NATIONAL INSTITUTE OF INDUSTRIAL TRAINING

PROJECT TITLE: FOOD ORDERING SYSTEM

COLLEGE NAME: RCC INSTITUTE OF INFORMATION TECHNOLOGY

NAME : DIBYA GHOSH

ROLL NO: BCA2023023

COURSE: BCA

SEMESTER: 5TH

YEAR: 3RD

SESSION: 2025

DATE: 7/10/2025

ACKNOWLEDGEMENT

I would like to express my deepest gratitude to Teacher, whose valuable guidance, constant encouragement, and constructive suggestions were instrumental throughout the completion of this project. Their expertise and continuous support helped me understand the practical aspects of the subject and provided direction at every stage of the project.

I am profoundly thankful to National Institute of Industrial Training , where I had the opportunity to complete my internship and conduct this project. The institute provided me with the necessary facilities, technical resources, and a practical learning environment, which allowed me to apply theoretical knowledge in real-world scenarios. The hands-on experience gained here has greatly enhanced my understanding of the subject and improved my practical skills.

I am also grateful to my college, RCC Institute of Information Technology, for allowing me to participate in this internship and providing the academic support required to successfully complete this project. Their encouragement and guidance ensured that I stayed motivated and focused throughout the project duration.

Furthermore, I would like to extend my sincere thanks to my family and friends for their unwavering support and motivation. Their patience and understanding allowed me to dedicate the necessary time and effort to complete this project efficiently and effectively.

Finally, I would like to acknowledge everyone who directly or indirectly helped me during this internship project. Their support, advice, and encouragement have been invaluable in making this project a successful and enriching learning experience.

ABSTRACT

The Food Ordering System is a web-based application developed as part of an internship project at National Institute of Industrial Training. This system aims to simplify the process of ordering food online, providing convenience to customers and efficiency for restaurants. Customers can browse the menu, select items, add them to a cart, and place orders seamlessly. The system also provides real-time order tracking and confirmation, ensuring a smooth and reliable service experience.

For the restaurant, the system offers an organized method to manage menu items, track customer orders, and update order statuses promptly. By automating manual processes, it reduces errors, saves time, and enhances operational efficiency.

This project integrates frontend design, backend programming, and database management to create a practical and functional solution. The experience gained during this internship has strengthened my understanding of real-world application development and demonstrated the importance of teamwork, problem-solving, and technical skills.

Future improvements can include online payment integration, customer feedback modules, and mobile application support to further enhance usability and reach.

INTRODUCTION

The Food Ordering System is a modern web-based application designed to simplify and streamline the process of ordering food online. This system allows customers to browse restaurant menus, select their preferred items, add them to a cart, place orders, and track their status in real-time. The application is developed using Next.js for the frontend, JavaScript for interactivity, CSS for styling, and MongoDB for database management.

This project was carried out as part of an internship at National Institute of Industrial Training, which provided hands-on experience in real-world web development. The use of Next.js ensures a fast, responsive, and SEO-friendly interface, while JavaScript powers dynamic functionalities such as cart management, order updates, and form validation. CSS is applied to create a clean and modern design, enhancing the overall user experience. MongoDB, a NoSQL database, efficiently stores user details, menu items, orders, and transaction history, providing a scalable backend solution.

The main objective of this system is to reduce manual work, minimize errors in order processing, and improve service efficiency for both customers and restaurants. Customers benefit from a user-friendly interface to place and manage orders conveniently, while restaurant staff can easily manage menu items, update order statuses, and track orders through the admin panel.

Furthermore, this project lays the foundation for future enhancements, such as online payment integration, customer feedback modules, promotional offers, and mobile-friendly versions. Overall, the Food Ordering System demonstrates the practical application of modern web technologies and database management to build a functional, scalable, and efficient platform for online food ordering.

OBJECTIVES

The primary objectives of the Food Ordering System project are outlined as follows:

Streamline the Food Ordering Process:

To provide customers with a seamless and efficient platform to browse menus, select dishes, place orders, and track them in real-time, thereby enhancing convenience and accessibility.

Enhance User Experience:

To develop a responsive, interactive, and visually appealing interface using Next.js, JavaScript, and CSS, ensuring a smooth and intuitive experience for users across devices.

Optimize Restaurant Operations:

To enable restaurant staff to manage menu items, process orders, and update order statuses effectively through a centralized admin panel, improving operational efficiency.

Reduce Errors and Manual Work:

To minimize errors associated with manual order-taking and record-keeping by automating key processes, thereby ensuring accuracy and reliability in order management.

Implement Robust Database Management:

To utilize MongoDB for storing and managing user data, menu items, orders, and transaction history securely, supporting scalability and efficient data retrieval.

Support Future Enhancements:

To design a flexible and scalable system that can accommodate future upgrades such as online payment integration, customer feedback modules, promotional offers, and mobile application support.

SCOPE OF THE PROJECT

The Food Ordering System is designed to provide a comprehensive solution for online food ordering, catering to both customers and restaurant management. The system enables customers to browse menus, select items, place orders, and track them in real-time. For restaurants, it offers an efficient way to manage menu items, monitor incoming orders, and update order statuses.

The project has the potential to be scaled further by integrating features such as online payment gateways, promotional offers, loyalty programs, and mobile application support. By adopting modern web technologies like Next.js, JavaScript, CSS, and MongoDB, the system ensures responsiveness, interactivity, and secure data management.

BENEFITS OF THE PROJECT

Convenience for Customers:

Customers can place orders from anywhere, at any time, without the need to visit the restaurant physically.

Efficiency for Restaurants:

The system allows restaurants to manage menu items, process orders efficiently, and reduce waiting times for customers.

Error Reduction:

Automation reduces manual errors in order processing, billing, and record-keeping.

Time and Resource Saving:

Both customers and restaurant staff save valuable time by streamlining the ordering and management processes.

Scalability:

The system is designed to accommodate future enhancements, including online payments, feedback collection, promotions, and mobile compatibility.

Improved Customer Satisfaction:

By providing a fast, reliable, and user-friendly interface, the system enhances overall customer experience and satisfaction.

SYSTEM REQUIREMENTS

Hardware Requirements:

To run the Food Ordering System efficiently, the following hardware specifications are recommended:

Processor: Intel Core i3 or higher

RAM: 4 GB or higher

Storage: Minimum 100 GB free disk space

Display: 1366×768 resolution or higher

Internet Connection: Stable broadband or mobile data for accessing the web application

Software Requirements:

The system is developed using modern web technologies and requires the following software components:

Operating System: Windows 10 Or 11

Web Browser: Google Chrome, Mozilla Firefox, Microsoft Edge, or any modern browser

Frontend Framework: Next.js (React)

Programming Language: JavaScript

Styling: CSS

Backend / Database: MongoDB for data storage

Development Environment: Visual Studio Code or any compatible code editor

Node.js: Required for running Next.js applications

Additional Requirements:

Internet connection for live deployment and database connectivity

Access to MongoDB Atlas (cloud database) or local MongoDB server for testing

Basic knowledge of Git for version control (optional but recommended)

PROJECT MODULES

The Food Ordering System is divided into several modules to ensure smooth functionality and organized development. Each module is designed to handle a specific part of the system, making it easier to manage and maintain. The main modules are described below:

User Module:

Allows customers to register, login, and manage their profile.

Customers can browse the menu, add items to the cart, place orders, and track their order status.

Includes features like search, filtering, and order history.

Admin Module:

Enables restaurant staff to manage menu items, including adding, updating, or removing dishes.

Tracks incoming orders and updates order statuses (e.g., preparing, ready, delivered).

Generates basic reports of orders and sales.

Cart and Order Management Module:

Allows users to add items to the cart, update quantities, and remove items.

Handles order placement, calculation of total price, and order confirmation.

Sends real-time updates to both customers and admin.

Database Module:

Uses MongoDB to store all data including user information, menu items, orders, and transaction history.

Ensures secure and efficient data storage and retrieval.

Frontend Module:

Developed using Next.js and JavaScript, providing a responsive and interactive user interface.

Styled with CSS to ensure a clean and modern look.

Ensures smooth navigation, forms validation, and interactivity across the website.

Optional / Future Modules:

Payment Integration Module: Allows users to pay online securely.

Feedback & Rating Module: Lets customers give reviews and rate their experience.

Promotional Module: Provides discounts, offers, and coupons for users.

USE CASE FOR FOOD ORDERING SYSTEM

The Food Ordering System is designed to simplify the online food ordering process for both customers and restaurant staff. It provides a user-friendly interface, smooth navigation, and efficient order management. The system mainly involves two types of users: Customer and Admin.

Actors

Customer/User: A person who uses the system to register, browse menu items, place orders, and track delivery.

Admin/Restaurant Staff: A person who manages menu items, tracks incoming orders, updates order status, and maintains records.

Use Case Description

Customer Side

1. Register / Login:

The customer creates a new account or logs in with valid credentials.

2. Browse Menu:

The customer views all available food items, categories, and prices.

3. Add to Cart:

The customer selects items, adjusts quantities, and adds them to the shopping cart.

4. Place Order:

The customer reviews their cart, provides delivery details, and confirms the order.

5. Track Order:

The customer can track the order status — Pending → Preparing → Ready → Delivered.

6. Order History:

Customers can view past orders and reorder items easily.

7. Logout:

The customer safely exits from the system.

Customer Flow

Start

↓

Login / Register

↓

Browse Menu

↓

Add to Cart

↓

Place Order

↓

Track Order

↓

Order History

↓

Logout

Admin Side

1. Login:

The admin securely logs in to access the dashboard.

2. Manage Menu:

Add, edit, or remove food items and update pricing.

3. View Orders:

Admin monitors incoming orders from customers.

4. Update Order Status:

Change order status to Preparing, Ready, or Delivered.

5. Generate Reports:

Admin checks order details, total sales, and performance statistics.

6. Logout:

Safely exits from the system.

Admin Flow

Start

↓

Login

↓

Manage Menu

↓

View Orders

↓

Update Status

↓

Generate Reports

↓

Logout

SYSTEM DESIGN

System design is the process of planning the overall architecture, components, and data flow of the system. It helps in understanding how different parts of the system work together to achieve the required functionality.

The Food Ordering System is designed using a modular and layered architecture, which makes it easier to maintain, scale,

and update. The system mainly consists of three layers: Frontend (Client Side), Backend (Server Side), and Database Layer.

System Architecture

Frontend (Client Side):

The frontend is developed using Next.js, HTML, CSS, and JavaScript.

It provides a responsive and user-friendly interface for customers and admins.

Users can browse menus, add food items to the cart, and place orders.

Admins can manage menu items and view orders through their dashboard.

Backend (Server Side):

The backend is developed using Node.js and Express.js.

It handles all business logic, server requests, and responses between the frontend and database.

The backend ensures that the correct data is fetched, stored, and displayed to the user.

API routes are created to handle user login, registration, menu management, and order tracking.

Database Layer:

The database is designed using MongoDB.

It stores all the important information such as user details, food items, orders, and payment history.

The database is connected to the backend through the Mongoose library for easy data handling.

Data Flow Design (DFD – Level 1)

You can insert this simple Data Flow Diagram (draw using Word → Insert → Shapes → Flowchart):

User

↓

Frontend Interface

↓

Backend Server (Node.js)

↓

Database (MongoDB)

↑

Response Back to User

IMPLEMENTATION

Implementation is the process of converting the project design into an actual working system. In this phase, all the planned modules, designs, and functionalities are developed and integrated using suitable technologies.

The Food Ordering System was implemented using Next.js, JavaScript, CSS, and MongoDB. These technologies together help in building a dynamic, responsive, and efficient web application that simplifies the food ordering process.

Technologies Used

Frontend Implementation (Client Side)

The frontend was developed using Next.js, HTML, CSS, and JavaScript.

Next.js was used for building dynamic web pages and handling routing easily.

CSS provided a responsive and attractive design for all screens (desktop, tablet, mobile).

Users can browse menus, view food items, and place orders smoothly through the interface.

Frontend Features:

- User-friendly interface
- Real-time updates using React components in Next.js
- Responsive design with CSS Flexbox and Grid

Backend Implementation (Server Side)

The backend was developed using Node.js and Express.js.

The backend handles all server-side logic, including user authentication, menu management, and order processing.

RESTful APIs were built to connect the frontend with the database.

Backend Features:

- API routes for Login, Register and Add Order
- Secure data transfer between client and server
- Validation of all input data

Database Implementation

MongoDB was used as the database for storing all system data.

The connection between the backend and the database was established using Mongoose, a MongoDB Object Data Modeling (ODM) library.

Data such as user information, food details, and order records are securely stored in collections.

Database Collections:

- users — stores user details like name, email, and password
- menuItems — stores food item details like name, price, and category
- orders — stores order details, status, and timestamps

Integration

- After individual modules (Frontend, Backend, and Database) were implemented, they were integrated together.
- The frontend communicates with the backend using API calls.
- The backend processes user requests and interacts with the database.

- The results are sent back to the frontend in JSON format for display to the user.
- This integration ensures smooth real-time communication between all layers of the system.

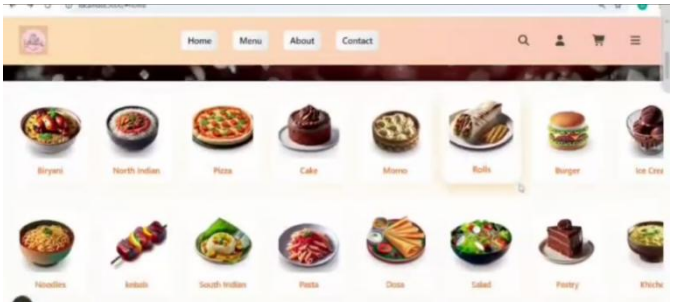
Challenges Faced

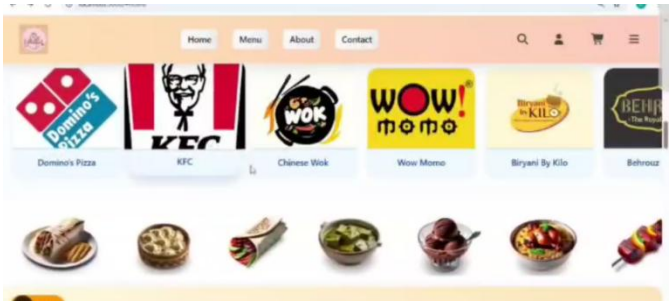
- Ensuring data synchronization between frontend and backend during real-time order updates.
- Maintaining responsive design for different devices.
- Handling multiple user sessions securely.

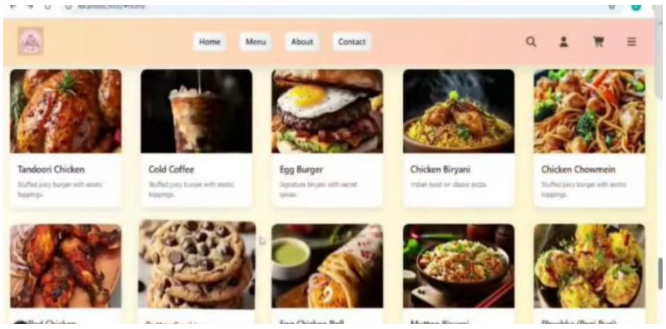
Final Outcome

- After implementation, the system was fully functional and met all project objectives.
- Customers could easily register, log in, browse menus, and order food online.
- Admins could manage menu items and track orders efficiently.
- The system was responsive, secure, and user-friendly.

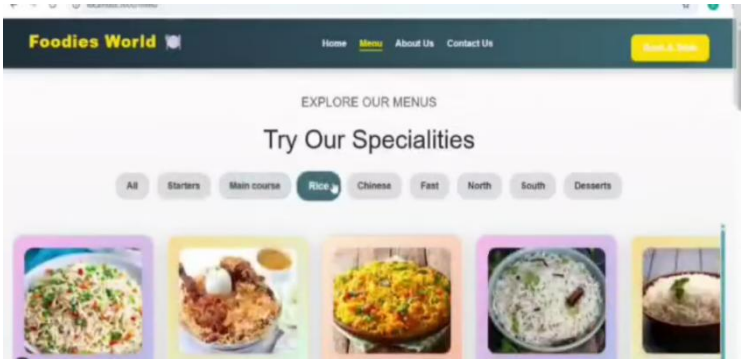
HOME

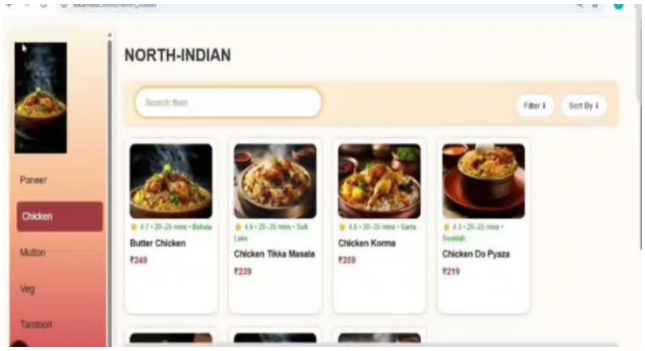




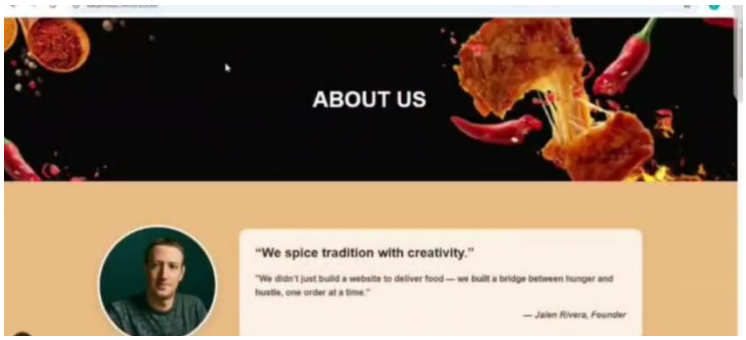


MENU

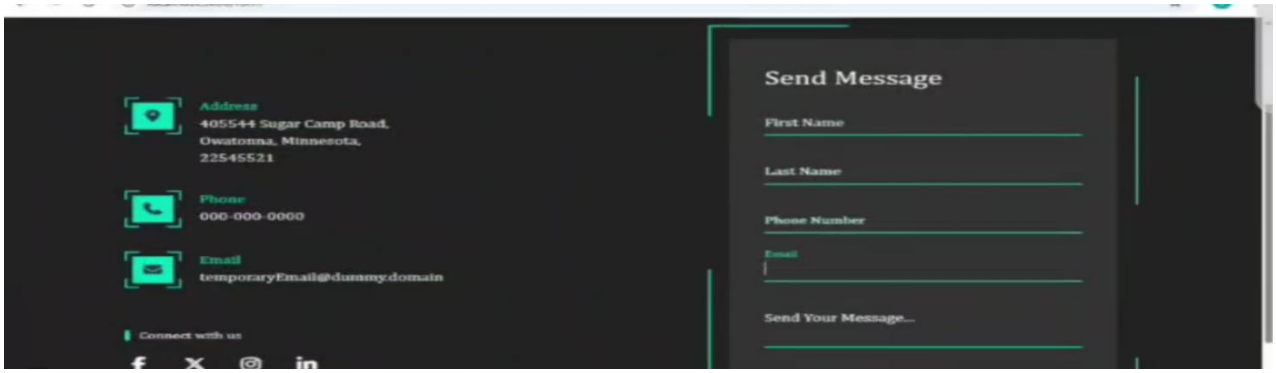




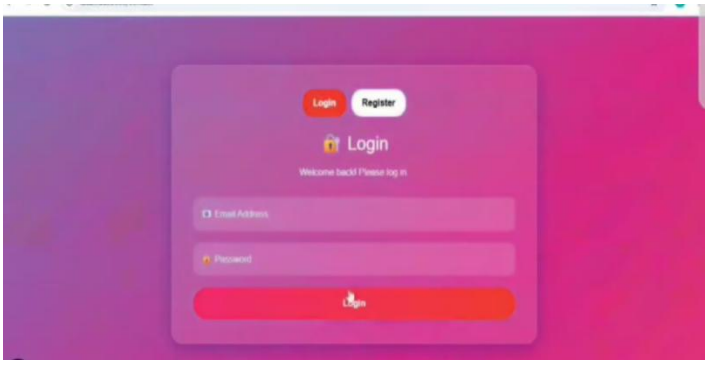
ABOUT

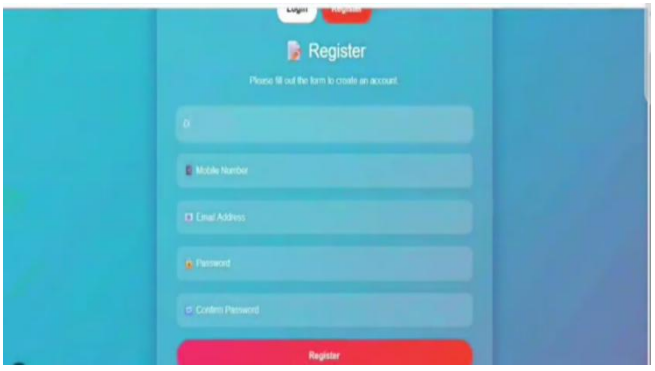


CONTACT US

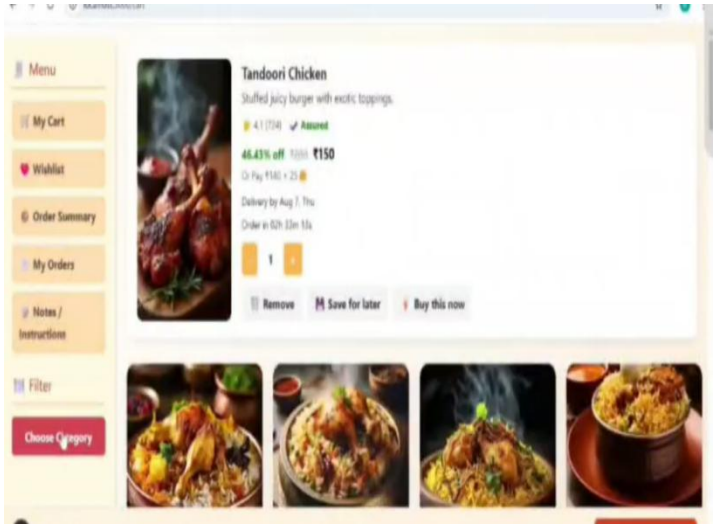


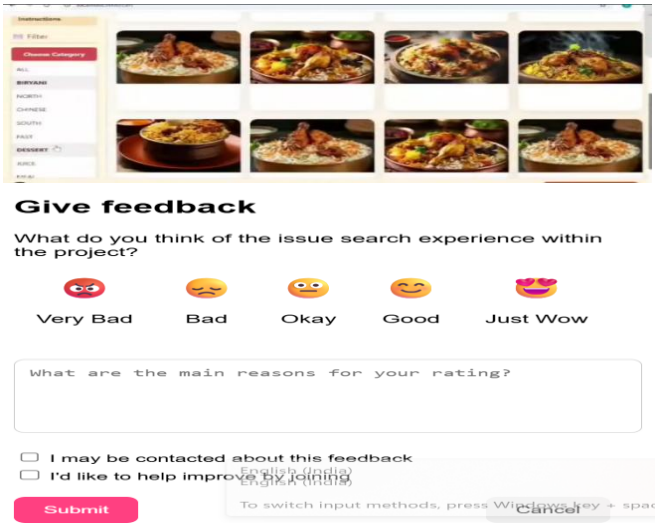
LOGIN AND REGISTER





ADD CART & FEEDBACK





FUTURE SCOPE

Although the current system is fully functional, several enhancements can be made to improve performance, usability, and features:

- **Online Payment Integration**

Integrate secure payment gateways such as Stripe or PayPal to allow customers to pay online.

- **Real-time Notifications**

Add push notifications or email alerts to notify users of order status updates instantly.

- **Advanced Search and Filtering**

Implement advanced search options like filtering by price, cuisine, or popularity.

- **Customer Reviews and Ratings**

Allow users to rate and review menu items, helping improve quality and customer satisfaction.

- **Delivery Tracking System**

Integrate GPS-based tracking to show customers the live location of delivery personnel.

- **Mobile Application Version**

Develop a dedicated Android/iOS mobile app for better accessibility and convenience.

- **Analytics and Reporting for Admin**

Add advanced reporting tools for admins to analyze sales, popular items, and customer behavior.

- **Multi-language Support**

Include multiple language options to cater to a wider audience.

CONCLUSION

The Food Ordering System has been successfully developed using Next.js, JavaScript, CSS, and MongoDB. It provides a convenient solution for customers to browse menus, place orders, and track them in real-time, while admins can efficiently manage menu items, orders, and generate reports.

The project involved phases like requirement analysis, system design, implementation, and testing. The frontend ensures a responsive and user-friendly interface, while the backend and database securely handle all operations. Extensive testing confirmed that all modules work correctly and the system is stable and reliable.

Overall, this project meets its objectives by providing a functional, scalable, and user-friendly platform for online food ordering. It improves customer satisfaction, simplifies order management for admins, and lays a foundation for future enhancements.

REFERENCE

- Swiggy Official Website. Online Food Ordering Platform. Available at: https://www.swiggy.com/
- Zomato Official Website. Food Delivery and Ordering Platform. Available at: https://www.zomato.com/
- W3Schools. Next.js Tutorial. Available at: https://www.w3schools.com/nextjs/
- GeeksforGeeks. Online Food Ordering System Project using JavaScript. Available at: https://www.geeksforgeeks.org/
- TutorialsPoint. HTML, CSS, JavaScript Guide. Available at: https://www.tutorialspoint.com/

TEAM MEMBERS

Dibya Ghosh:	Main project coordination; Frontend (Home, Login, Registration, Cart) Backend
Arpita Nath:	Frontend – Menu page (Biryani, North-Indian)
Sampurna Khan:	Frontend – Contact page
Krishna Dhali:	Frontend – About page
Rupsa Sen:	Feedback page