**SB Foods - Food Ordering Application - Report**

**1. Introduction**

**Project Title:** SB Foods - Food Ordering Application

**Team Members:**

* Harish.N
* Anandha Krishnan.V
* Sanjay Kumar.S
* Mani Kumar.G.K
* Ahmed Thanveer Nishad.M

**2. Project Overview**

**Purpose:**  
FoodMine is a comprehensive food ordering platform designed to streamline the process of browsing, ordering, and managing food orders. The application offers an intuitive user experience and robust administrative tools, making food ordering simple for users and efficient for restaurant operators.

Features:

Menu Display: Interactive browsing of food items with search and filtering capabilities.

Item Details: Comprehensive descriptions, pricing, and dietary information.

Order Management: Users can add items to the cart and complete the checkout process.

Authentication: Secure user login and registration.

Real-time Order Tracking : Users can track the status of their orders.

**3. Architecture**

**Frontend:**

* Built using React.js with state management through Redux Toolkit.
* **Key Components**:
  + **Menu Component**: Displays available food items with filter and search options.
  + **Cart System**: Manages items and calculates totals.
  + **Authentication**: Uses JWT for secure login and signup.
  + **Admin Panel**: Tools for restaurant management.

**Backend:**

* Developed using Node.js and Express.js for scalable API services.
* **API Endpoints**:
  + /menu for fetching and managing food items.
  + /orders for processing user orders.
  + /users for authentication and profile management.
* **Middleware**: Authentication and error-handling using JWT.

**Database**:

* **MongoDB** stores:
  + **Menu**: Details of food items.
  + **Users**: Authentication data and order history.
  + **Orders**: Records of user orders and statuses.

**4. Setup Instructions**

**Installation**:

1. Clone the repository:

<https://github.com/Harish032004/nm-project.git>

**Frontend Setup**:

1. Navigate to the frontend directory:

cd frontend

1. Install dependencies:

npm install

1. Start the development server:

npm run dev

**Backend Setup**:

1. Navigate to the backend directory:

cd backend

1. Install dependencies:

npm install

1. Start the backend server:

npm run dev

**Access the Application**:

* Open your browser and navigate to http://localhost:3000.

**5. Folder Structure**

**Client**:

* src/components: Reusable components (e.g., Navbar, FoodItem, Cart).
* src/pages: Application pages like Home, Cart, and AdminPanel.
* src/redux: Manages state for menu, authentication, and orders.

**Server**:

* routes: RESTful endpoints for menu, orders, and user management.
* controllers: Business logic for backend operations.
* models: MongoDB schemas for Menu, Users, and Orders.
* middleware: JWT-based authentication and error handling.

**6. API Documentation**

**Endpoints**:

* **GET /menu**: Retrieves all available food items.
* **POST /orders**: Places a new order.
* **GET /orders/**

: Fetches order details by ID.

**Response Examples**:

{

"message": "Order placed successfully",

"orderId": "12345"

}

**7. User Interface**

**Screens**:

* **Home Page**: Lists food items with filters.
* **Details Page**: Shows details of selected food items.
* **Cart Page**: Displays selected items and checkout options.
* **Admin Dashboard**: Allows CRUD operations for managing items and orders.

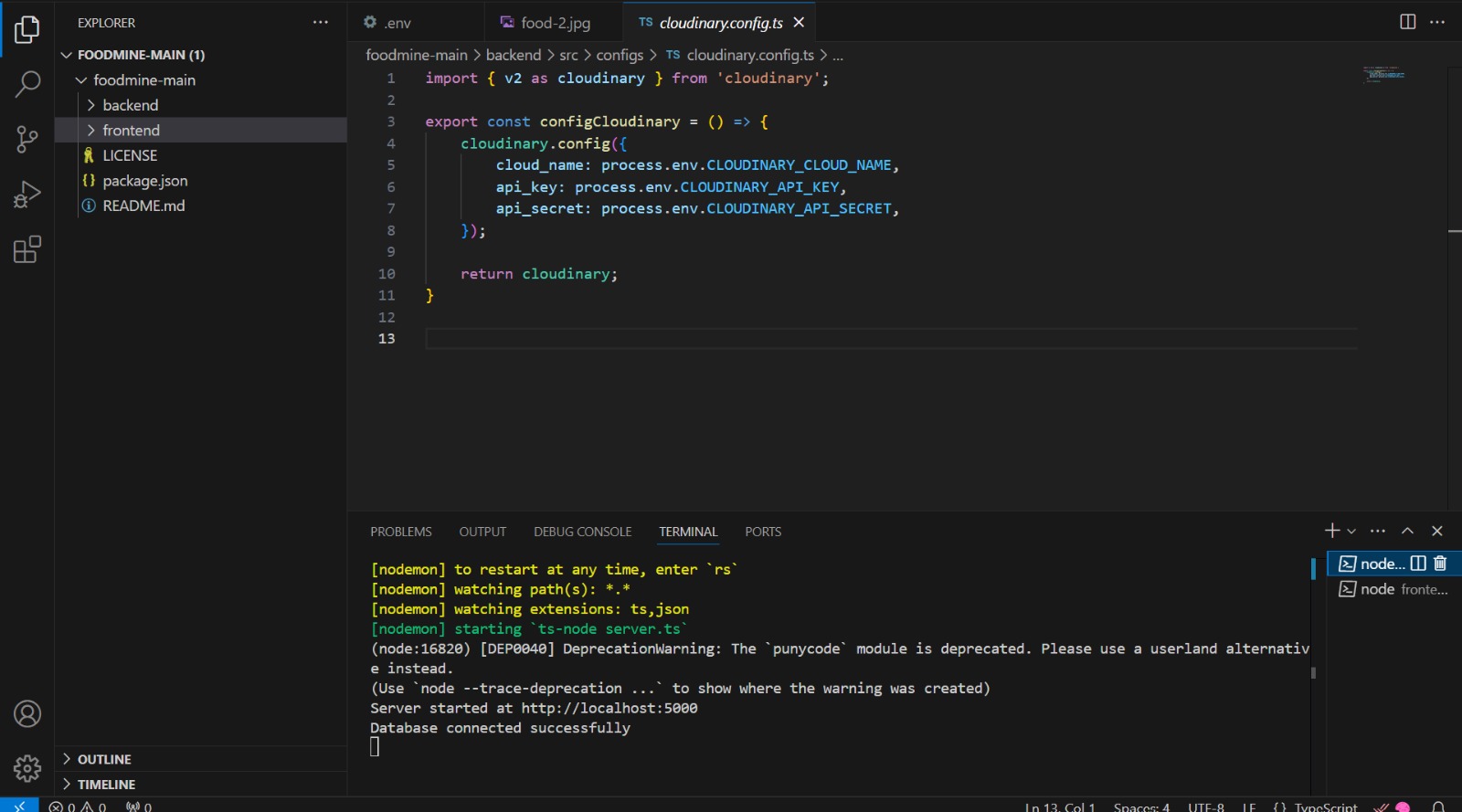
**8. Known Issues**

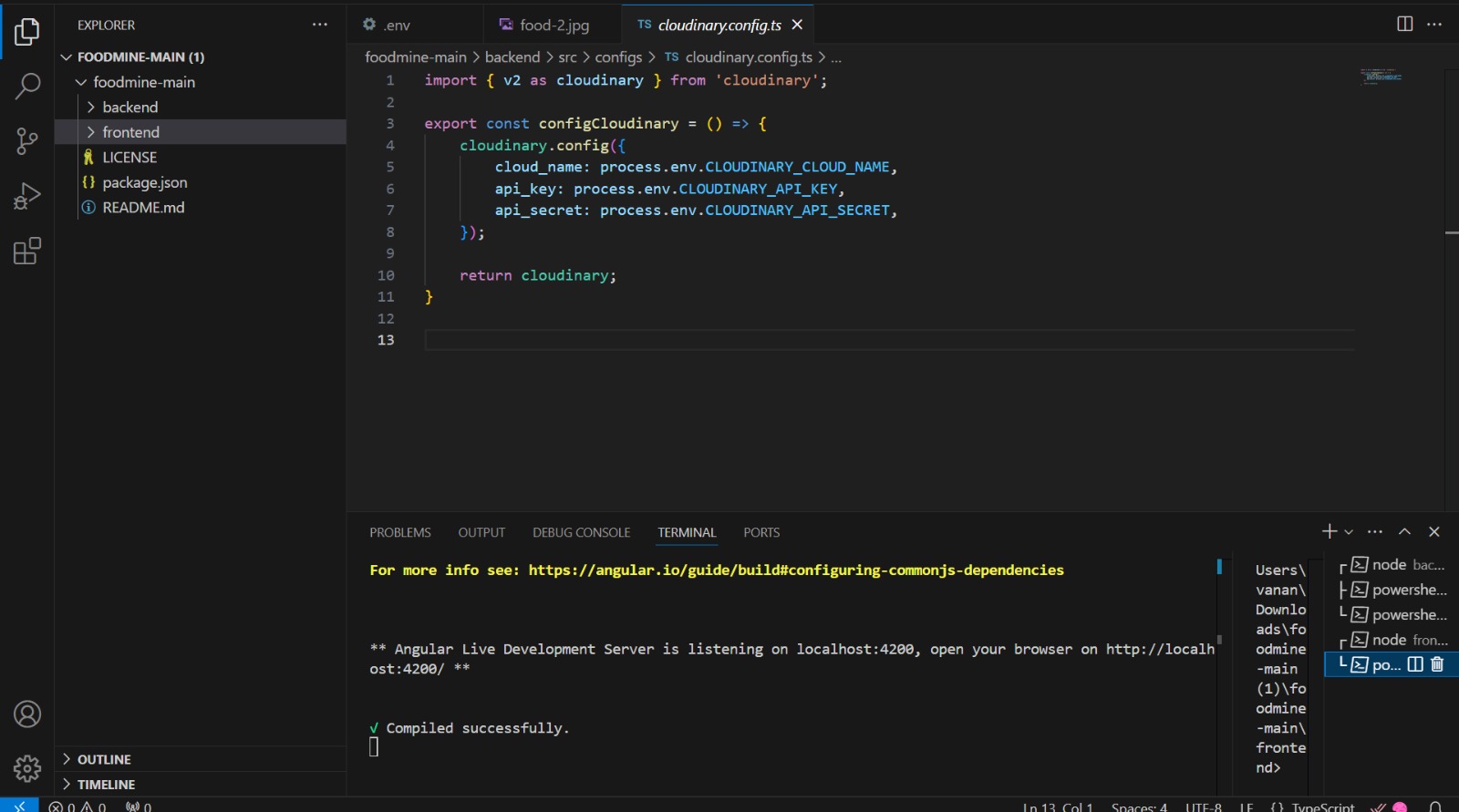
* Minor UI bugs in mobile views.
* Backend optimization needed for handling high volumes of data.

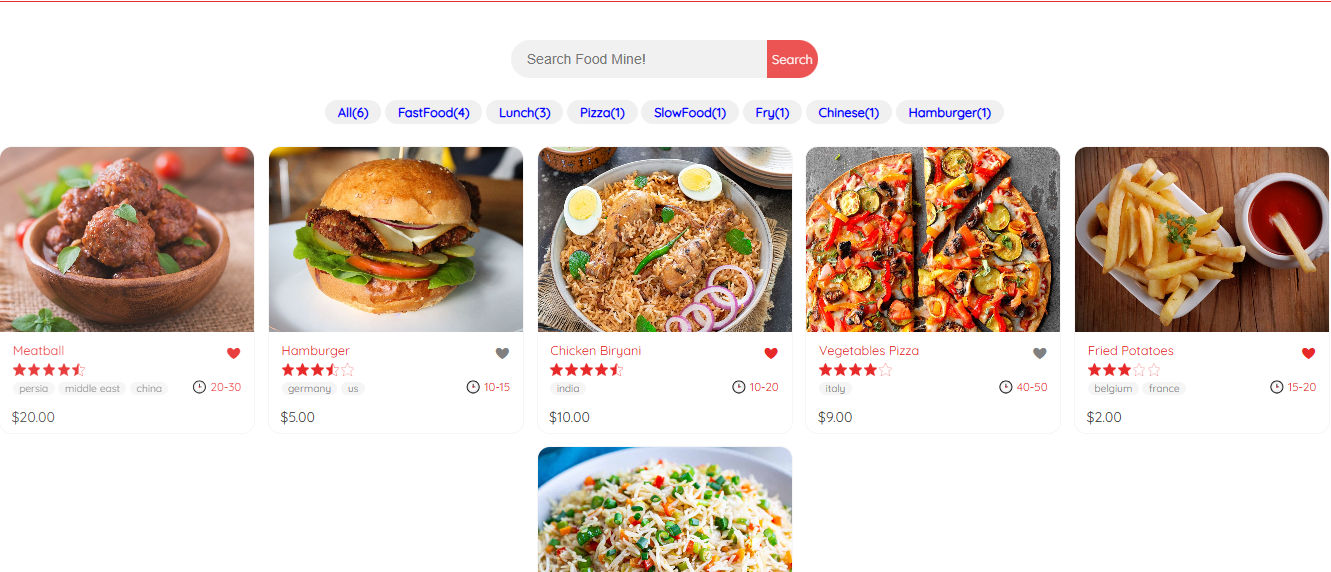
**9. Future Enhancements**

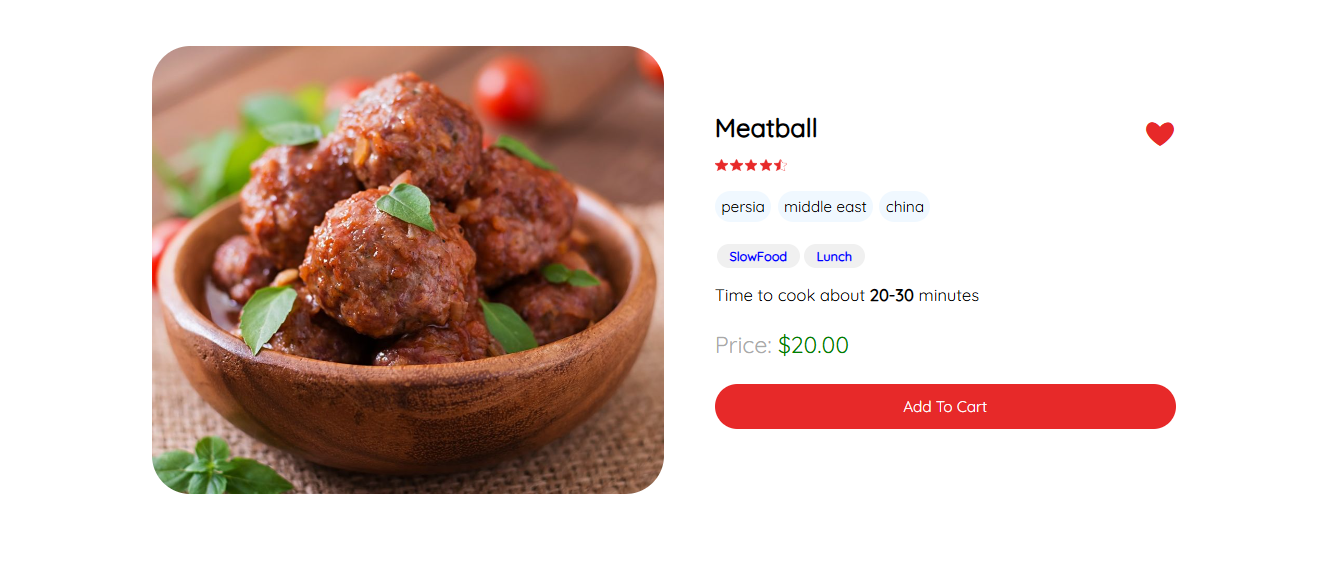
* Integrate user reviews and ratings for food items.
* Implement push notifications for order updates.
* Create a mobile version using React Native.

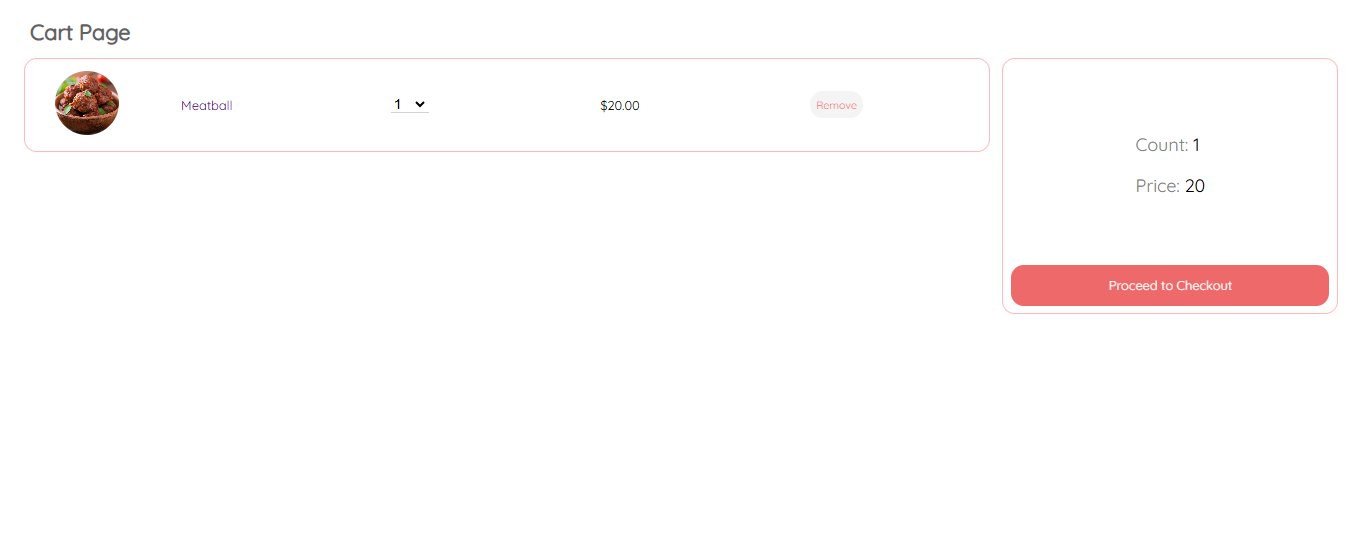
**10. Screenshots or Demo:**

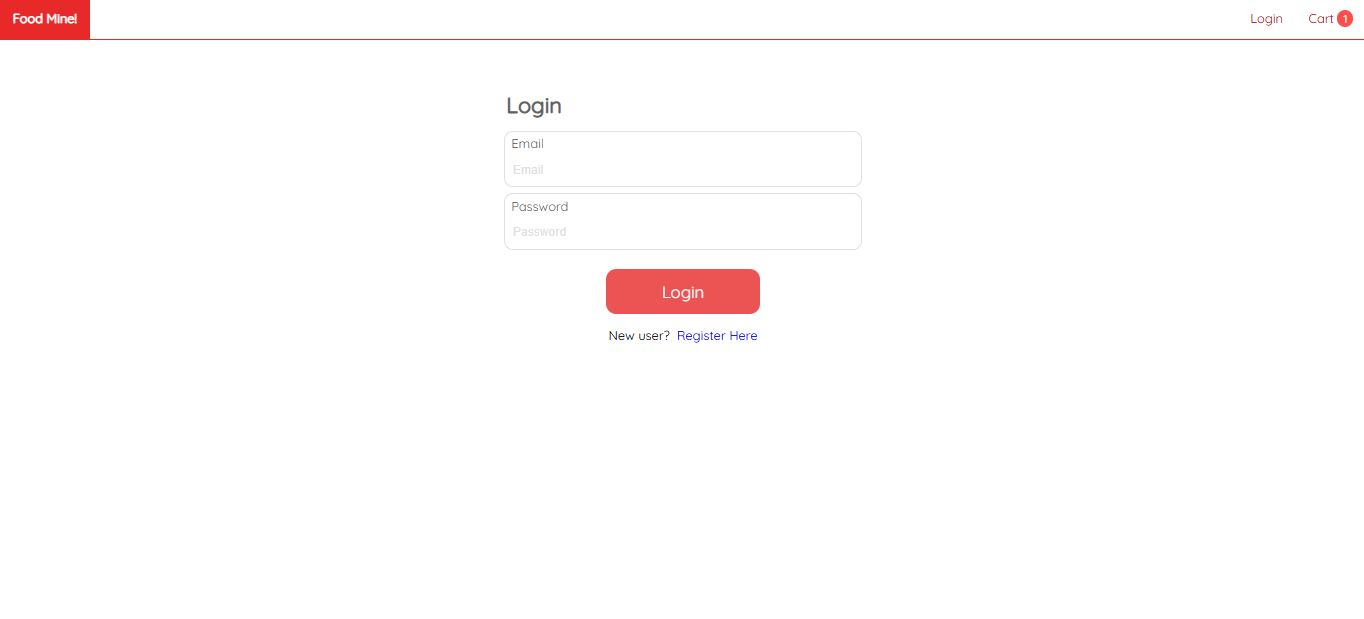


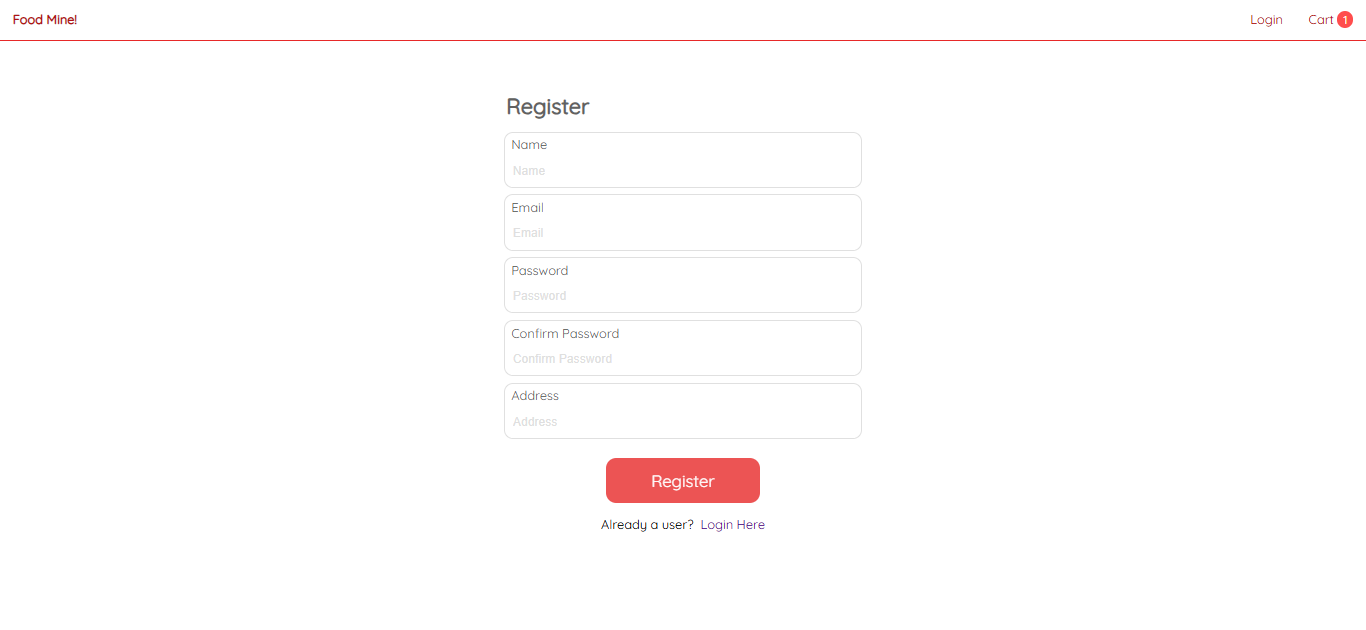












Demo video link: