Full Stack Development with MERN

OrderOnTheGo: Your On-Demand Food Ordering Solution

1. Introduction :

1.1 Purpose The purpose of the DHK FOODIES project is to develop a highly interactive and user-friendly frontend prototype for a modern food ordering system. The application aims to eliminate the common friction points in online food ordering by providing a seamless, responsive, and visually engaging platform for users to discover restaurants, filter choices, and manage their orders.

2. Ideation Phase

2.1 Problem Statement Many existing food ordering platforms suffer from cluttered interfaces, inadequate filtering options, and a disjointed user experience from browsing to checkout. This can lead to user frustration and cart abandonment. The DHK FOODIES app addresses this by focusing on a clean, intuitive design and powerful user-centric features.

2.2 Empathy Map Canvas (User Perspective)

- **Wants:** A wide variety of restaurants and food choices, easy-to-use filters, a simple and quick checkout process, the ability to see past orders.
- **Frustrations:** Confusing menus, lack of sorting options (e.g., by price), complicated payment forms, losing order history.
- **Needs:** A reliable way to track orders and a personalized experience (e.g., a user profile).

2.3 Brainstorming The ideation process led to the following core features:

- Real-time filtering and sorting of restaurants.
- Persistent user sessions with profiles and order history.
- A multi-step, mock checkout process to simulate a real-world transaction.
- A fully responsive design for both mobile and desktop users.
- Dynamic menu display and interactive cart management.

3. Requirement Analysis

3.1 Customer Journey Map Registration -> Login -> Browse Restaurants -> Apply Filters & Sort -> View Menu -> Add/Update Items in Cart -> Proceed to Checkout -> Fill Delivery Address -> Select Payment Method -> Receive Order Confirmation -> View Profile & Order History.

3.2 Solution Requirements

- User Registration & Login system.

- Secure password handling with a show/hide toggle.
- Advanced filtering (by diet, categories) and sorting (by price).
- Dynamic menu population based on the selected restaurant.
- Stateful shopping cart.
- Persistent storage of user data and order history using localStorage.

4. Technology Stack

- **Frontend:** React (via CDN), Tailwind CSS (via CDN), JavaScript (ES6+).
- **Transpiler:** Babel (via CDN) to convert modern JSX into browser-compatible JavaScript in real-time.
- **Backend (Simulated):** Browser localStorage is used to mimic a database for storing user accounts and order history, making it a persistent single-user experience.
- **Database:** MongoDB (Planned for future full-stack implementation).

5. Project Design

5.1 Problem-Solution Fit The application directly addresses user frustrations by providing powerful filtering, sorting, and a streamlined checkout process, all within a clean and responsive interface.

5.2 Proposed Solution A role-based web application (initially for the 'User' role) that provides a comprehensive frontend experience for food ordering. The app is built as a Single-Page Application (SPA) for speed and interactivity.

5.3 Solution Architecture Frontend (React) <-> Browser localStorage (Simulated Database)

6. Project Planning & Scheduling

Week	Task
1	Requirements & UI Mockups
2	Core Component Development (Restaurants, Menu)
3	Authentication & User Profile
4	Cart, Checkout Flow & Order History
5	Advanced Filtering & Sorting
6	Final Testing & Documentation

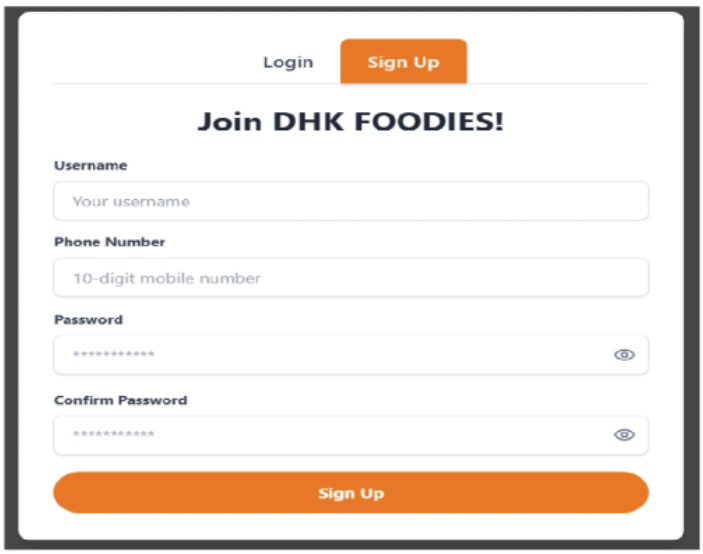
7. Functional and Performance Testing

- **Performance Testing Tools:** Google Chrome DevTools (Lighthouse, Performance monitor).

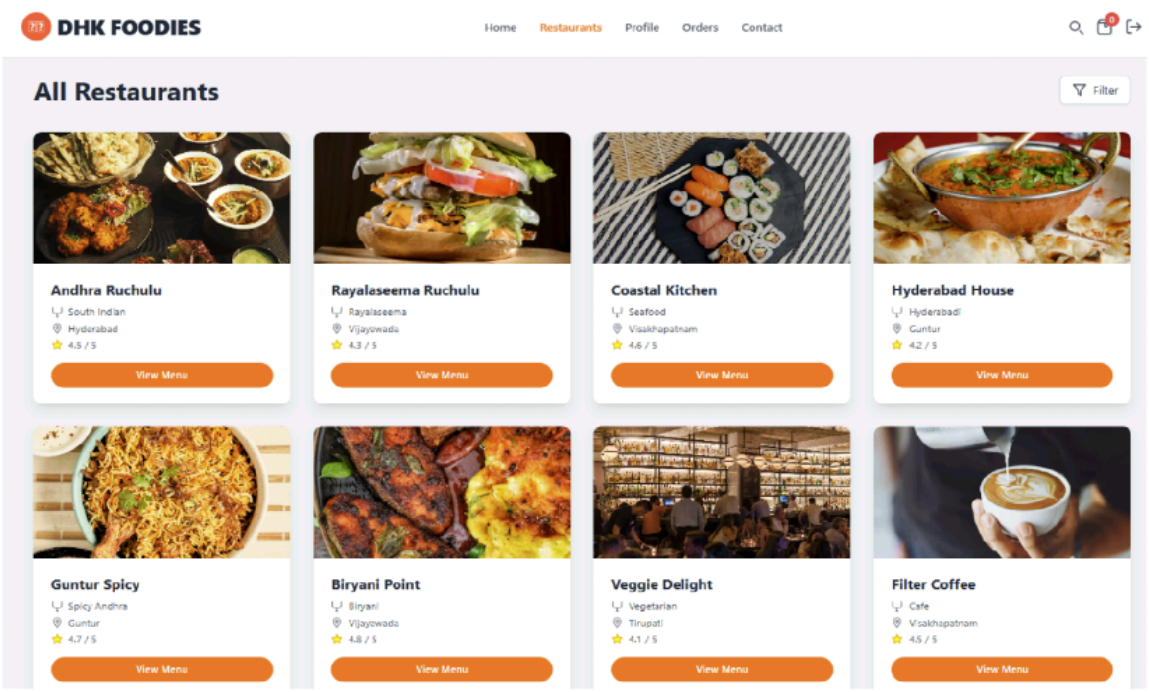
- **API Response Time:** All data is loaded from a local mock object, resulting in near-instantaneous load times.
- **Database Query Time:** N/A (Data is accessed from localStorage).
- **Frontend Testing:** Manually tested across modern browsers (Chrome, Firefox) for responsiveness and interactive functionality.

8.Results:

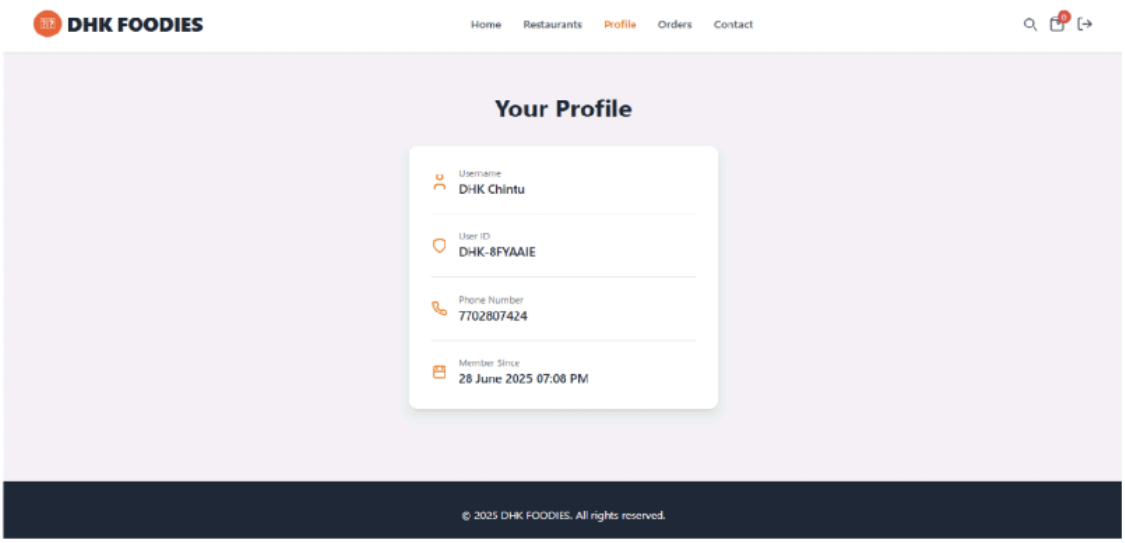
1) User Sign Up & Login Page



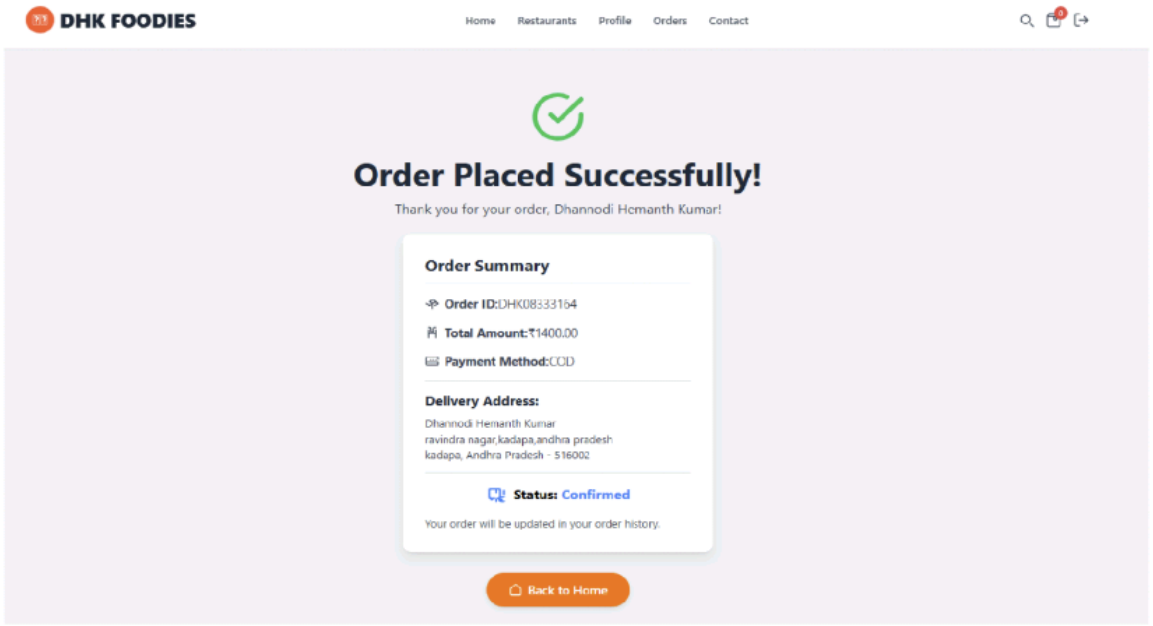
2.Restaurant & Filter Page



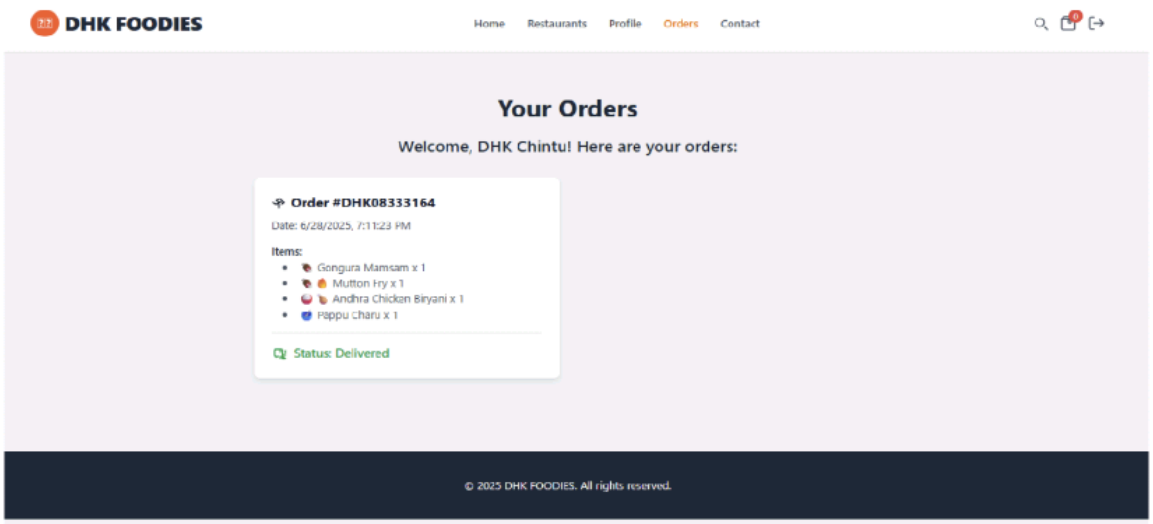
3.User Profile Page



4.Payments & Order Status



5.Order History



9. Advantages & Disadvantages

Advantages:

- **Highly Interactive UI:** React provides a fast and responsive user experience.
- **No Backend Dependency:** As a prototype, it runs in any modern browser without needing a server or database setup.
- **Persistent Data:** User accounts and order history are saved in the browser, providing a continuous experience for returning users on the same device.
- **Simple & Scalable Codebase:** The component-based architecture is easy to understand and extend.

Disadvantages:

- **No Real-time Database:** Data is stored locally and cannot be shared across different devices or users.
- **No Payment Gateway:** The payment process is a simulation and does not handle real financial transactions.
- **Basic Authentication:** The current login system does not use secure methods like JWT and is for demonstration purposes only.

10. Conclusion

The DHK FOODIES app successfully demonstrates a modern, feature-rich frontend for a food ordering system. It meets all the initial requirements, providing a seamless user journey from browsing to checkout, complete with advanced filtering and persistent user profiles.

11. Future Scope

- **Full-Stack Implementation:** Integrate a Node.js/Express.js backend and MongoDB database to enable real-time, multi-user functionality.
- **JWT Authentication:** Implement JSON Web Tokens for a more secure and robust authentication system.
- **Payment Gateway Integration:** Integrate a real payment gateway like Stripe or Razorpay.
- **Mobile App:** Develop a native mobile application using React Native.
- **SMS/Email Notifications:** Add real-time order status notifications.