FOOD ORDERING APP PROJECT REPORT

1. INTRODUCTION

- **Purpose**: This project is a full-stack food ordering application developed using the **MERN stack** (MongoDB, Express.js, React, and Node.js). It allows users to browse available food items, add items to their cart, place orders, and make payments.
- **Scope**: The application is designed for restaurant customers, providing a seamless user experience for ordering food online. Additionally, an admin interface enables restaurant staff to manage menu items, track orders, and update availability.

2. PROJECT OBJECTIVES

- Provide a convenient platform for users to order food online.
- Develop a responsive user interface with a smooth ordering experience.
- Enable restaurant admins to add, update, or delete menu items and manage orders.
- Implement a secure and scalable database to store user, order, and menu information.

3. TECHNOLOGIES USED

- **Frontend**:
    - **React**: Used for building dynamic and responsive user interfaces. It allows modular component-based design, making the application scalable and easy to maintain.
    - **CSS**: Styling frameworks for creating a responsive layout, ensuring a seamless experience across different devices.
- **Backend**:
    - **Node.js**: Provides the runtime environment for executing JavaScript server-side, allowing a consistent language across the stack.
    - **Express.js**: A web application framework for building RESTful APIs. It handles HTTP requests and connects the frontend with the database.
- **Database**:

    - **MongoDB Atlas**: A cloud-based, NoSQL database that stores data as collections and documents, making it scalable for large volumes of data. This database holds information about users, food items, and orders, supporting the app's real-time data needs.

- **Authentication and Authorization**:
    - **JWT (JSON Web Token)**: Used to manage user sessions securely. Tokens are generated during login and validated on each request, ensuring that only authenticated users can access certain features.

- **Payment Processing**:
    - **Stripe Integration**: Stripe is integrated to securely process payments within the application. It provides a reliable and secure gateway, supporting multiple payment methods, and offering an optimized user experience with minimal setup complexity.

4.SYSTEM INTERACTION FLOW

1. **User Browsing and Cart Management**:
    - The user accesses the app and can browse food items by category.
    - When a user adds items to the cart, the React frontend maintains this data in state and displays a summary for review.

2. **Order Placement and Payment**:
    - Upon checkout, the frontend sends an order request to the backend.
    - The backend verifies the cart items and calculates the total price.
    - The payment request is sent to Stripe via the backend, securely handling sensitive information.

3. **Order and Payment Updates**:
    - MongoDB is updated to reflect order details and payment status, and users receive feedback on their order status.

5.API ROUTE CONFIGURATION

```
app.use("/api/food",foodRouter)
app.use("/images",express.static('uploads'))
app.use('/api/user', userRouter)
app.use('/api/cart', cartRouter)
app.use('/api/order', orderRouter)
```

API Endpoint Descriptions

- **Food API (/api/food)**:
  - Used to manage food items available in the app.
  - Endpoints might include:
    - GET /api/food: Retrieve all food items.
    - GET /api/food/:id: Retrieve details of a specific food item.
    - POST /api/food: Add a new food item (admin functionality).
    - PUT /api/food/:id: Update a food item (admin functionality).
    - DELETE /api/food/:id: Delete a food item (admin functionality).
- **Image API (/images)**:
  - A static route serving images directly from the uploads folder.
  - Images can be accessed by specifying the filename in the path (e.g., /images/food_1.png).
- **User API (/api/user)**:
  - Manages user-related operations, including authentication.
  - Endpoints might include:
    - POST /api/user/register: Register a new user.
    - POST /api/user/login: Authenticate a user and return a JWT.
    - GET /api/user/profile: Retrieve user profile information (requires authentication).
- **Cart API (/api/cart)**:
  - Handles cart operations for adding or modifying items before checkout.
  - Endpoints might include:
    - POST /api/cart/add: Add an item to the user’s cart.
    - DELETE /api/cart/remove/:id: Remove an item from the user’s cart.
    - GET /api/cart: View current items in the cart.
- **Order API (/api/order)**:

- Manages the order process, from creation to viewing past orders.
- Endpoints might include:
  - POST /api/order/create: Create a new order for the items in the cart.
  - GET /api/order/:id: Retrieve details of a specific order.
  - GET /api/order/history: View a list of past orders (requires authentication).

6.FEATURES

User Features

- **Browse Food Items**:
  - Users can explore a wide range of food items displayed by category or type. Each food item card provides essential details such as name, description, price, and an image, along with the restaurant name it belongs to. This gives users insight into the source of each dish and helps in decision-making.
  - A search bar and filter options allow users to quickly find specific food items or sort them by preferences, like popular dishes or dietary options.
- **Add to Cart**:
  - Users can select food items and add them to their cart, with the option to view and adjust cart contents at any time. Each cart item displays the food details along with the restaurant name, ensuring transparency about where each item is from.
  - The cart also allows users to update quantities or remove items before proceeding to checkout, providing flexibility in finalizing orders.
- **Place Orders**:
  - Once users are ready to order, they can proceed to checkout, where they can review their order summary, which includes item details, prices, restaurant names, and the total amount. The checkout process includes order confirmation and payment, supported by a secure payment gateway (e.g., Stripe) for a seamless and safe transaction.
  - After placing an order, users can view order history, which displays past orders, including order status, items, and restaurant names, for easy reference.
- **Authentication**:
  - Users can securely sign up, log in, and log out using email and password credentials. Authentication is managed with session tokens for security. Only

authenticated users can place orders and access certain features like order history and profile updates.

Admin Features

The Admin Panel provides robust tools for administrators to manage food items, view the catalog, and handle orders seamlessly.

Add Items

- **Upload Image**:
    - Admins can upload an image for each food item directly through the interface. This image is essential for users, helping them visually identify each item on the menu and enhancing the overall appeal.
    - The uploaded image is saved securely in the backend, making it readily accessible and ensuring that each item has a consistent, high-quality appearance in the catalog.
- **Project/Food Item Name**:
    - Admins can specify a unique name for each food item, which will be displayed to users. This name serves as the primary identifier for users as they browse the catalog.
- **Description**:
    - A brief but informative description can be added for each food item. This description provides details about the item, such as ingredients, flavors, and special attributes, giving users an idea of what they're ordering.
- **Product Category**:
    - Food items can be assigned to specific categories (e.g., salads, main courses, desserts) to help users filter items easily. This categorization improves the browsing experience by allowing users to find what they're looking for more efficiently.
- **Product Price**:
    - Each food item has a price field where admins can input the cost of the item. This price is displayed alongside the item details, giving users transparency about the cost before they add it to their cart.

List Items

- **View All Food Items**:

    - The admin panel includes a listing page that displays all available food items. Each item is shown with its essential details, including the name, description, price, category, and image.
    - Admins can use this view to **quickly review** the catalog, ensuring that all items are accurately represented. This view allows admins to manage the item status, such as updating descriptions, adjusting prices, or changing availability.

Orders

- **Manage Orders**:
    - The orders section lists all active and past user orders, displaying each order’s ID, customer details, items ordered, total cost, and order date.
    - **Order Status**: Admins can update each order’s status (e.g., processing, prepared, out for delivery, delivered). This status is reflected on the user’s account, keeping them informed about their order’s progress.
    - **View Order Details**: Admins can access individual order details, including itemized lists with product names, quantities, prices, and total amount. This comprehensive view helps admins confirm and organize orders accurately.
    - **Order Management**: This section also allows for handling order issues like cancellations, refunds, or order modifications to ensure a smooth experience for users.

7.IMPLEMENTATION DETAILS

- **Backend (Node.js & Express.js)**
    - Database Connection: MongoDB Atlas is connected using Mongoose in config/db.js.
    - Models: Defined in the models folder (e.g., Food.js and User.js) to structure the data for food items and users.
    - Controllers: Handle business logic for food items and users (e.g., foodController.js, userController.js).
    - Routes: Define endpoints for food and user-related functionalities.

- **Frontend (React.js)**
    - Components: Separate components are created for each part of the UI (e.g., FoodItem, Cart, Navbar).

  - Pages: Divided into pages such as HomePage and CartPage to represent different views.
  - State Management: React’s Context API or libraries like Redux are used for global state management.

- **Database (MongoDB)**

  - Collections: Collections are set up for users, food items, and orders.
  - Schemas: Mongoose schemas define the structure of documents in each collection.

8.CHALLENGES AND SOLUTIONS

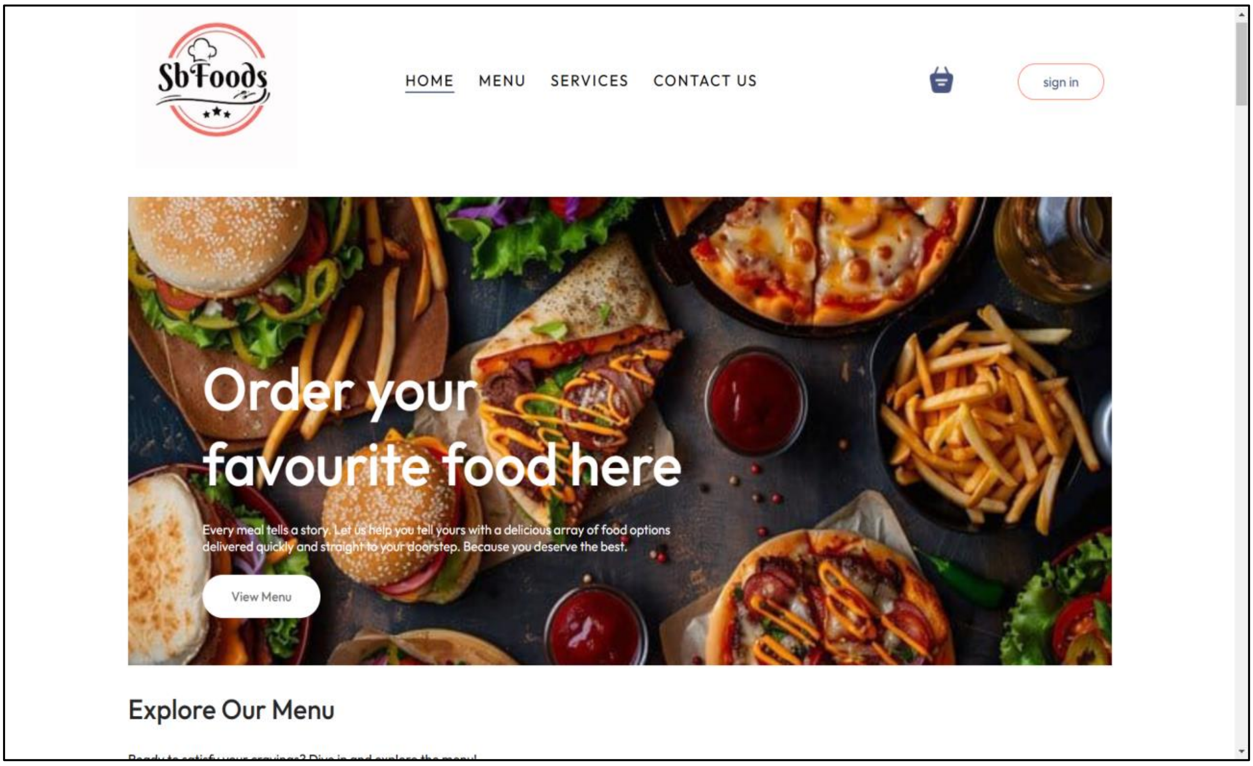
- **Data Synchronization**: To ensure real-time updates, the backend uses efficient API calls to keep frontend data synchronized with MongoDB.
- **Image Storage**: Images are stored on the server, with MongoDB storing file paths for easy access.
- **Scalability**: MongoDB Atlas provides a scalable solution, handling high volumes of data for growing user demand.

9.FUTURE ENHANCEMENTS

- **Payment Gateway Integration:** Integrate a payment gateway for real transactions.
- **Enhanced Admin Dashboard:** Create a dashboard with analytics for the admin.
- **User Feedback and Ratings:** Allow users to rate and review food items.

10.SCREENSHOTS OF FOOD ORDERING APP:

HOME PAGE



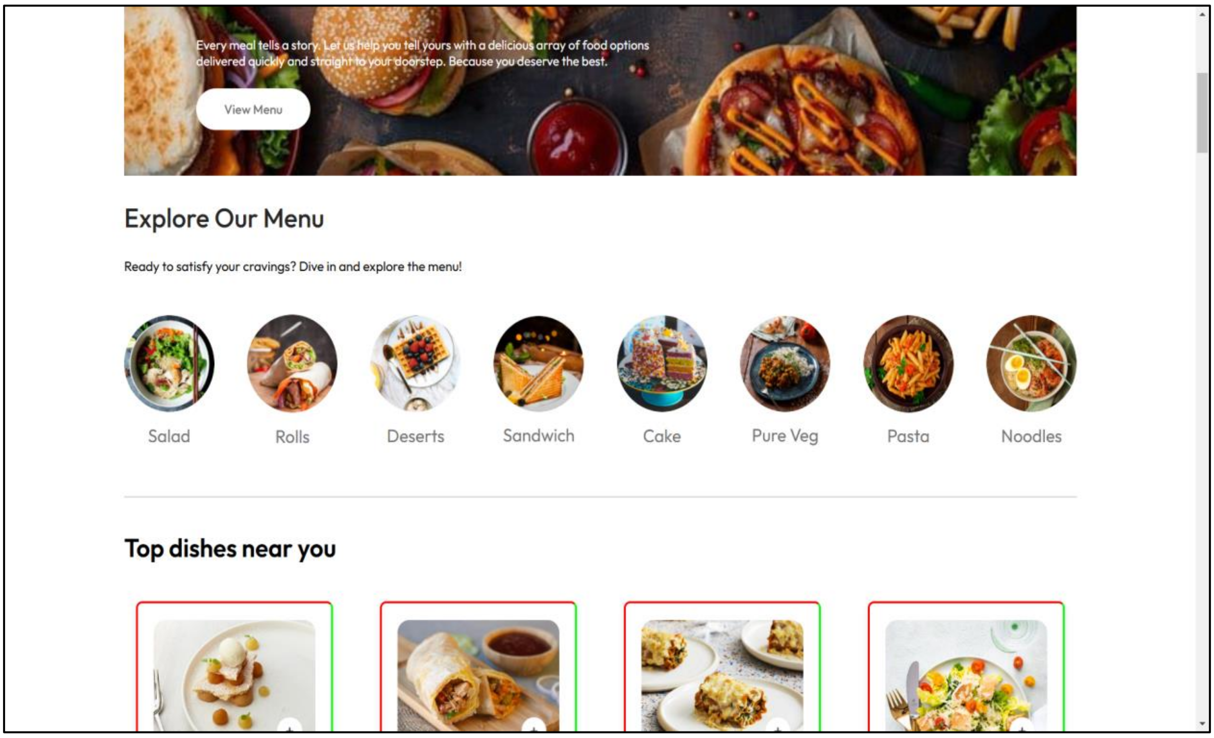
Order your favourite food here

Every meal tells a story. Let us help you tell yours with a delicious array of food options delivered quickly and straight to your doorstep. Because you deserve the best.

Explore Our Menu

Ready to satisfy your cravings? Dive in and explore the menu!

MENU PAGE



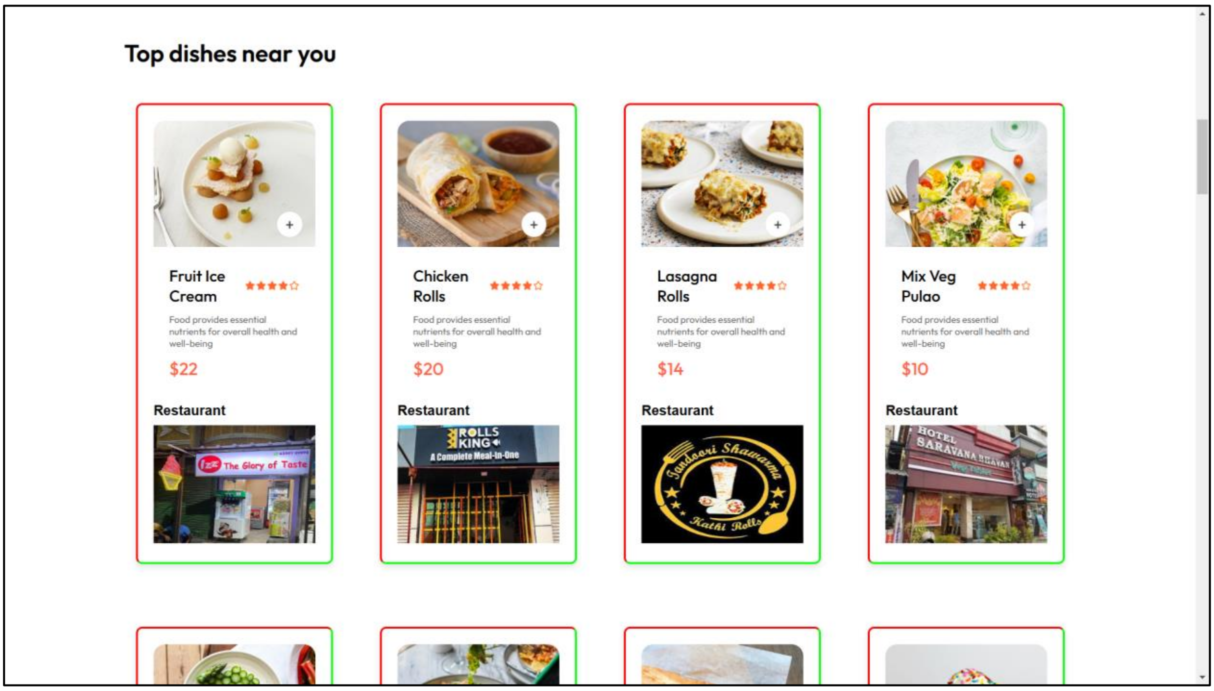
Every meal tells a story. Let us help you tell yours with a delicious array of food options delivered quickly and straight to your doorstep. Because you deserve the best.

Explore Our Menu

Ready to satisfy your cravings? Dive in and explore the menu!

Top dishes near you

FOOD ITEMS WITH RESTAURANT



Top dishes near you

Fruit Ice Cream ★★★★☆

Food provides essential nutrients for overall health and well-being

$22

Restaurant

Chicken Rolls ★★★★☆

Food provides essential nutrients for overall health and well-being

$20

Restaurant

Lasagna Rolls ★★★★☆

Food provides essential nutrients for overall health and well-being

$14

Restaurant

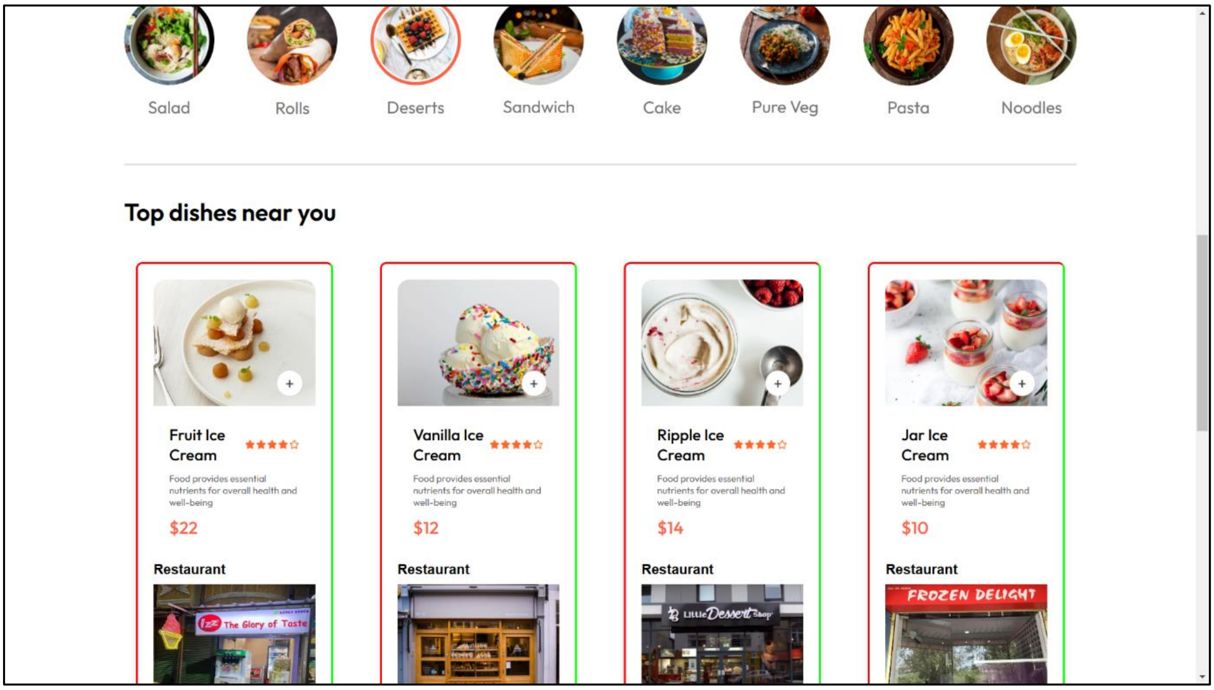
Mix Veg Pulao ★★★★☆

Food provides essential nutrients for overall health and well-being

$10

Restaurant

FOOD ITEMS BASED ON CATEGORY-“DESERTS”



Top dishes near you

Fruit Ice Cream ★★★★☆

Food provides essential nutrients for overall health and well-being

$22

Restaurant

Vanilla Ice Cream ★★★★☆

Food provides essential nutrients for overall health and well-being

$12

Restaurant

Ripple Ice Cream ★★★★☆

Food provides essential nutrients for overall health and well-being

$14

Restaurant

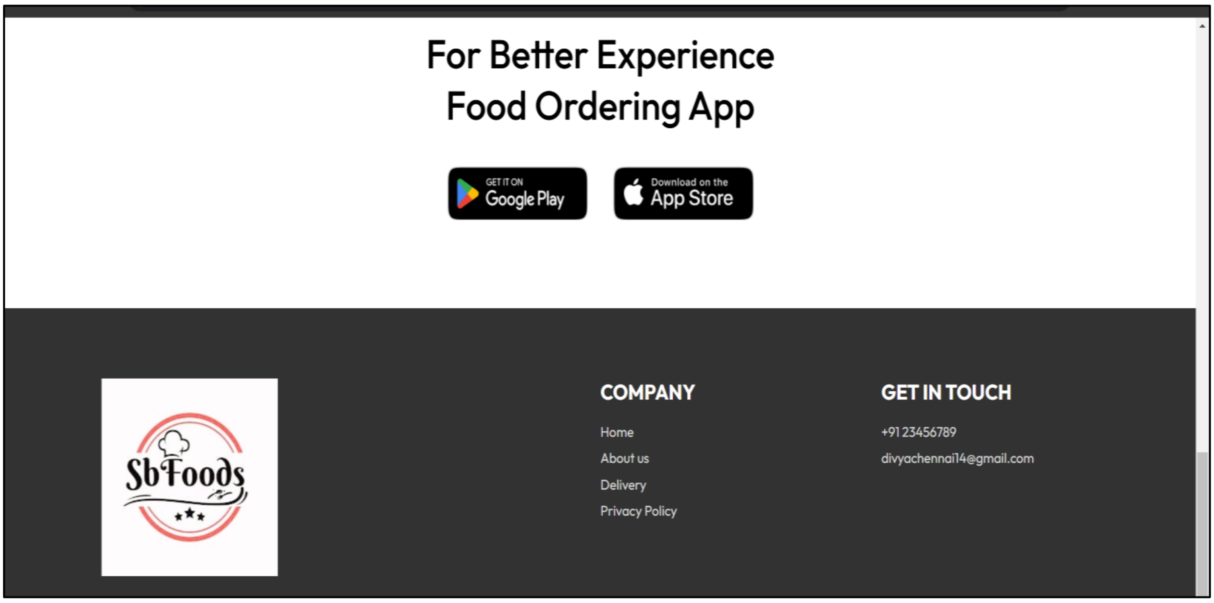
Jar Ice Cream ★★★★☆

Food provides essential nutrients for overall health and well-being

$10

Restaurant

SERVICES AND CONTACT US



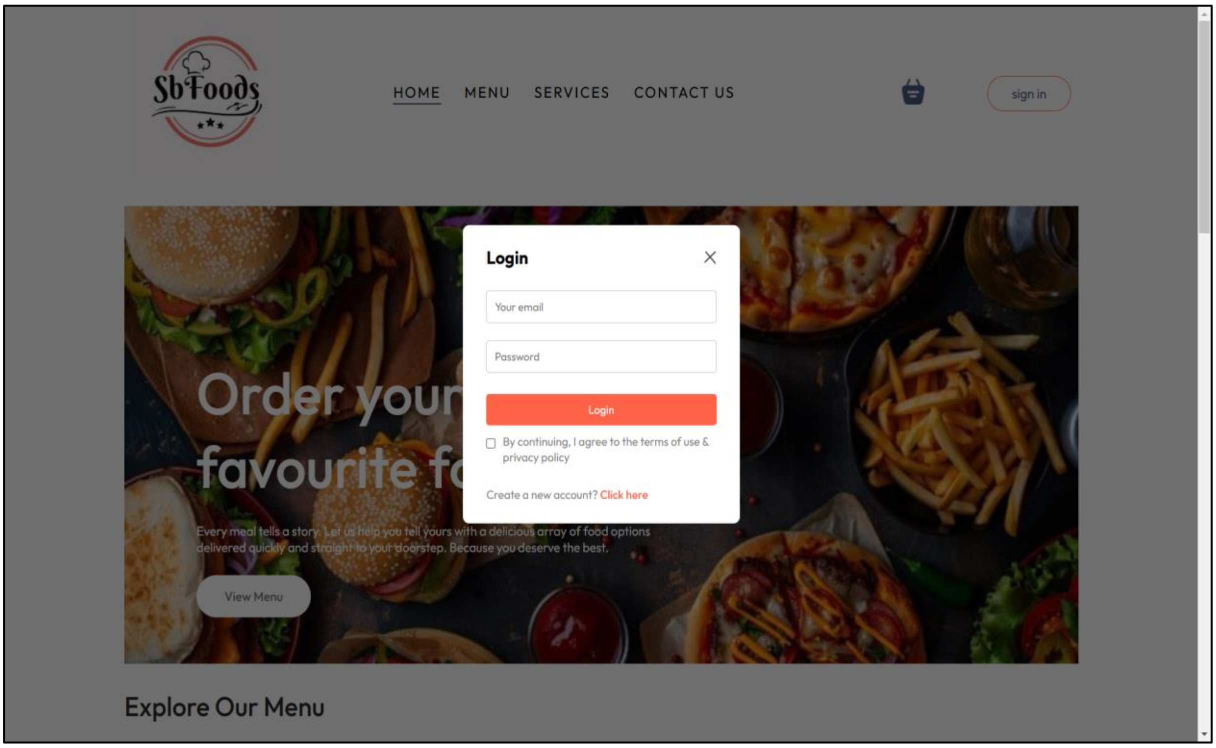
For Better Experience
Food Ordering App

COMPANY

GET IN TOUCH

+91 23456789
divyachennai14@gmail.com

LOGIN PAGE



Login

□ By continuing, I agree to the terms of use & privacy policy

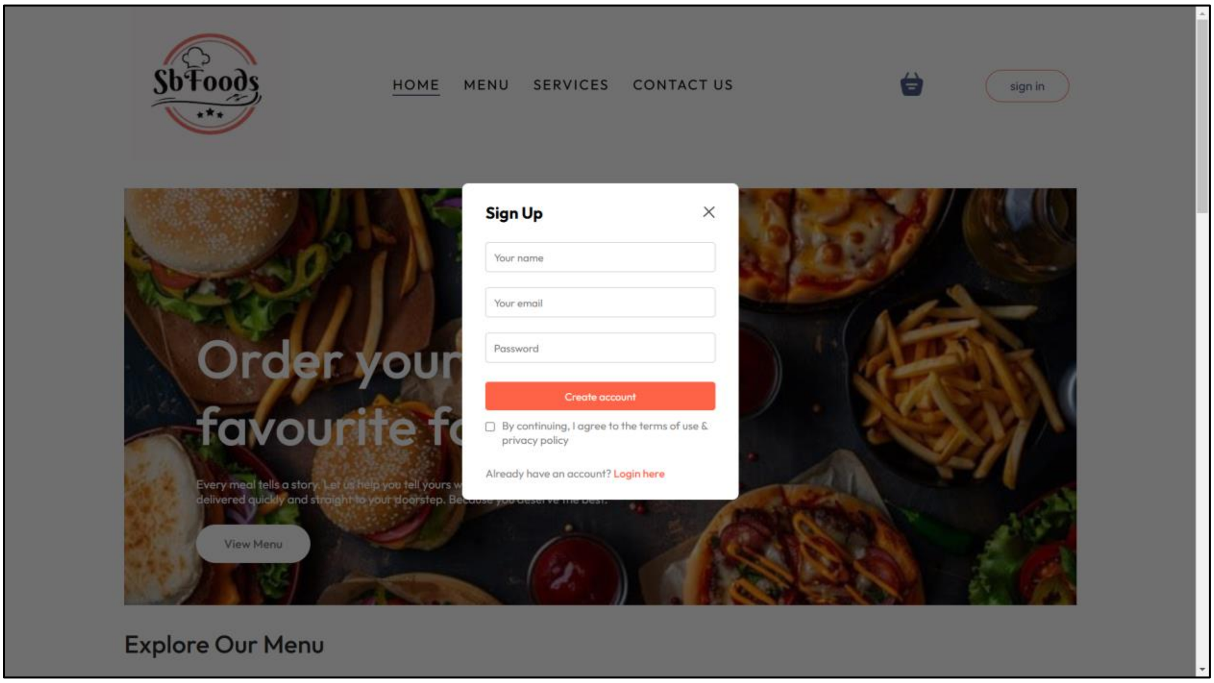
Create a new account? Click here

Order your favourite fo

Every meal tells a story. Let us help you tell yours with a delicious array of food options delivered quickly and straight to your doorstep. Because you deserve the best.

Explore Our Menu

SIGN UP PAGE:



Sign Up

Your name

Your email

Password

Create account

□ By continuing, I agree to the terms of use & privacy policy

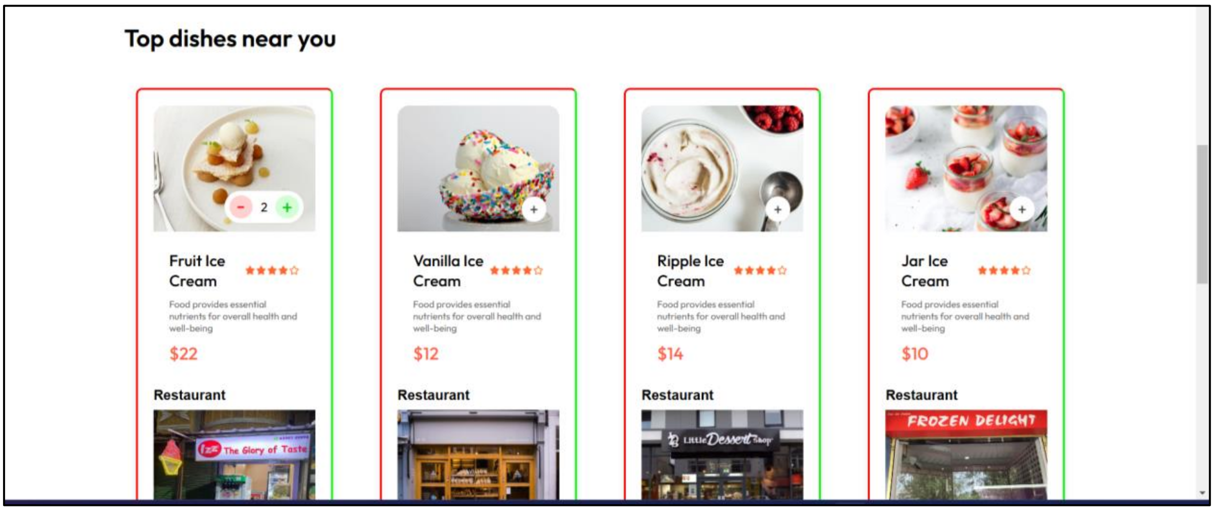
Already have an account? Login here

Order your favourite fo

Every meal tells a story. Let us help you tell yours w
delivered quickly and straight to your doorstep. Because you deserve the best.

Explore Our Menu

ADDING FOODITEM



Top dishes near you

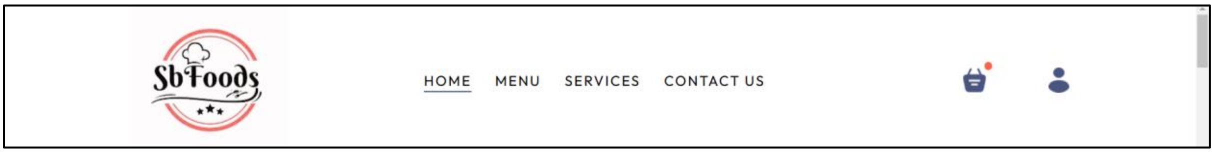
Fruit Ice Cream ★★★★☆ – 2 +
Food provides essential nutrients for overall health and well-being
$22
Restaurant

Vanilla Ice Cream ★★★★☆ +
Food provides essential nutrients for overall health and well-being
$12
Restaurant

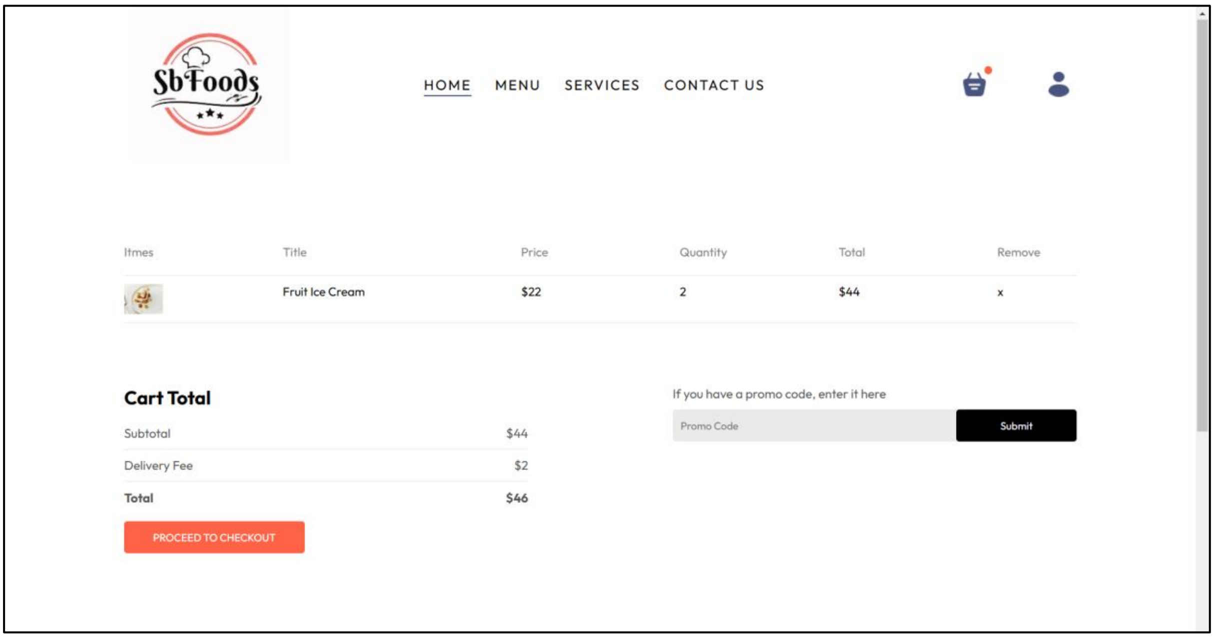
Ripple Ice Cream ★★★★☆ +
Food provides essential nutrients for overall health and well-being
$14
Restaurant

Jar Ice Cream ★★★★☆ +
Food provides essential nutrients for overall health and well-being
$10
Restaurant

CART NOTIFICATION



CART DETAILS:



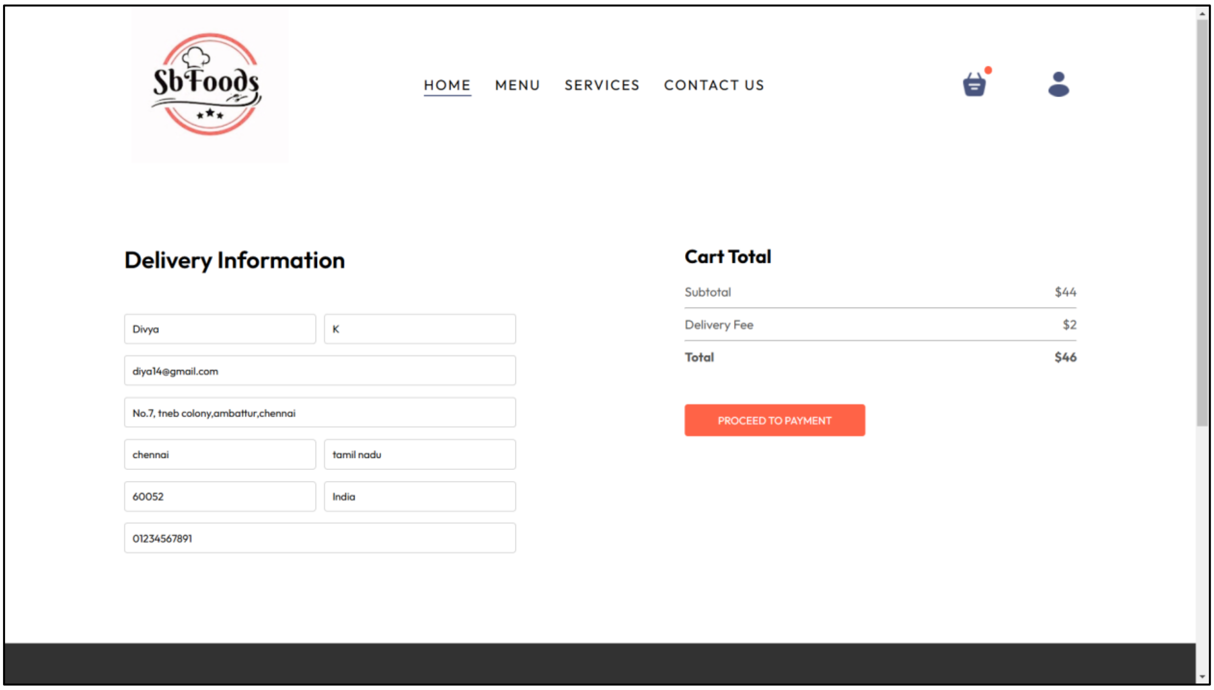
Itmes	Title	Price	Quantity	Total	Remove
	Fruit Ice Cream	$22	2	$44	x

Cart Total

Subtotal	$44
Delivery Fee	$2
Total	**$46**

If you have a promo code, enter it here

DELIVERY INFORMATION



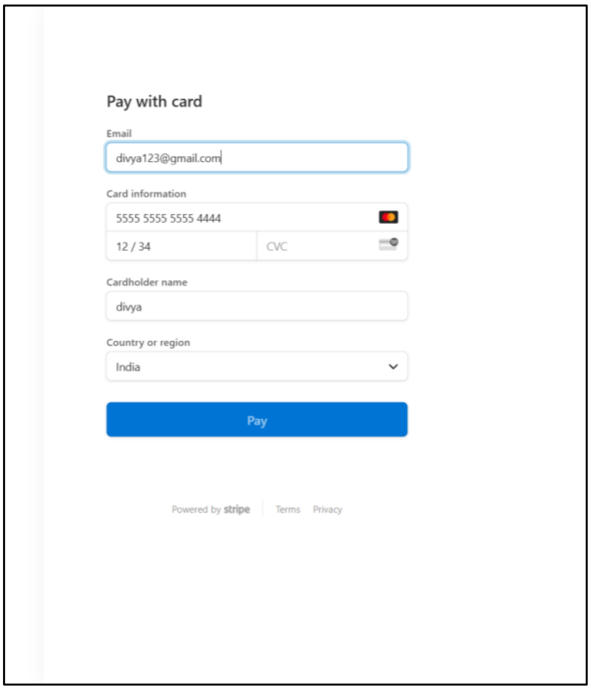
Delivery Information

Divya	K
diya14@gmail.com	
No.7, tneb colony,ambattur,chennai	
chennai	tamil nadu
60052	India
01234567891	

Cart Total

Subtotal	$44
Delivery Fee	$2
Total	**$46**

CARD DETAILS FOR PAYMENT



Pay with card

Email

divya123@gmail.com

Card information

5555 5555 5555 4444

12 / 34

CVC

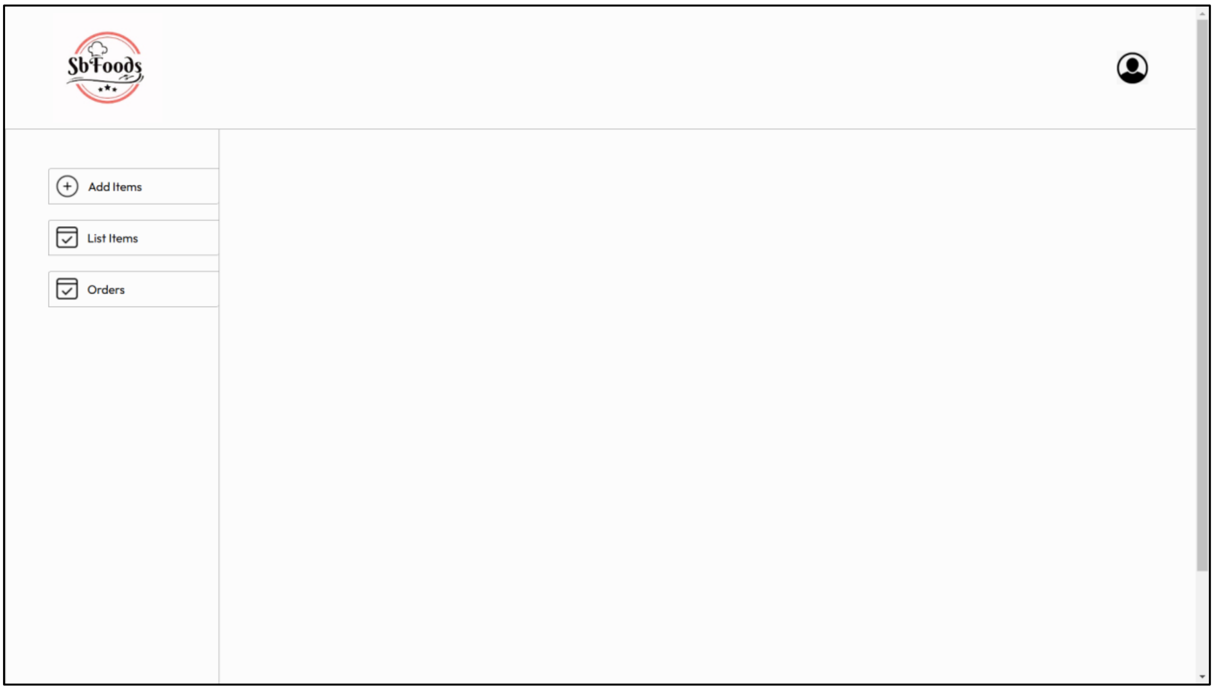
Cardholder name

divya

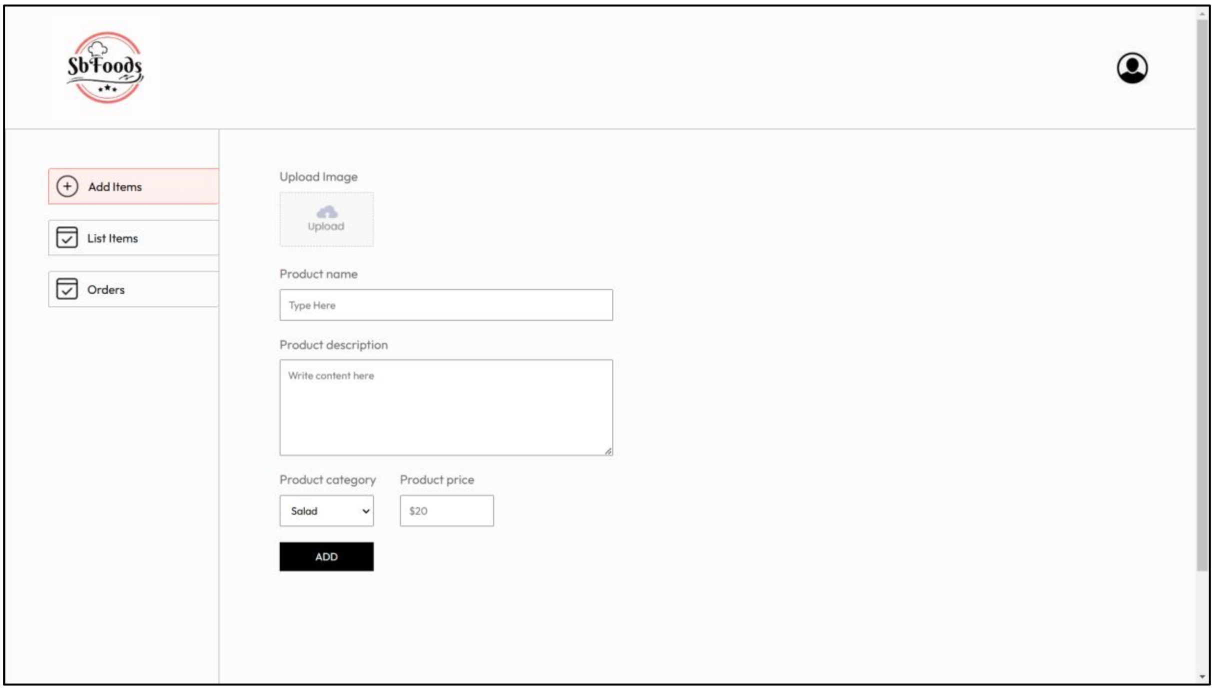
Country or region

India

ADMIN PAGE



ADD ITEMS



Upload Image

Product name

Type Here

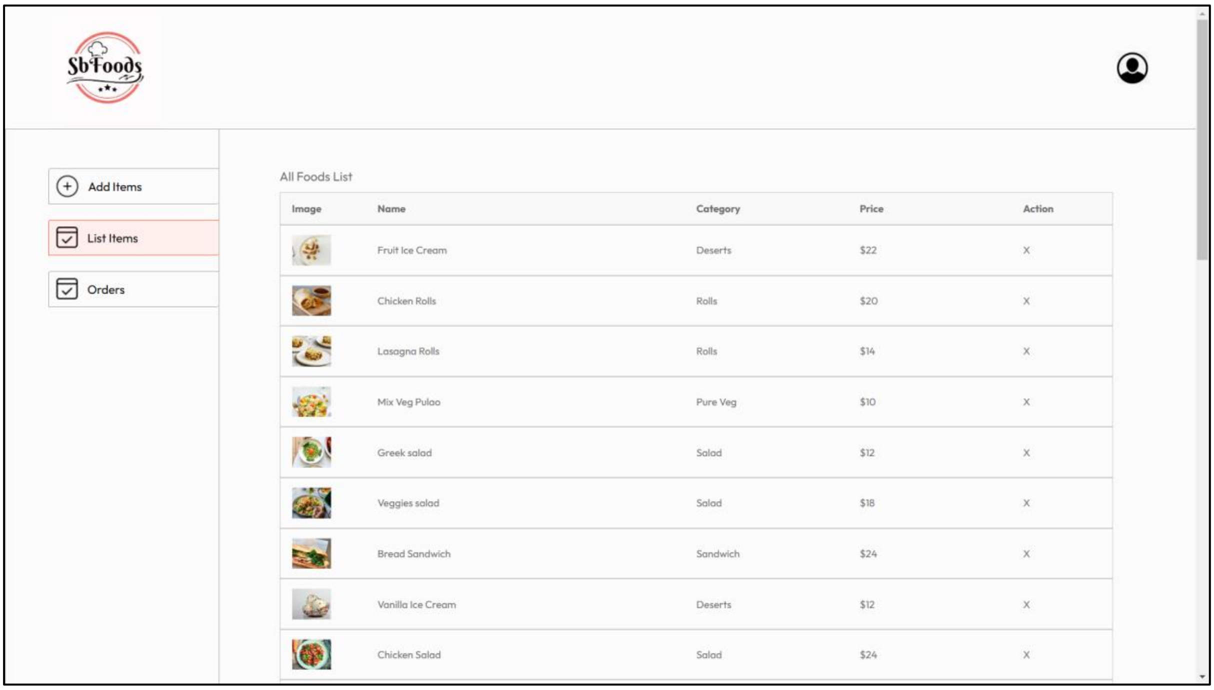
Product description

Write content here

Product category: Salad

Product price: $20

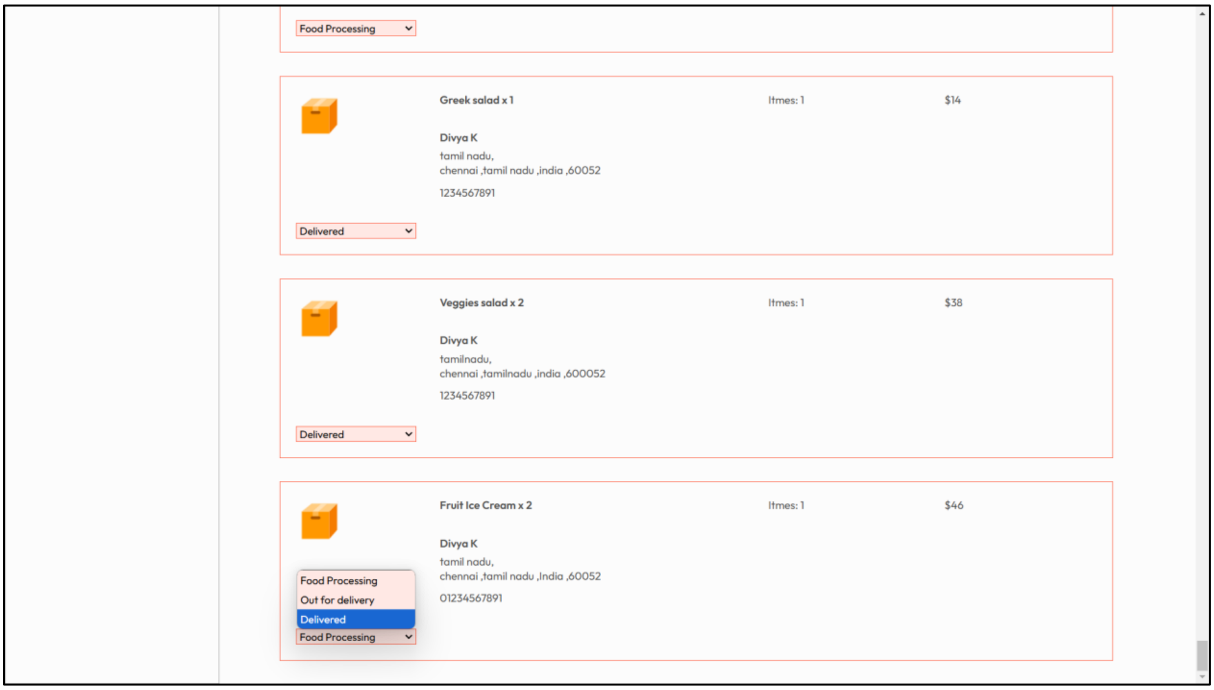
LIST ITEMS



All Foods List

Image	Name	Category	Price	Action
	Fruit Ice Cream	Deserts	$22	X
	Chicken Rolls	Rolls	$20	X
	Lasagna Rolls	Rolls	$14	X
	Mix Veg Pulao	Pure Veg	$10	X
	Greek salad	Salad	$12	X
	Veggies salad	Salad	$18	X
	Bread Sandwich	Sandwich	$24	X
	Vanilla Ice Cream	Deserts	$12	X
	Chicken Salad	Salad	$24	X

ORDERS



Food Processing

Greek salad x 1 Itmes: 1 $14

Divya K
tamil nadu,
chennai ,tamil nadu ,india ,60052
1234567891

Delivered

Veggies salad x 2 Itmes: 1 $38

Divya K
tamilnadu,
chennai ,tamilnadu ,india ,600052
1234567891

Delivered

Fruit Ice Cream x 2 Itmes: 1 $46

Divya K
tamil nadu,
chennai ,tamil nadu ,India ,60052
01234567891

Food Processing

11. CONCLUSION

This project successfully demonstrates a scalable and user-friendly food ordering app. The use of the MERN stack provides both performance and flexibility, making it an ideal solution for food delivery applications. With planned enhancements, the app can evolve into a robust, real-world solution for restaurant ordering systems.