging cycle.

SAMPLE SCREENSHOTS





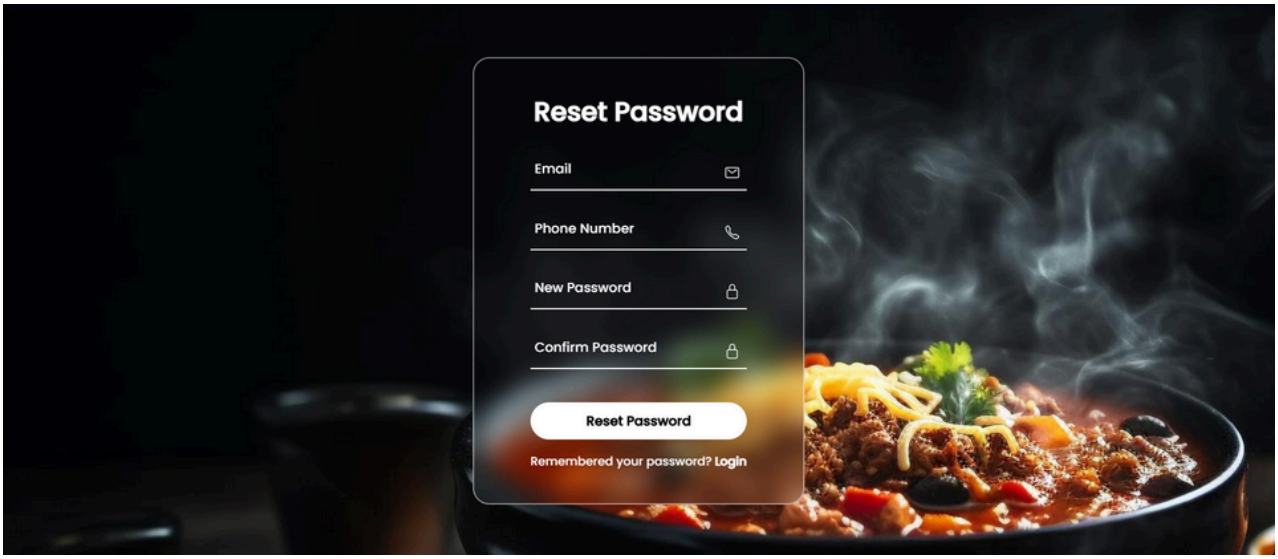


Fig 7.1, 7.2, 7.3- Login Page, Registration Page, Reset/Forgot Password Page

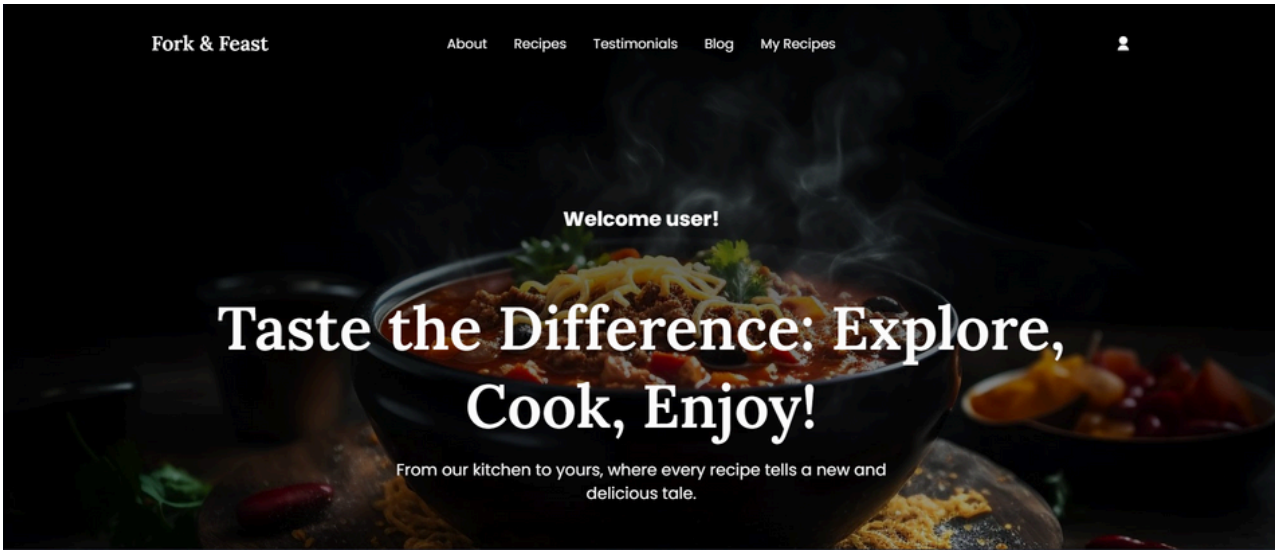


Fig 7.4- Welcome/Home Page

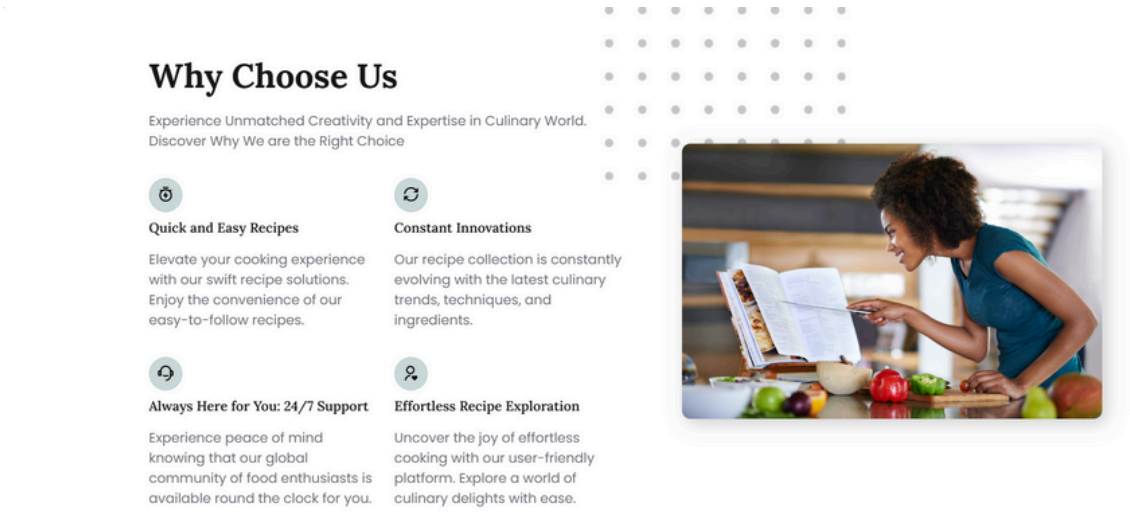


Fig 7.5- About Us Page

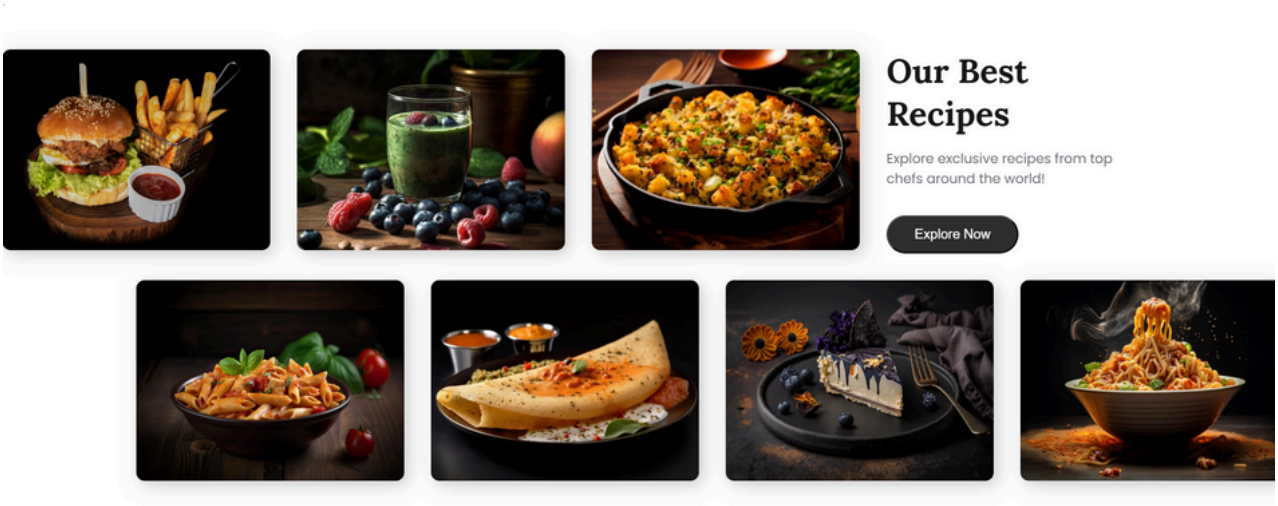


Fig 7.6- Recipes Page

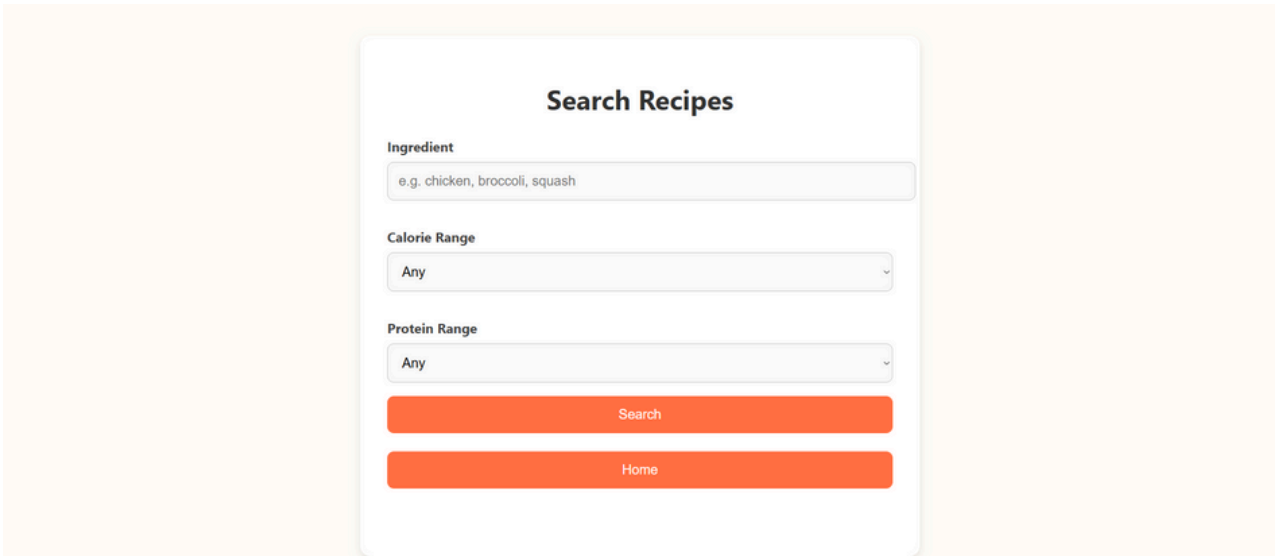


Fig 7.7- Search Recipes Page

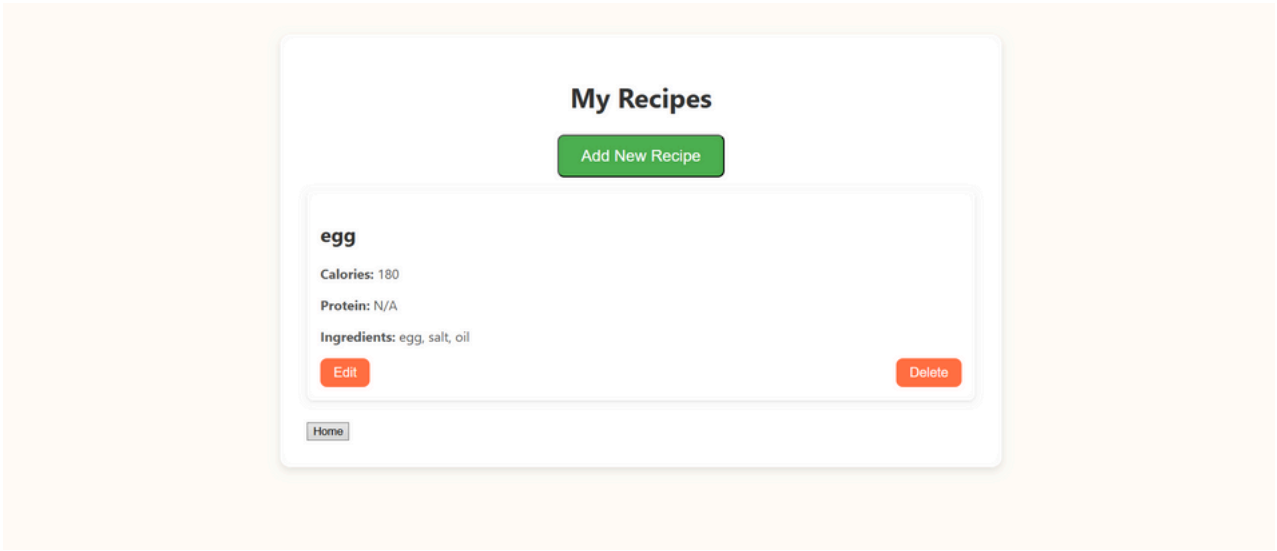


Fig 7.8- My Recipes Page

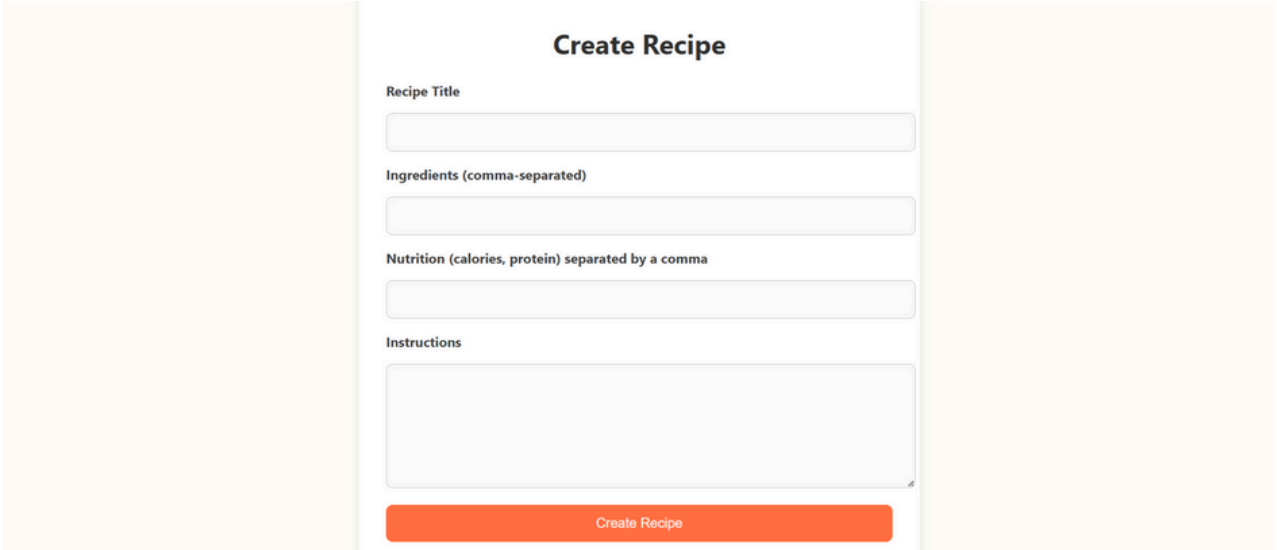


Fig 7.9- Create a Recipe Page

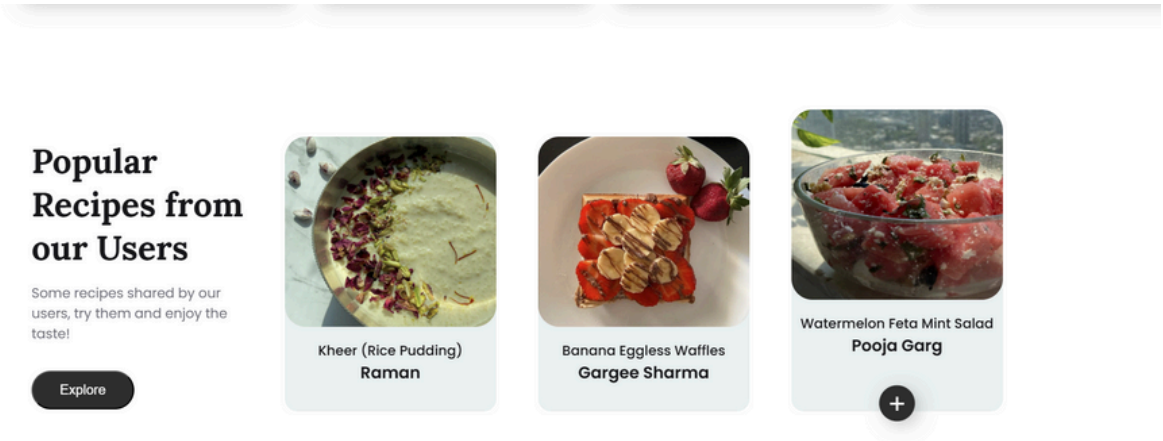


Fig 7.10- Explore Recipes Page

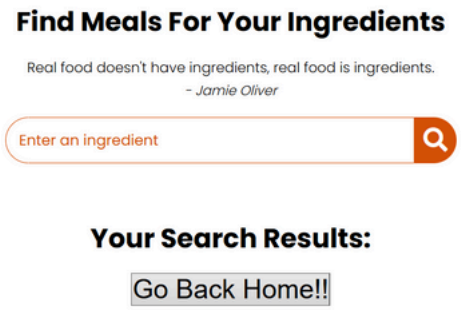


Fig 7.11- Enter Ingredients Page

Fig 7.12- Testimonials Page

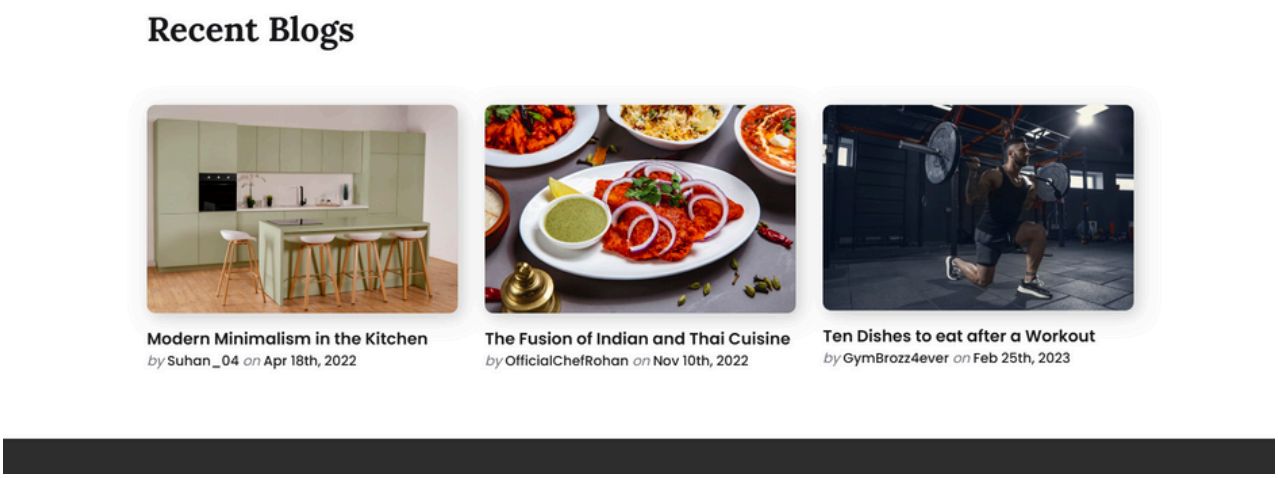


Fig 7.13- Blogs Page

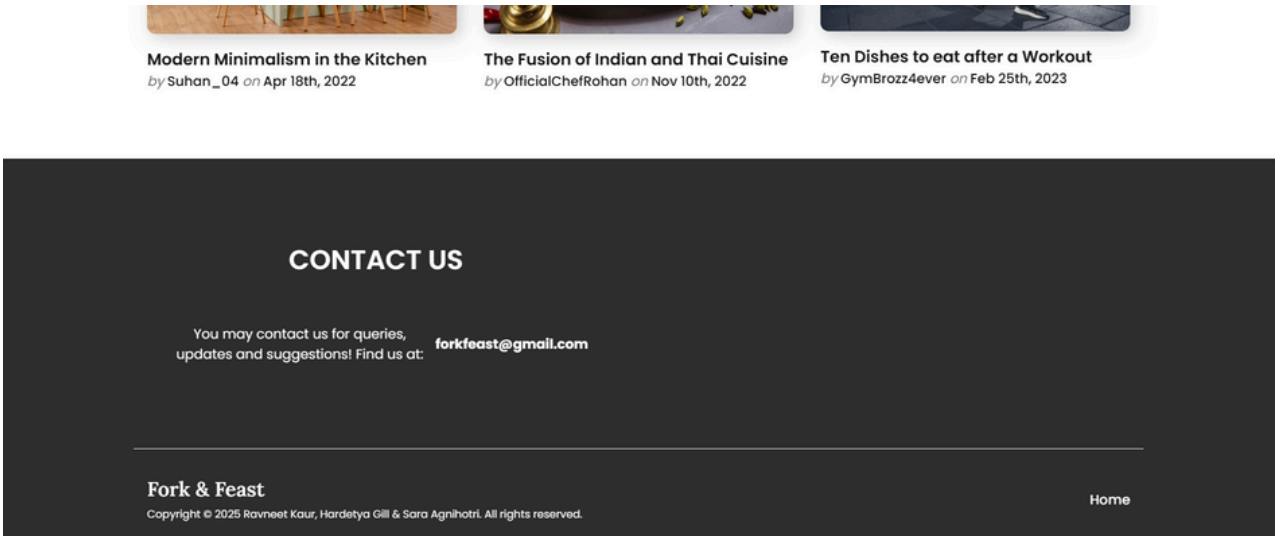


Fig 7.14- Footer Page

OUTPUT REPORTS

This project had a lot of benefits from the reports generated.

- *Improved Debugging:* The detailed test logs and error traces allow developers to replicate and resolve bugs faster.
- *Transparency*: Clear visibility of what was tested and what failed builds trust among team members and stakeholders.
- *Regression Tracking:* Test results from previous builds can be archived and compared, helping evaluate system stability over time.
- *Efficiency in Collaboration:* QA engineers, developers, and project leads can all refer to the same source of truth when discussing testing status and issues.

CONCLUSION

The Fork & Feast project successfully bridges the gap between culinary creativity and digital convenience by offering users a streamlined, intuitive, and secure platform for recipe management and discovery. With features like advanced search filters, secure user authentication, and a personalized recipe tracker, the platform addresses key pain points in modern digital recipe platforms.

Security was a top priority in this project, with careful attention given to password recovery mechanisms that match phone numbers to email identities for authentication. The inclusion of MongoDB as a backend database ensures scalability and future extensibility. Additionally, the modular design of the system enables future features like shopping lists, AI-powered meal suggestions, or integration with fitness tracking devices.

By combining thoughtful design with robust backend functionality, the platform not only meets but exceeds user expectations in both form and function. It represents a scalable solution for both casual users and health-conscious individuals aiming to take control of their dietary journey.