**Full Stack Development with MERN**

**Project Documentation format**

**1. Introduction**

* **Project Title:** SB Food Ordering Website
* **Team Members:**

S.Tejashree

G. Swathi

S. Karthikeyan

M.Surya Kumar

S.Thiru Murugan

**2. Project Overview**

* **Purpose:**

SB Food Ordering app is a online platform app developed using MERN Stack. It is a easy and simple food ordering app. It allow us to search the restaurant, order it and make payment in the website using the basic features.

**Features:**

* + Browse restaurant
  + Add to cart functionality
  + Secure payments
  + Order Summary Page
  + User account system with password reset, recovery.

**3. Architecture**

* **Frontend:** The React-based frontend ensures a responsive and utilizing React Router for navigation.
* **Backend:** The backend, powered by Node.js and Express.js, provides user authentication, restaurant management, and order processing , stripe for Payment.
* **Database:** MongoDB serves as the database, storing information about users, products, orders, and transactions. The schema is optimized for scalability and quick retrieval of data.

**4. Setup Instructions**

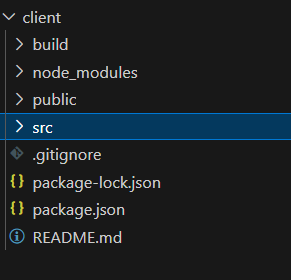
* **Prerequisites:**
* Node.js (v14 or later)
* MongoDB (locally or hosted)
* Git

**Installation:**

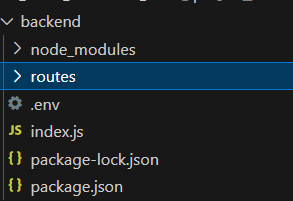
* Create a folder. With two subfolders for Client and Backend.
* In Backend, use command: **npm i mongodb express bcrypt jsonwebtoken**
* Also install other dependencies
* In Frontend Folder, use command**: npm i react**
* Add .env files in the client and server directories with necessary credentials (MongoDB URI, Stripe API key, etc.)

**5. Folder Structure**

* **Client:**



* **Server:**

****

* **6. Running the Application**
* Provide commands to start the frontend and backend servers locally.
  + **Frontend:** npm start in the client directory.
  + **Backend:** npm start in the server directory.

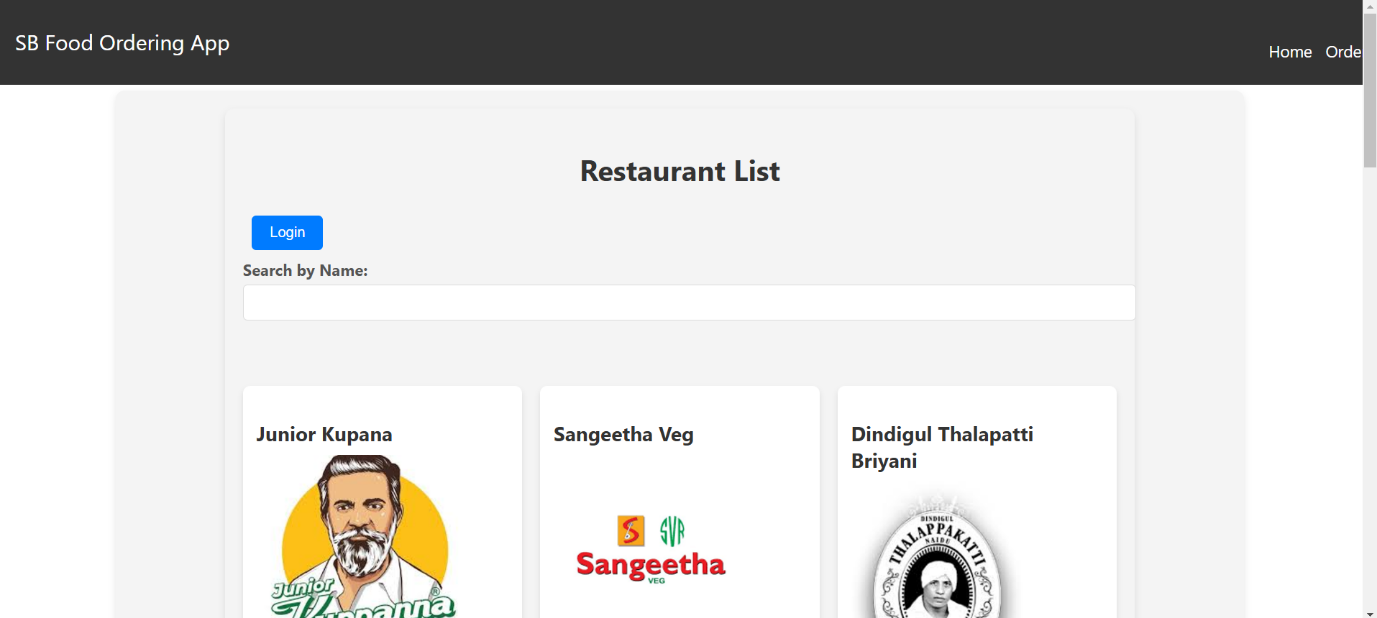
**7. Authentication**

* The application uses JSON Web Tokens (JWT) and Bcrypt for secure user authentication.
* Password recovery and reset processes are implemented

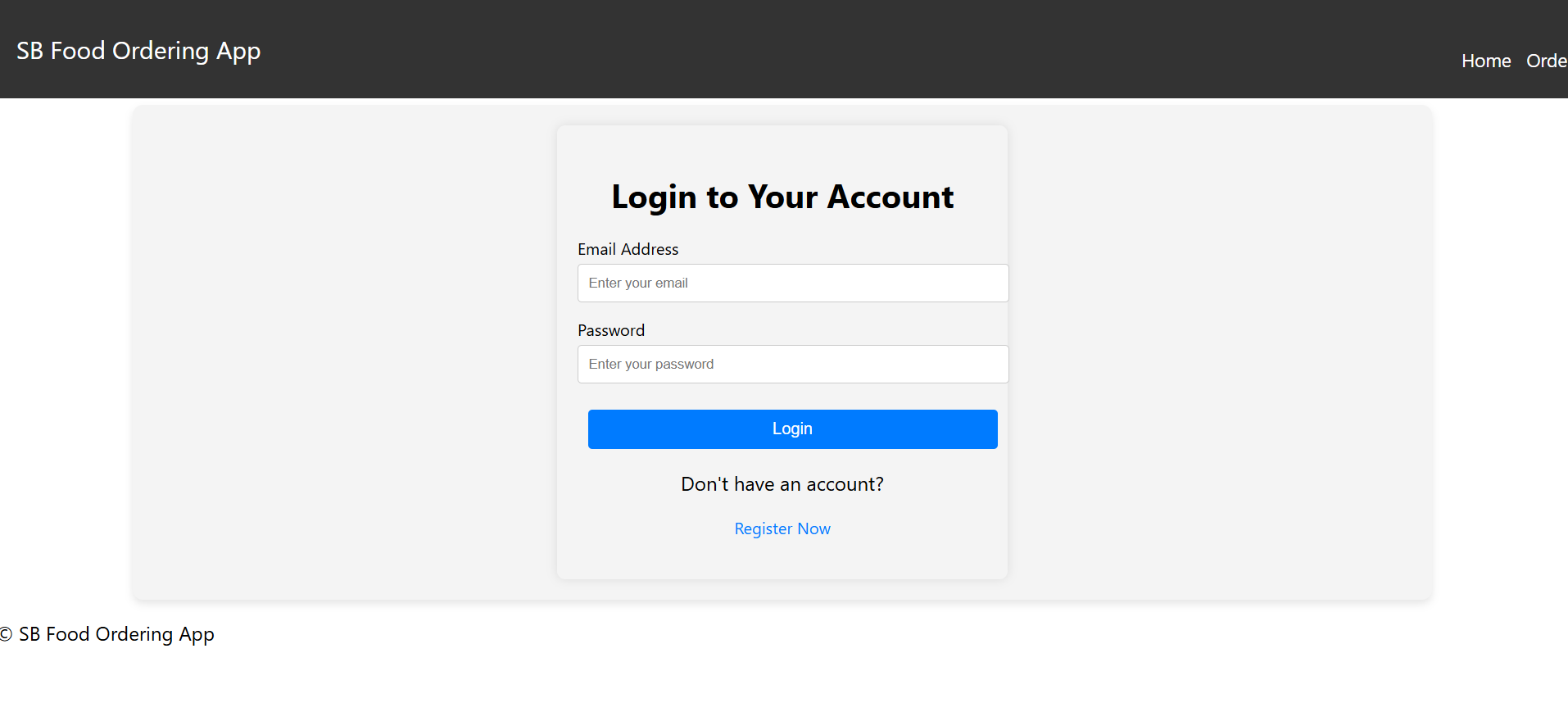
**8. User Interface**

* The user interface is designed with modern styling libraries such as CSS.
* Key features include a responsive navbar, restaurant listing , and a Order summary page.
* Various features include searching, sorting, payment gateway and many more.

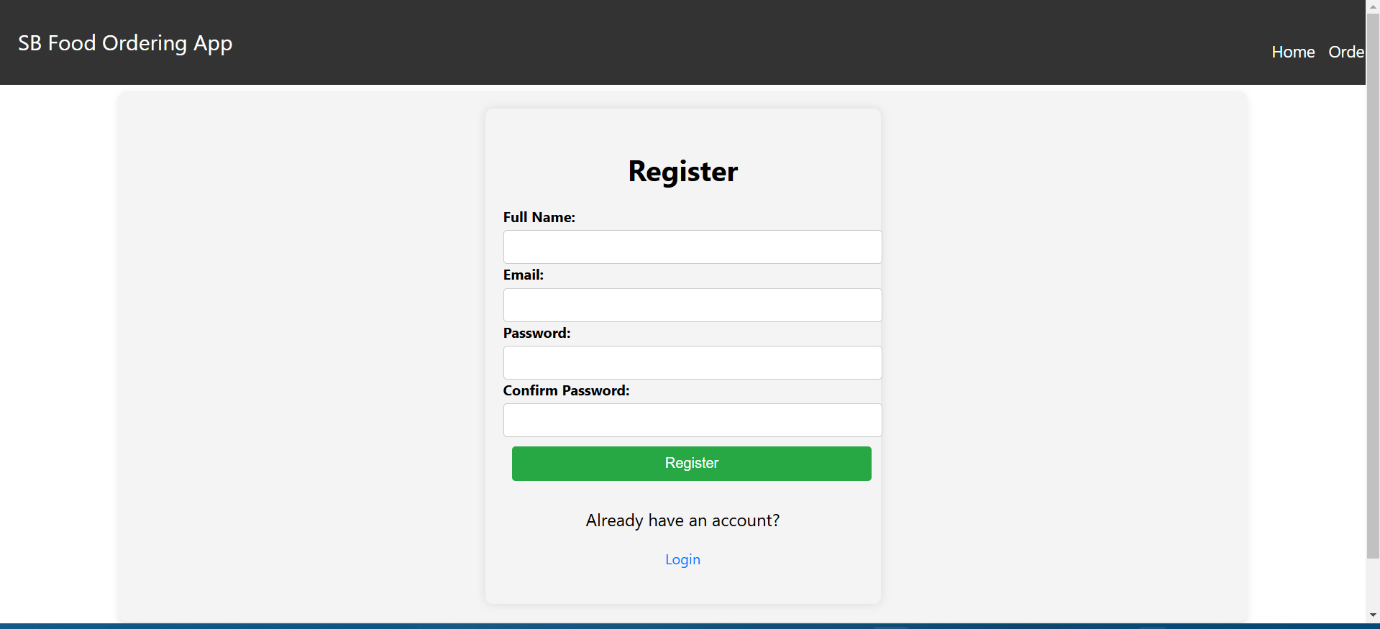
**HOME PAGE**

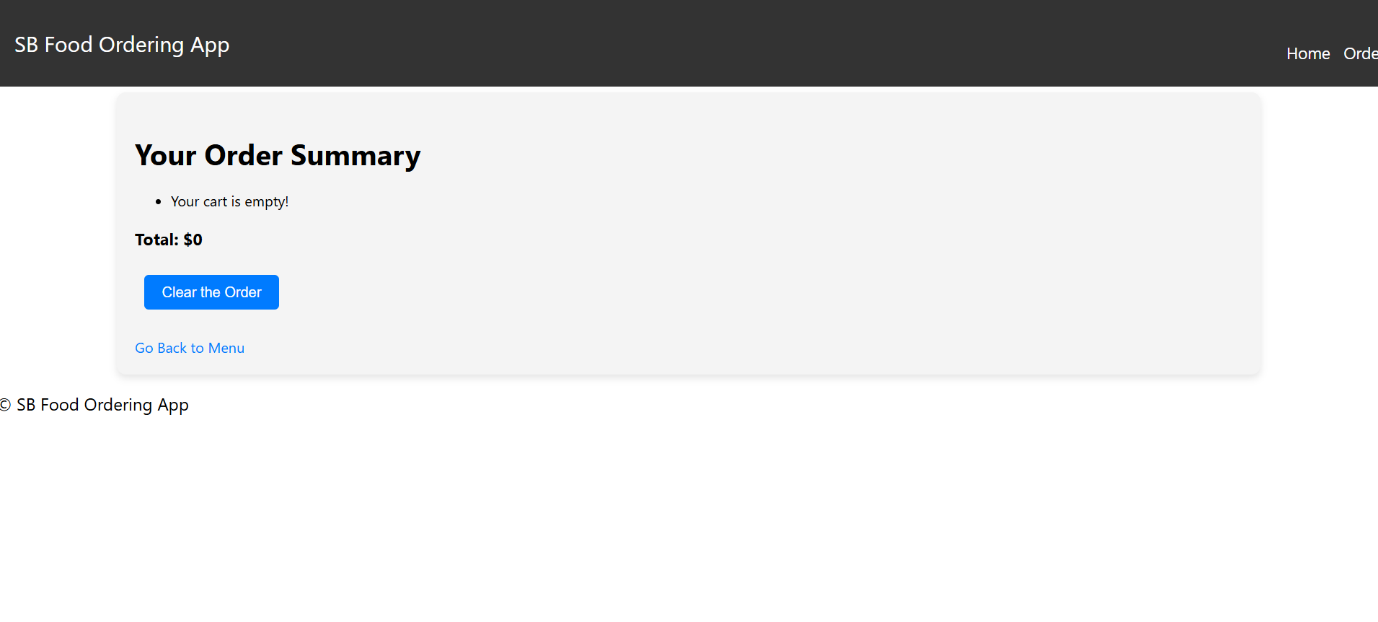


**LOGIN PAGE**

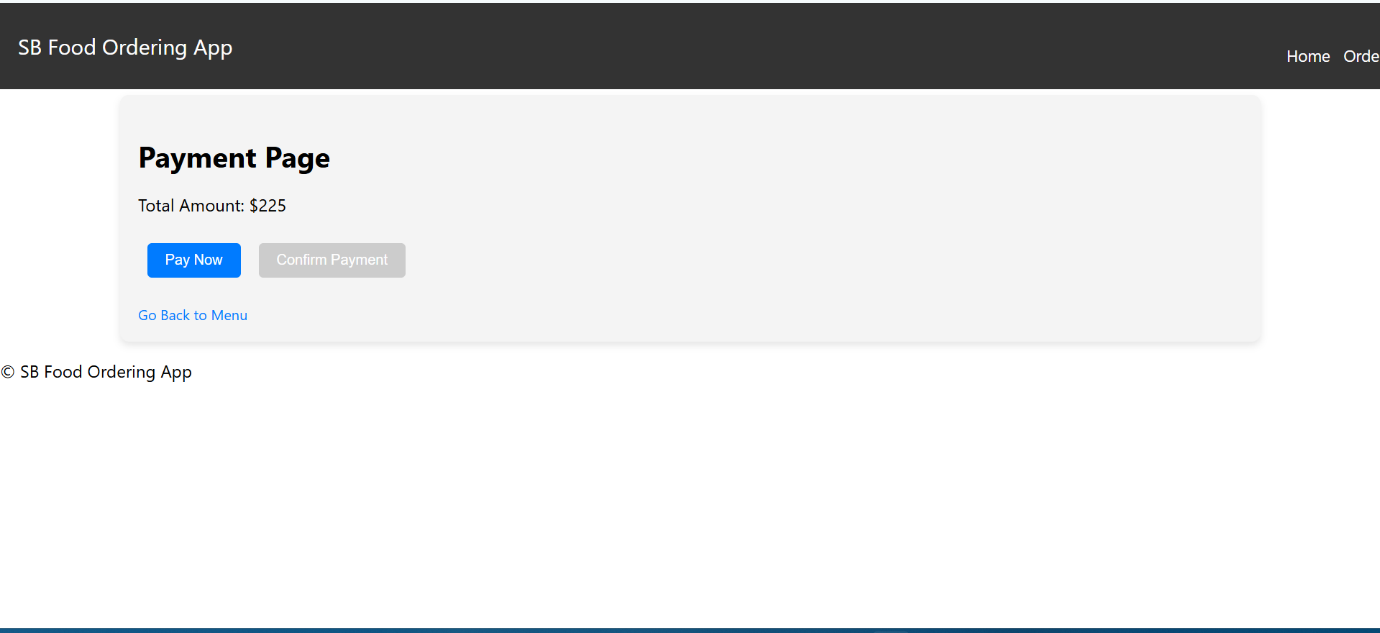
****

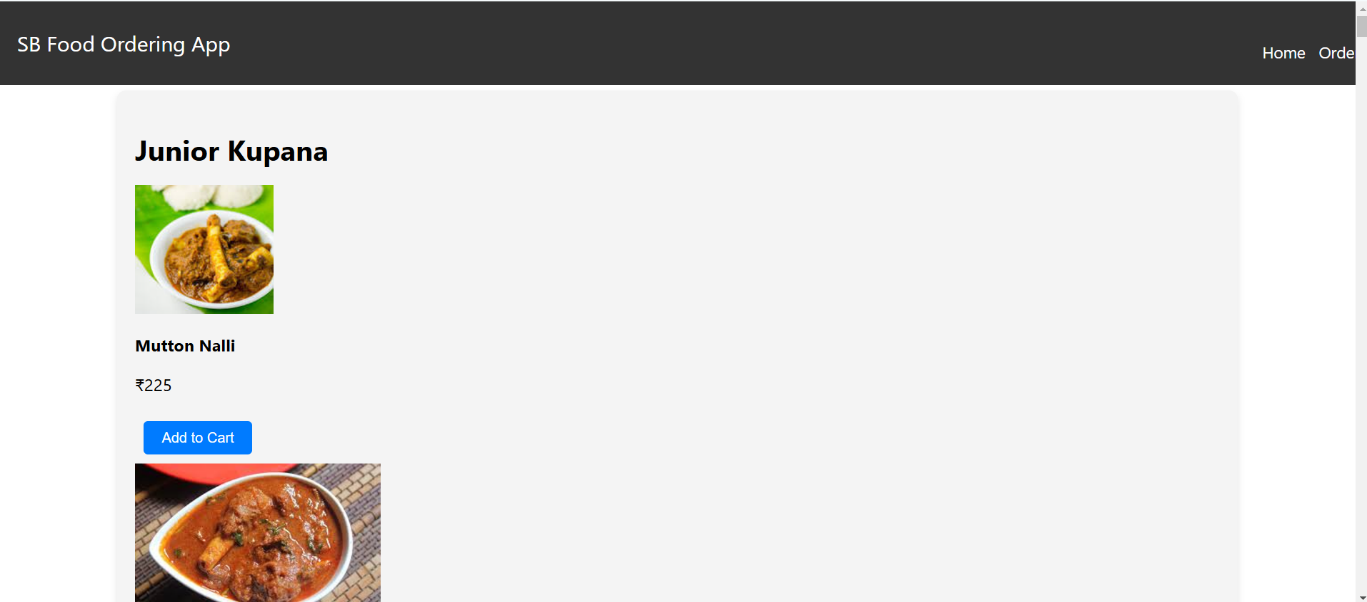
**REGISTER PAGE**

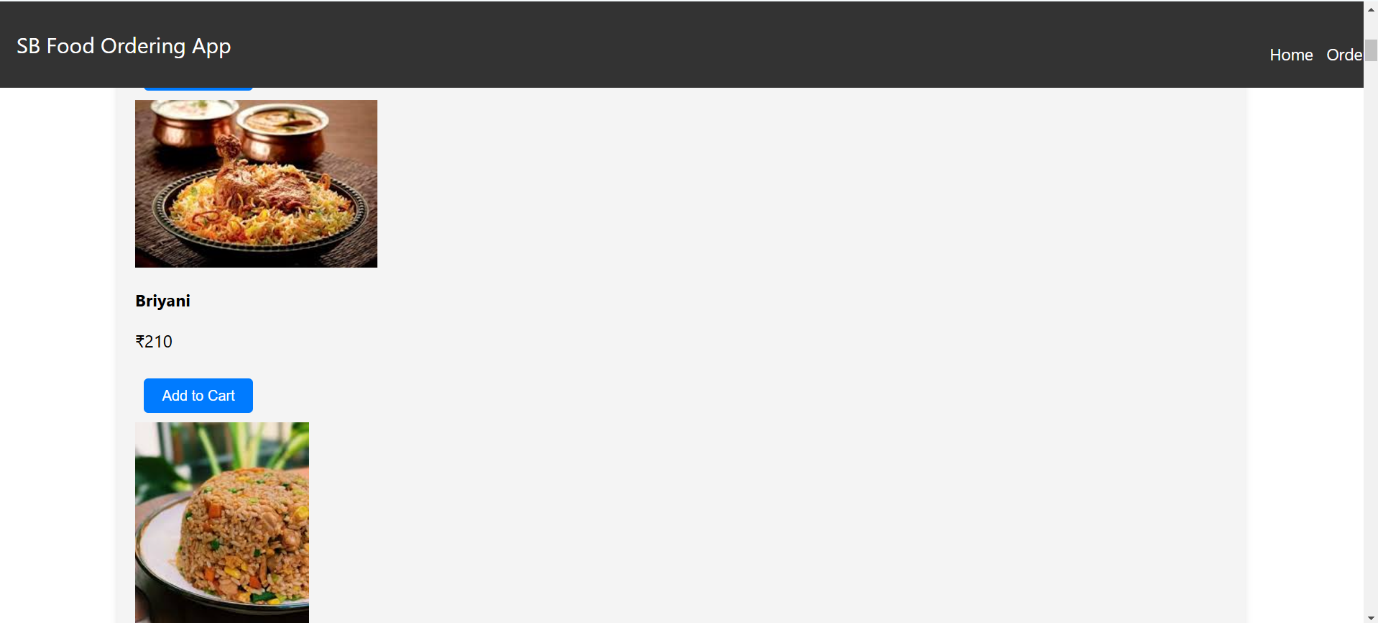
****

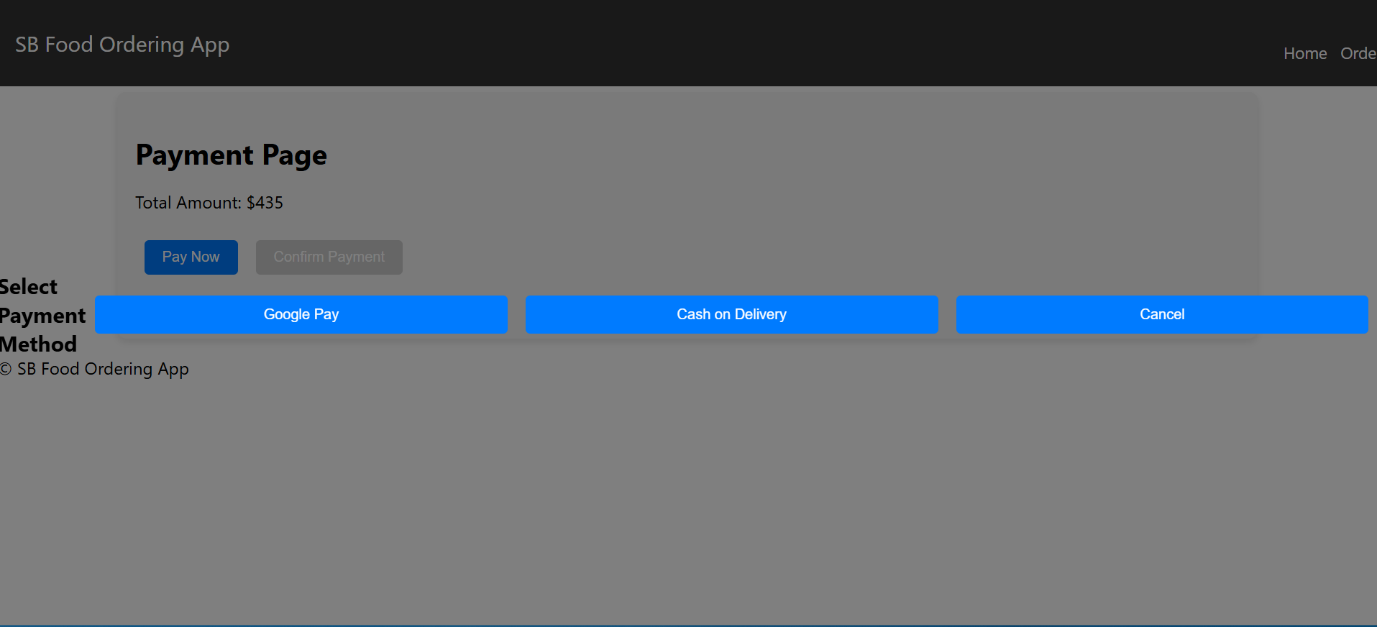
**ORDER SUMMARY PAGE**

**PAYMENT PAGE**

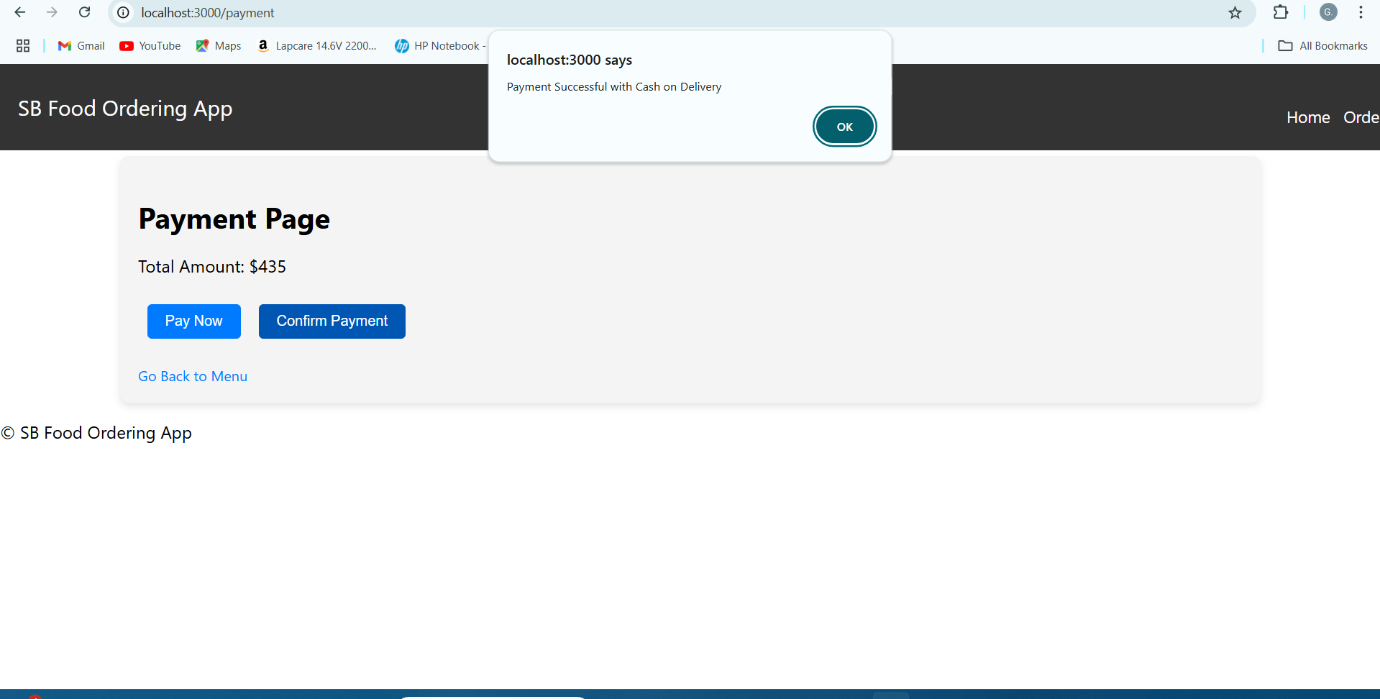
****

**MENU PAGE**

****

**PAYMENT PROCESS PAGE**

**PAYMENT SUCESSFUL PAGE**

****

**9. Testing**

* Tools: Postman
* Unit tests are written for key components
* End-to-end testing ensures the overall functionality of the system.

**10. Known Issues**

* Not Responsive

**11. Future Enhancements**

* Implement AI-driven product recommendations.
* Add a chatbot for customer support.
* Integrate real-time order tracking.
* Create more UI/UX Design to make it more impressive