FOOD ORDERING MANAGEMENT SYSTEM

A MINI-PROJECT REPORT

Submitted by

CAROLINE SUJA J S   2116220701048

DEEPIGA DHARSHINI C   2116220701057

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

AUTONOMOUS, CHENNAI

NOV/DEC, 2024

BONAFIDE CERTIFICATE

Certified that this Mini Project "**FOOD ORDERING MANAGEMENT SYSTEM**" is the Bonafide work of **"DEEPIGA DHARSHINI C (2116220701057), CAROLINE SUJA J S (2116220701048)"** who carried out the project work under my supervision.

SIGNATURE

Dr. Duraimurugan

Associate 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 honorable 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 ever supporting force during my project work.

I also extend my sincere and hearty thanks to my internal guide **DR. DURAIMURUGAN** 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 **Food Ordering Management System** is a full-stack web application designed to streamline the online food ordering process for both restaurant administrators and customers. Built using **React, MongoDB, and Node.js**, the platform offers an efficient solution for managing food items, processing orders, and enhancing customer experiences. Administrators can add, update, and remove food items, view a consolidated list of products, and manage orders with real-time status updates such as "Order Processing," "Out for Delivery," and "Delivered." Customers can explore food items with detailed descriptions and images, add items to a cart, like products for personalization, and seamlessly place orders through a user-friendly checkout process.

The application features a robust backend for handling data storage and retrieval, a responsive frontend for engaging user interaction, and a scalable architecture to accommodate future enhancements. By addressing the operational challenges of restaurants and providing a convenient, enjoyable experience for customers, this project delivers a comprehensive, modern solution for the online food ordering domain.

TABLE OF CONTENTS

Chapter 1

INTRODUCTION

1.1 INTRODUCTION

The Food Ordering Management System is a comprehensive web-based application tailored to simplify the food ordering and management process for both users and administrators. With the growing demand for convenient online services, this platform caters to the needs of restaurants and customers by offering a user-friendly interface and robust backend capabilities. The portal is designed using the MERN stack, comprising MongoDB for efficient data storage, Express.js and Node.js for server-side operations, and React.js for creating an intuitive and interactive frontend.

The platform addresses key pain points in traditional food ordering systems, such as limited customization, inefficient order tracking, and outdated inventory management. It empowers administrators to have full control over food item listings, order status updates, and customer interactions, while users benefit from a seamless shopping experience with features like liking products, adding them to a cart, and placing orders effortlessly. Additionally, the portal is highly scalable, capable of supporting growing customer bases and large volumes of data, making it suitable for both small and large food businesses.

One of the standout features of the portal is its modular architecture, which allows for easy integration of additional functionalities in the future, such as payment gateways, real-time delivery tracking, and promotional offers. This makes it an ideal solution for businesses aiming to stay competitive in the rapidly evolving food service industry. The following sections of this report detail the system's hardware and software requirements, individual modules, and their impact on overall efficiency and usability.

1.2. SCOPE OF WORK

The scope of the Food Ordering Management System involves creating a user-friendly web application that serves both administrators and users, ensuring smooth food ordering and management. For administrators, the platform provides robust features such as food item management, allowing them to add new items

with details like name, cost, description, and image, as well as update or remove existing items and view a consolidated list of all products. Additionally, the order management module enables admins to receive and track customer orders in real time, update order statuses such as "Order Processing," "Out for Delivery," and "Delivered," and efficiently manage both active and past orders. On the user side, the portal offers features like product browsing, allowing users to view food items with detailed descriptions, pricing, and images, and mark their preferences by liking items. Furthermore, users can manage their cart, add items, and complete orders through a streamlined checkout process, ensuring a seamless and satisfying experience.

1.3. PROBLEM STATEMENT

The increasing demand for convenient and efficient food ordering systems highlights the need for a robust online platform that caters to both customers and restaurant administrators. This project aims to develop a Food Ordering Management System using React, MongoDB, and Node.js to address these requirements. The portal will provide administrators with tools to manage food items by adding, updating, or removing details such as cost, description, and images, while also allowing real-time order tracking and status updates like "Order Processing," "Out for Delivery," and "Delivered." For customers, the platform offers a visually engaging interface to browse food items, add favourites to a cart, like products, and seamlessly place orders. By enhancing user experience and streamlining backend operations, this solution seeks to bridge the gap between restaurants and their patrons, ensuring a smooth and enjoyable food ordering experience.

1.4 AIM AND OBJECTIVES OF THE PROJECT

The aim of this project is to develop a modern **Food Ordering Portal** using **React, Node.js, and MongoDB** to streamline food ordering and management for both administrators and customers. Administrators can efficiently manage food items, track orders in real-time, and update statuses, while customers enjoy a seamless experience with features like product browsing, personalized preferences, and a smooth checkout process. The portal leverages a responsive frontend, a robust backend, and secure data storage to ensure scalability,

reliability, and future enhancements, delivering a comprehensive solution that enhances operational efficiency and user satisfaction.

Chapter 2

SYSTEM SPECIFICATIONS

2.1 HARDWARE SPECIFICATION:

To deploy and run the Food Ordering Portal effectively, the following hardware components are recommended:

Server Requirements:

- o **Processor:** Intel Xeon or equivalent

- o **Memory:** Minimum 8 GB RAM

- o **Storage:** 256 GB SSD (expandable as needed for database storage)

- o **Network:** High-speed internet connection with a minimum of 1 Gbps bandwidth

Client Requirements:

- o **Device:** Laptop/Desktop with at least 4 GB RAM

- o **Browser:** Google Chrome, Mozilla Firefox, or Microsoft Edge (latest versions)

- o **Screen Resolution:** Minimum 1280x720 for optimal user experience

Hosting Environment: Cloud-based solutions like AWS, Azure, or Heroku for scalability and availability.

2.2 SOFTWARE SPECIFICATIONS:

The portal relies on a robust software stack to ensure high performance and flexibility:

- **Frontend:** React.js

- **Backend:** Node.js with Express.js

- **Database:** MongoDB

Development Tools:

- o Visual Studio Code for coding and debugging.

- o Postman for API testing.

Libraries and Packages:

- o Mongoose for MongoDB object modelling.

- o Axios for API requests.

- o Redux for state management in React.

Platform: The application runs on cross-platform environments, compatible with Windows, macOS, and Linux.
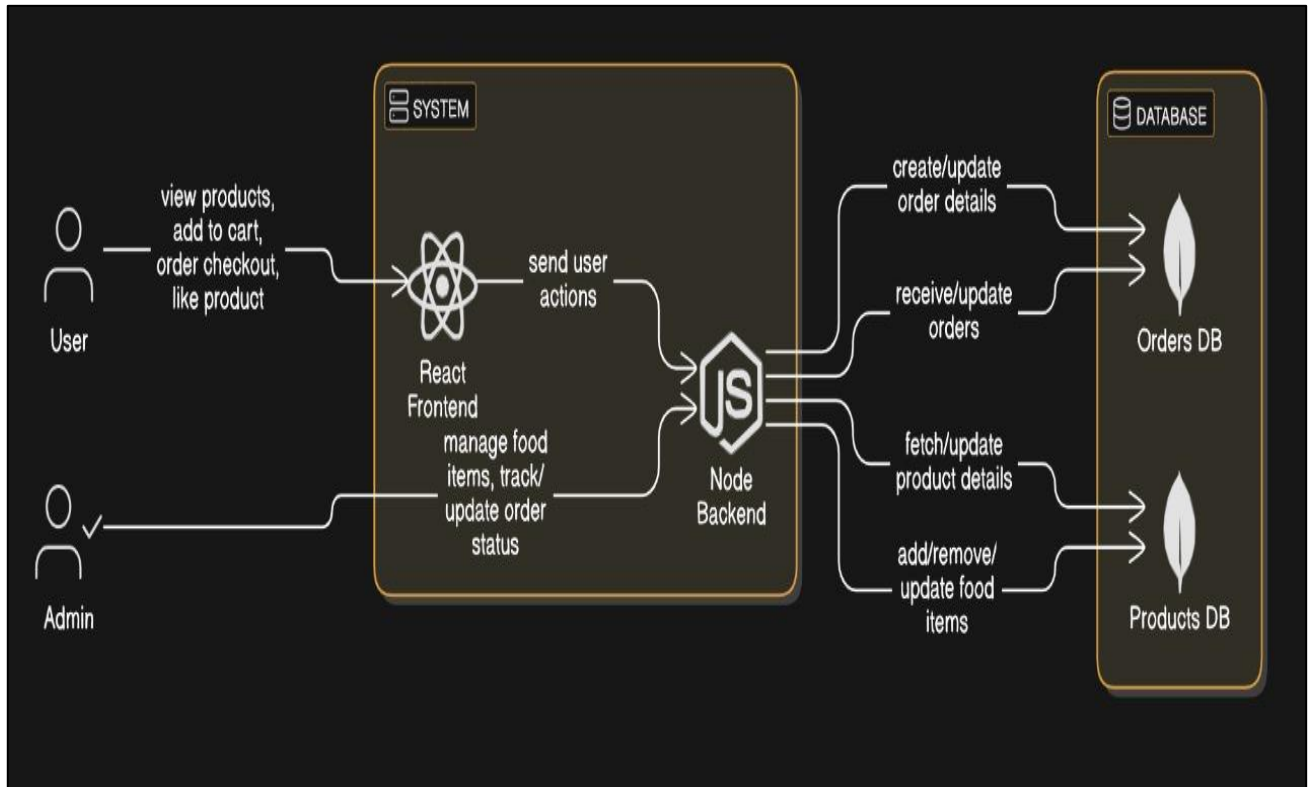
Chapter 3

ARCHITECTURE DIAGRAM

Fig.3.1 Architecture Diagram

This diagram shows the architecture of the Food Ordering Portal, highlighting interactions between **Users**, **React Frontend**, **Node.js Backend**, and **MongoDB** databases. Users and Admins perform actions like viewing products, managing items, and updating orders via the frontend, while the backend processes requests and communicates with **Products DB** and **Orders DB**. This ensures smooth functionality and data flow across the system.

Chapter 4
MODULE DESCRIPTION

3.1. Food Item Management

- **Overview:** This module allows administrators to efficiently manage the menu of food items displayed to users.

- **Features:**

    - Add new food items with cost, description, and an attractive image.
    - Update existing food items to reflect changes in price, description, or availability.
    - Remove outdated or unavailable food items from the menu.

- **Benefits:** Ensures that users always view accurate and up-to-date product information, enhancing their browsing experience.

3.2. Order Management

- **Overview:** This module streamlines the process of handling customer orders from placement to delivery.

- **Features:**

    - Admins receive real-time notifications of new orders.
    - Track the status of each order as it progresses through various stages.
    - Update order statuses, such as "Order Processing," "Out for Delivery," and "Delivered."

- **Benefits:** Helps businesses maintain transparency and keep customers informed, leading to improved satisfaction.

3.3. User Account Management

- **Overview:** Handles user registration and account-related functionalities.

- **Features:**

    - Users can register using their email or social media accounts.

- o Edit personal details like name, address, and contact information.
- o View order history and track ongoing orders.

- **Benefits:** Provides a personalized experience for users and ensures secure access to their accounts.

3.4. Cart Management

- **Overview:** Enables users to create a temporary collection of items they wish to purchase.

- **Features:**

  - o Add or remove items from the cart.
  - o Automatically calculate the total price based on selected items.
  - o Retain cart items until the user checks out or empties the cart.

- **Benefits:** Offers flexibility for users to modify their selection before finalizing their order.

3.5. Search and Filter Options

- **Overview:** Provides users with the ability to find specific items quickly.

- **Features:**

  - o Search for products using keywords.
  - o Apply filters based on categories, price ranges, or ratings.
  - o Sort results by popularity, price, or customer ratings.

- **Benefits:** Saves time and improves the overall user experience by simplifying the browsing process.

3.6. Notification System

- **Overview:** Keeps users and admins informed about critical updates.

- **Features:**

  - o Notify users of successful order placement and status changes.
  - o Alert admins about new orders and stock shortages.
  - o Provide promotional notifications to users about discounts or new arrivals.

- **Benefits:** Ensures timely communication, fostering trust and engagement.

3.7. Feedback and Ratings

- **Overview:** Allows users to share their experiences and rate food items.

- **Features:**
    - Submit reviews and ratings for individual food items.
    - View aggregate ratings for each item to aid decision-making.
    - Enable admins to analyze feedback for quality improvement.

- **Benefits:** Builds a transparent ecosystem where users feel valued and admins gain actionable insights.

SYSTEM DESIGN

5.1 USE CASE DIAGRAM

FOOD ORDERING PORTAL

Login

View Menu

Select Item

Add Items

Add to Cart

Delete Items

Add Favourites

Update Items

Confirm Order

View Orders

Log Out

Track
Order Status
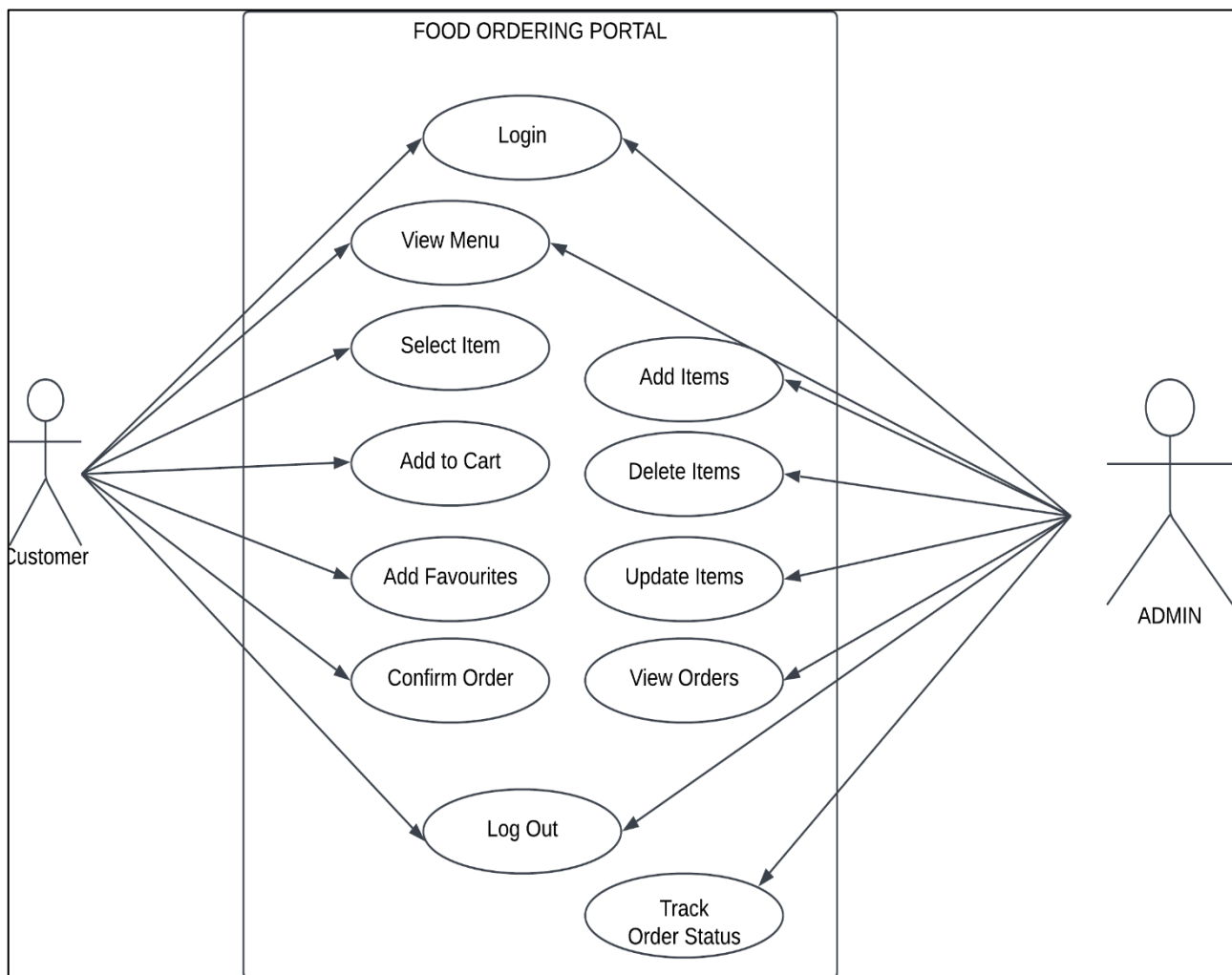
Customer

ADMIN

Fig.5.1 Use Case Diagram

This diagram illustrates the interactions between **Users** (Customer and Admin) and the **System**, showcasing key functionalities like login, menu viewing, order management, and administrative tasks such as adding or updating food items.

5.2 ENTITY-RELATIONSHIP DIAGRAM
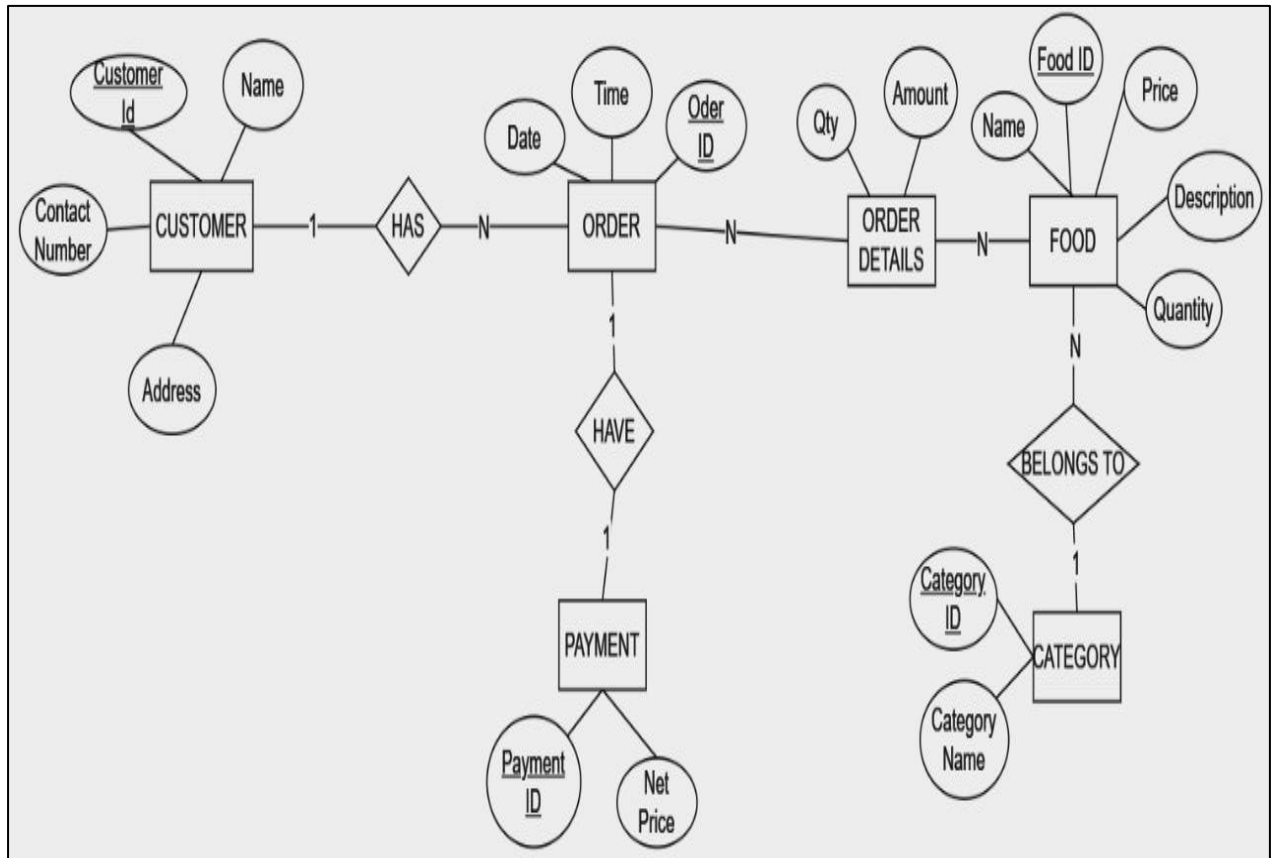
**Fig.5.2 Entity-Relationship
Diagram**

The ER diagram represents the database structure of the Food Ordering Portal, connecting entities like **Customer**, **Order**, **Payment**, **Food**, and **Category**. Customers place orders linked to order details and payments, while food items are organized into categories. This design supports efficient data handling for orders and product management.

5.3 DATA-FLOW DIAGRAM
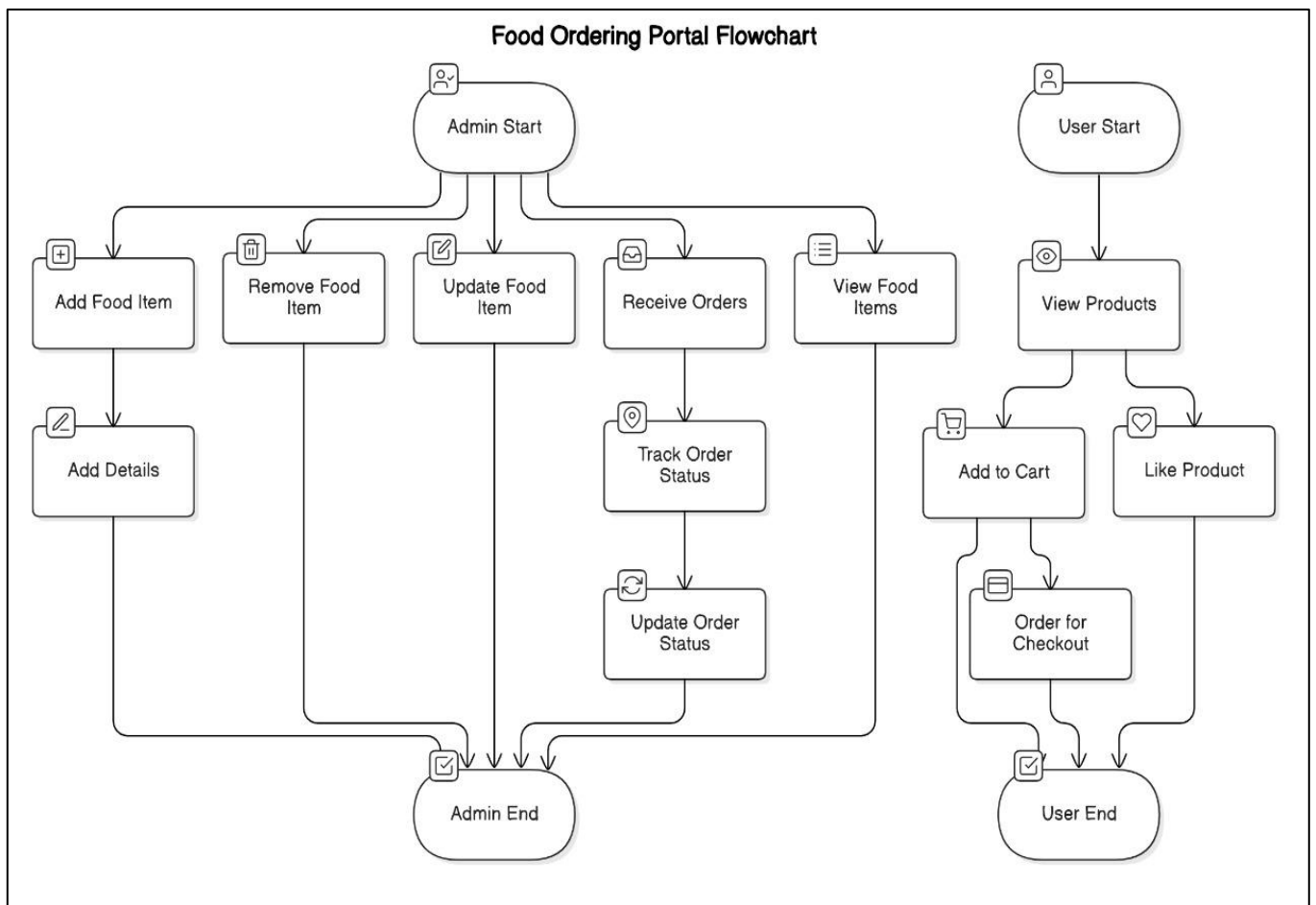
Fig.5.3 Data-Flow Diagram

The DFD outlines the flow of data between different modules of the food ordering portal, demonstrating processes like food item management, order tracking, and cart operations for both Admin and User roles.

5.4 ACTIVITY DIAGRAM
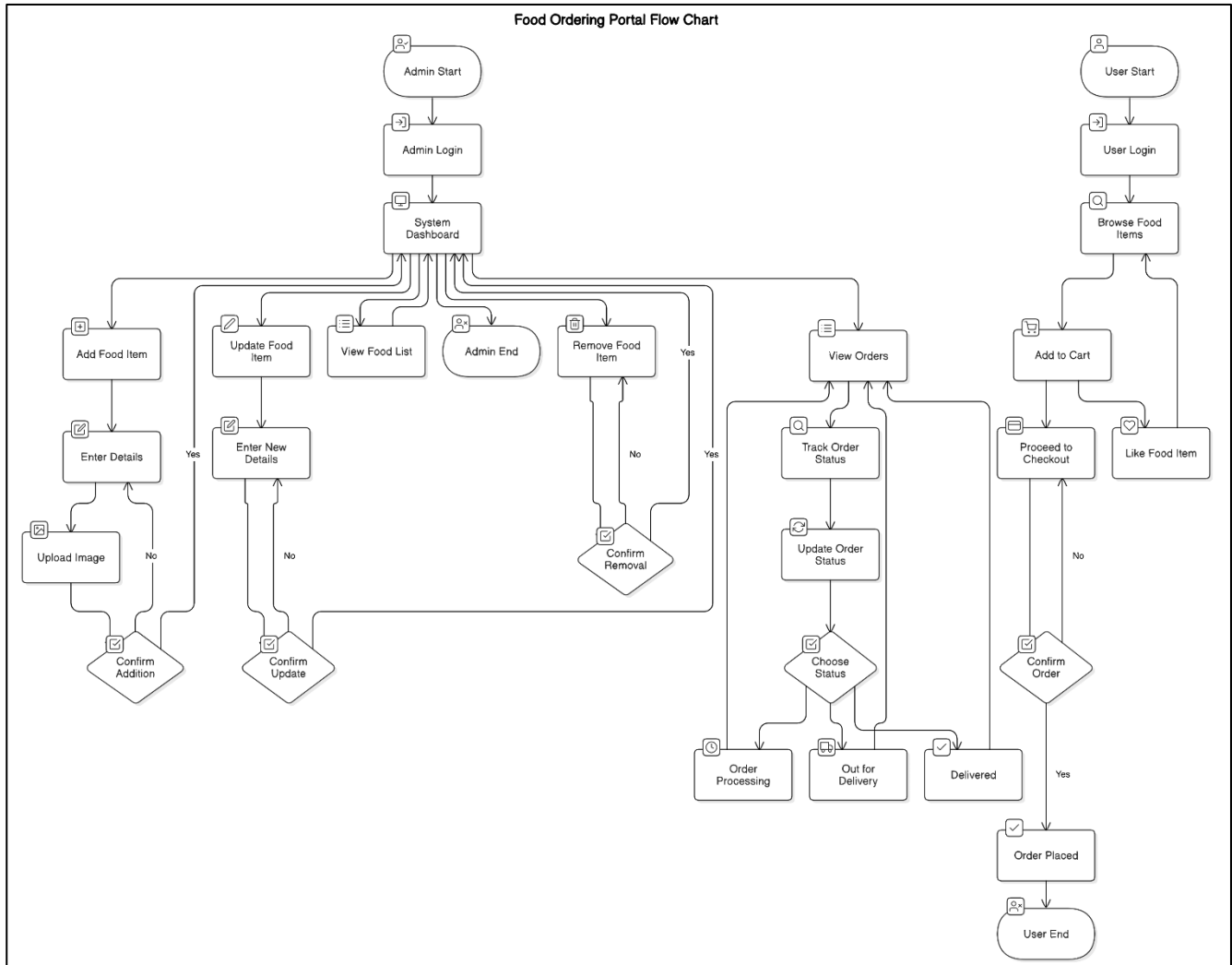
Fig.5.4 Activity Diagram

This diagram shows the workflows for Admin and User roles in the Food Ordering Portal. Admin manages food items and order statuses, while Users browse products, add to cart, and place orders. It outlines task progression and interactions.

Chapter 6

IMPLEMENTATION SCREENSHOTS

6.1 ADMIN

Fig.6.1.1 Add Item Page

The Add Item page allows administrators to add new products by providing
details such as the product image, name, description, category, and price.

Fig.6.1.2 List Items Page

The List Items page displays a comprehensive list of all added products,
including their images, names, descriptions, categories, and prices, for
easy management.
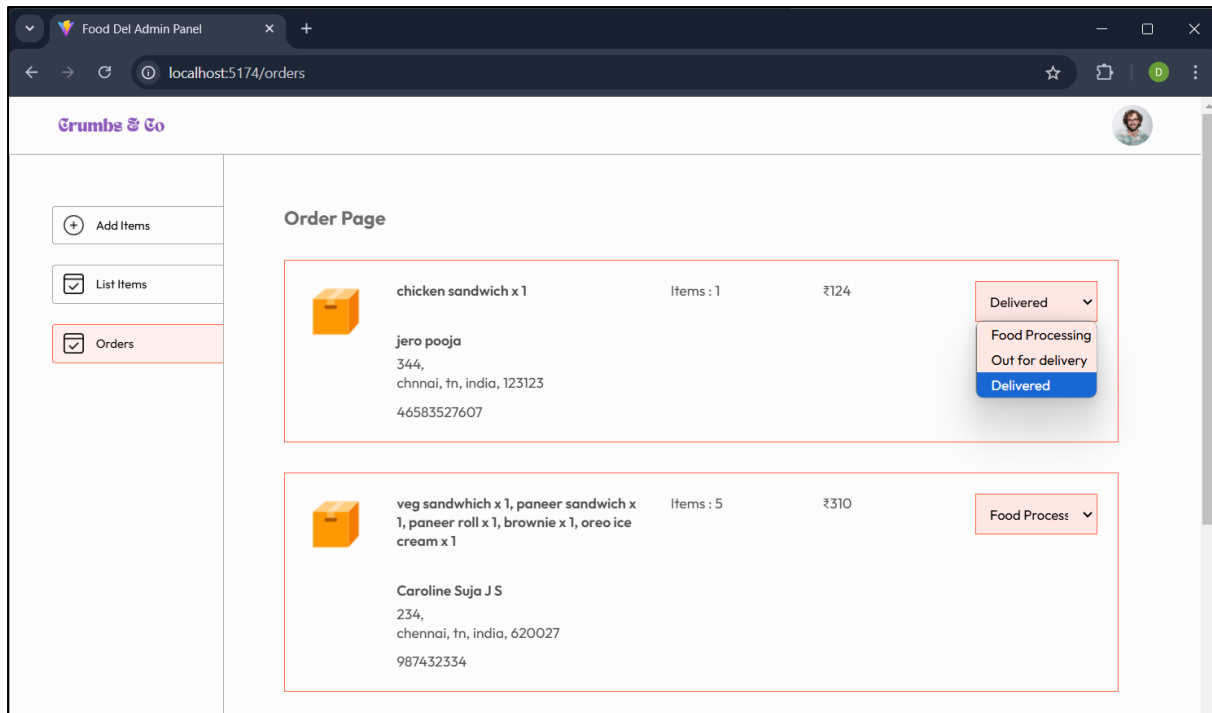
Fig.6.1.3 Orders Page

The Orders Page provides detailed information about customer orders, including the items purchased and customer details, along with the option to edit and update the status of each order.

6.2 USER INTERFACE:
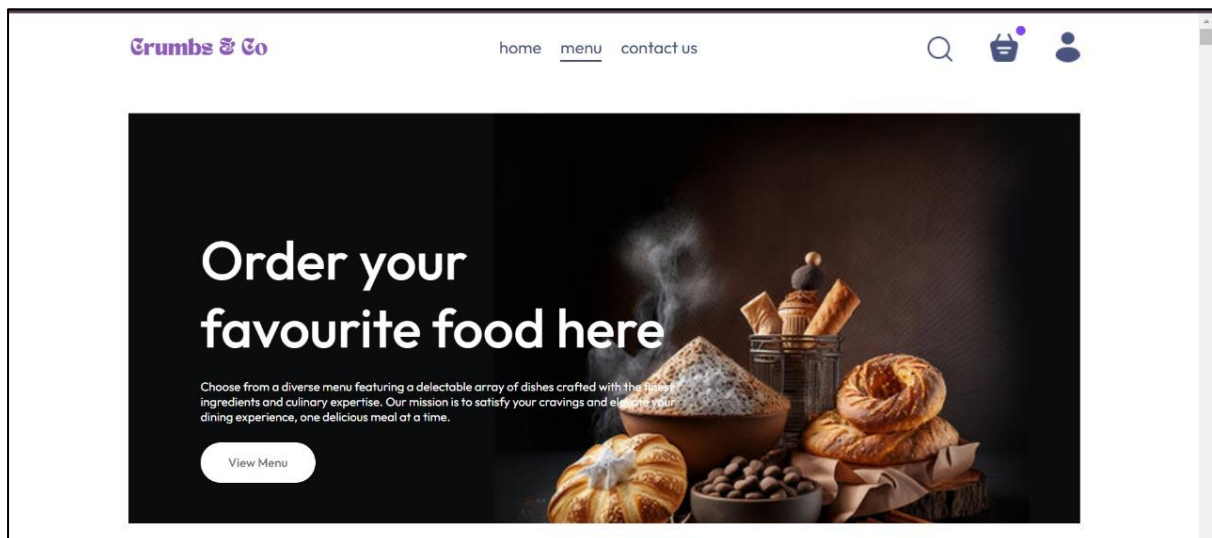
Fig.6.2.1 Home Page

The Home Page acts as the central hub, offering intuitive navigation and an engaging overview of the platform's features.
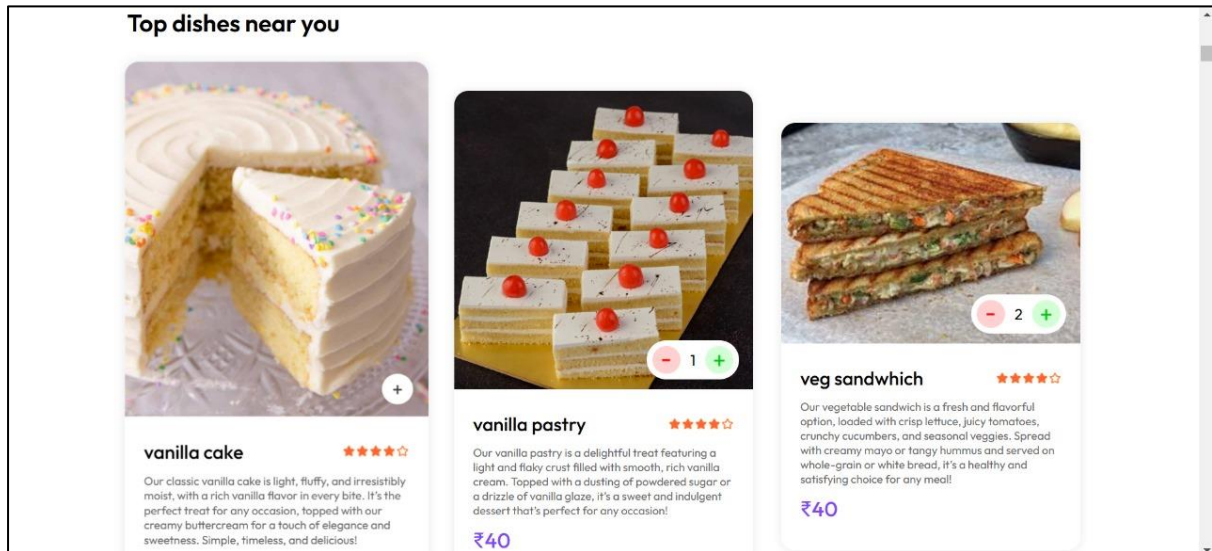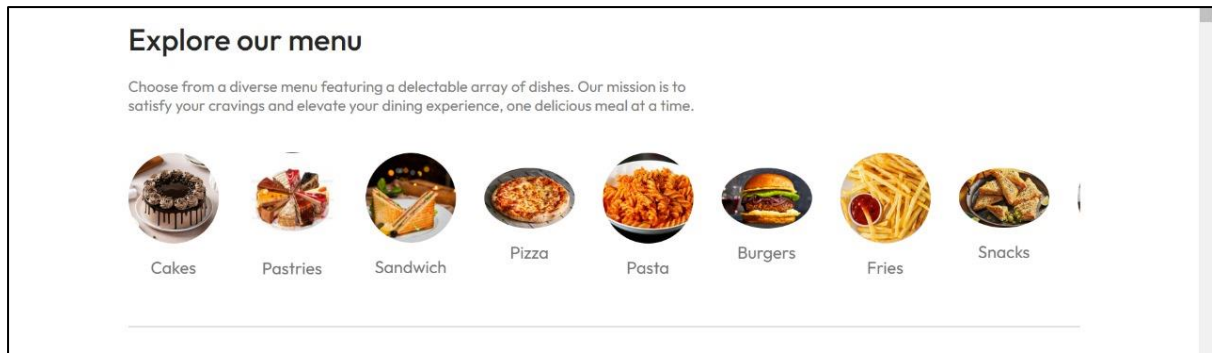
Fig.6.2.2 Menu Page

The Menu Page lists out the available food from the restaurant with their price, description and ratings.
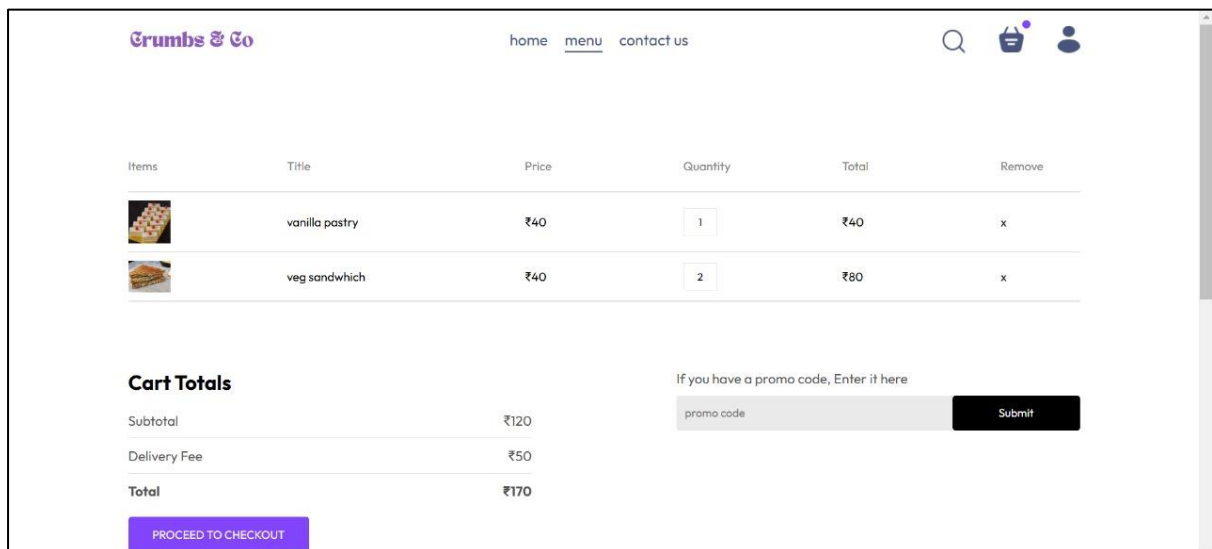

Fig.6.2.3 Cart Page

The Cart Page displays the selected menu items, allowing users to review, modify quantities, and proceed to checkout.

Fig.6.2.4 Delivery Information Page

The Delivery Information page collects and displays user details such as name, address, mobile number, and other contact information required for order delivery.

Fig.6.2.5 Orders Page

The Orders Page displays a list of user orders along with their current status, such as "Order Processing," "Out for Delivery," or "Delivered."
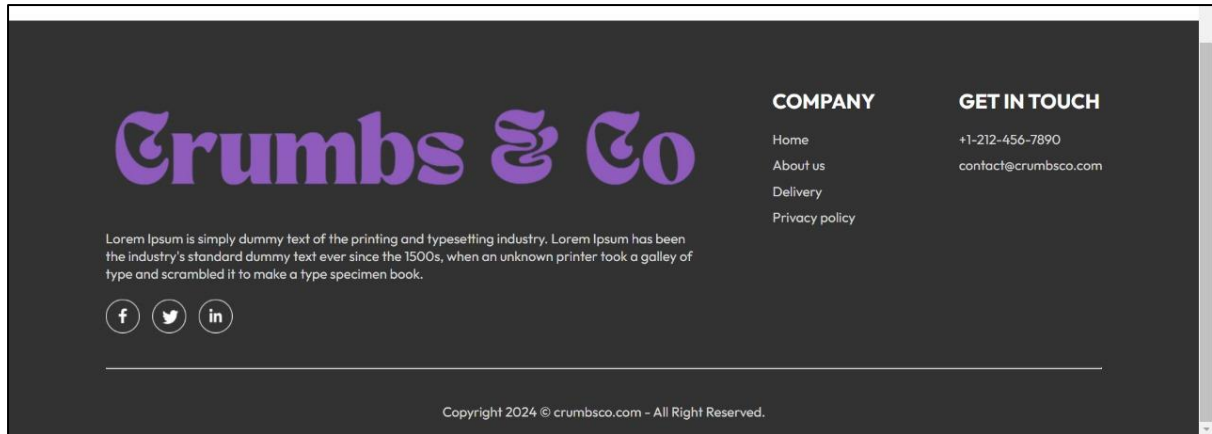
Fig.6.2.6 Contacts Page

The Contact Us page provides detailed information about the shop, including its address, contact number, email, and operating hours, along with a form for customer inquiries.

Chapter 7
RESULT AND CONCLUSION

The Food Ordering Portal project has been successfully completed, delivering an efficient and user-friendly platform for managing food items and facilitating seamless online food ordering. The key features outlined in the requirements, including admin capabilities for managing food items and orders, and user functionalities like browsing products, adding to cart, and placing orders, have been implemented effectively, ensuring a comprehensive solution that meets the needs of both administrators and customers.

7.1. CONCLUSION

The Food Ordering Portal efficiently streamlines the process of managing food items and placing orders, offering distinct functionalities for both Admin and User roles. The admin side allows for the seamless management of food items, including adding, updating, and removing items with relevant details like cost, description, and images. Admins can also receive and monitor orders, track their statuses, and update them to reflect progress such as "Order Processing," "Out for Delivery," or "Delivered." This ensures smooth operations and timely delivery of services.

The User side provides an intuitive interface for browsing products, adding items to the cart, liking products, and proceeding to checkout. Users can easily explore food options and place orders, enhancing the convenience and overall customer experience. The portal effectively bridges the gap between customers and administrators, creating a unified system that is efficient, user-friendly, and responsive.

In conclusion, the Food Ordering Portal demonstrates how modern web technologies like React, Node.js, and MongoDB can be utilized to build an interactive, dynamic, and functional system. By enabling effective communication between users and administrators, the system achieves its goal of simplifying the food ordering process. This project highlights the potential of integrating full-stack development with streamlined workflows to create practical and impactful applications in the domain of e-commerce.

Chapter 8

REFERENCES

8.1 DOCUMENTATIONS

- React Documentation: https://reactjs.org/docs/getting-started.html

- Node.js Documentation: https://nodejs.org/en/docs/

- MongoDB Documentation: https://www.mongodb.com/docs/

- Express.js Guide: https://expressjs.com/en/starter/guide.html

8.2 ONLINE TUTORIALS

- MERN Stack Tutorials: Online resources and blogs.