PROJECT SYNOPSIS REPORT

ON

Food Express: A Web Application

SUBMITTED

TO

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

FOR

Full Stack Engineering(22CS037)

Submitted To:
Mr. Rahul Singh

Submitted By:
Samar (2210992233)
Nandini Kaushik (2210991958)
Ishanesh Srivastava (2210991685)
Sambhav Gupta (2210992236)

Index

1. Problem Statement

In today's fast-paced world, ordering food efficiently is a crucial requirement for both customers and restaurant owners. Traditional food ordering systems suffer from inefficiencies such as long waiting times, miscommunication, and lack of real-time tracking. **Food Express** aims to solve these problems by providing a seamless and user-friendly food ordering platform that integrates restaurants, delivery services, and customers onto a single system.

2. Title of Project

"Food Express: A Smart Food Ordering and Delivery Platform"

3. Objective & Key Learning's

To design and develop a real-time food ordering and delivery system that allows users to browse restaurant menus, place orders, and track deliveries with minimal delay. The system should offer:

- **Real-time order tracking**
- **Secure and multiple payment options**
- **AI-based personalized recommendations**
- **Efficient order management for restaurants**

Key Learning's:

- **Web Technologies: Using React.js and Node.js to create a scalable full-stack solution.**
- **Real-Time Communication: Implementing live order tracking using WebSockets.**
- **Database Management: Storing restaurant details, orders, and user preferences using MongoDB.**
- **Security Practices: Implementing JWT authentication and secure payment gateways.**
- **Scalability: Designing a system that can handle multiple users and restaurants efficiently.**

4. Options Available to Execute the Project

a. Frontend Technologies:

- React.js: A widely used JavaScript library for building user interfaces with reactive components.
- HTML/CSS: For basic structure and styling.
- Flutter (for mobile): To create a cross-platform mobile chat app.

b. Backend Technologies:

- **Node.js:** Widely used for real-time applications with its non-blocking I/O and event-driven architecture.
- **MongoDB** – For database management.

c. Communication Protocols:

- **WebSockets** – For real-time updates (order status, delivery tracking).
- **HTTP/2 & REST APIs** – For efficient communication between client and server.

d. Databases:

- **MongoDB** – NoSQL database for dynamic data storage.
- **Firebase Realtime Database** – For instant order updates.

e. Additional Tools & Libraries:

- **Socket.io** – For WebSocket-based real-time order tracking.
- **Firebase Cloud Messaging** – For sending notifications.
- **Stripe / Razorpay** – For secure payments.
- **Google Maps API** – For restaurant discovery and tracking.

5. Advantages/Disadvantages

Advantages :

- **Real-Time Order Tracking: Users can track their orders from preparation to delivery.**
- **Secure Payments: Multiple payment gateways ensure seamless transactions.**
- **AI-Powered Recommendations: Suggests food items based on user behavior.**
- **Restaurant Management Dashboard: Helps restaurants manage orders efficiently.**
- **Scalable Architecture: Can handle multiple restaurants and customers concurrently.**

Disadvantages:

- **High Server Load: Real-time tracking requires optimized server resources.**
- **Security Challenges: Online transactions require strong encryption and fraud detection.**
- **Complex Deployment: Managing cloud hosting and databases requires expertise.**

6. References

- **Node.js Official Site:** https://nodejs.org/en
- **React Documentation:** https://react.dev
- **Express.js Documentation:** https://expressjs.com
- **MongoDB Documentation:** https://www.mongodb.com/docs
- **Stripe Payment Gateway:** https://stripe.com/docs
- **Google Maps API:** https://developers.google.com/maps