A close-up of a sign

Description automatically generated

**MERN Stack Using Food Ordering App**

**Department of Computer Science and Engineering**

Submitted By

**1. SYED GHULAM RASOOL 110121104095**

**2. SAIF RAHMAN 110121104081**

**3. MOHAMMED SAHITH 110121104064**

**4. MOHAMMED AYMAAN 110121104061**

**Introduction:**

The **Food Ordering App** is an innovative solution designed to address a growing need in the fast-paced, convenience-driven world of today—access to food delivery at any time, especially during late hours. As more individuals face hectic schedules, work late into the night, or find themselves with limited time and energy to prepare meals, the demand for quick and reliable food delivery services has skyrocketed.

In particular, **college students**, like Lisa in our project scenario, are often faced with late-night cravings while juggling academic responsibilities. Whether it’s working on assignments or pulling an all-nighter for exams, cooking or going out to get food can feel like too much of an interruption. This app solves that problem by offering a convenient and user-friendly platform that provides **late-night food ordering services**—ensuring users can enjoy their favourite meals with a few simple taps on their smartphones.

Built using the **MERN stack** (MongoDB, Express.js, React.js, Node.js), this Food Ordering App is designed to offer a seamless, dynamic, and scalable experience for users. The app not only makes it easy to browse a variety of local restaurants and food options but also allows users to place orders, make payments, and track deliveries in real time. Its design emphasizes **ease of use, security, and speed**, ensuring users like Lisa can satisfy their late-night hunger without compromising their workflow or peace of mind.

This report details the **Food Ordering App’s** development process, the technologies used, and its key features, along with challenges faced during its creation and solutions implemented. The app is intended to cater to users' needs, particularly when conventional dining options are unavailable, by offering an efficient and enjoyable food ordering experience.

**Scenario:**

**Meet Lisa** – a college student who often finds herself working late into the night. Whether it’s finishing a research paper, preparing for exams, or completing assignments, Lisa’s late-night study sessions are routine. However, with her busy schedule, there’s one issue that always arises: hunger.

After hours of work, Lisa’s stomach begins to growl, signaling the inevitable — she skipped dinner earlier in the day and now has no energy or time to cook. Venturing out to get food isn’t a viable option, as it would take too much time and effort. Cooking would also break her concentration, making it harder to get back into her work.

**The Solution: A Late-Night Food Ordering App**

Enter the **Food Ordering App**, the perfect solution to Lisa’s problem.

1. **Accessing the App**:  
   Lisa opens the **Food Ordering App** on her smartphone. The app immediately catches her attention with an easy-to-navigate interface. She finds a dedicated section for **late-night delivery options**, where restaurants open for late orders are highlighted.
2. **Browsing Options**:  
   Lisa scrolls through a list of local restaurants, each showing their menu items along with reviews and ratings. She’s pleased to find several comfort food options—perfect for a study snack. She quickly browses through menus, narrowing her choices based on cravings and her budget.
3. **Choosing the Meal**:  
   After some thought, Lisa decides to order a bowl of **chicken noodle soup** with a side of **garlic bread**—a perfect combination to soothe her hunger and keep her energized. The app allows her to easily customize her order (adding extra bread) and move on to the next step.
4. **Adding to Cart**:  
   With a few taps, Lisa adds her meal to the **cart**. The app auto-fills her address based on previous orders, saving her time. She double-checks her cart to ensure everything is correct.
5. **Checkout and Payment**:  
   After confirming her order, Lisa proceeds to checkout. The app provides multiple **payment options**, including credit card and digital wallets, ensuring flexibility. She selects her preferred payment method, enters payment details, and confirms the purchase.
6. **Order Confirmation and Tracking**:  
   Within seconds, Lisa receives an **order confirmation** on the screen with a clear **estimated delivery time**. The app also sends a push notification to confirm the order, allowing her to continue working uninterrupted.
7. **Order Delivery**:  
   While Lisa continues with her assignment, she receives real-time **tracking updates** on the delivery status of her order. She gets a notification when the food is out for delivery and another when the delivery is about to arrive.

**Project Overview:**

The **Food Ordering App** is a web and mobile-based application designed to provide users with a convenient and seamless platform for ordering food, especially during late-night hours. The app is built using the **MERN stack** (MongoDB, Express.js, React.js, Node.js), which ensures a modern, efficient, and scalable architecture. The project aims to cater to the needs of college students, professionals, and anyone who finds themselves in need of food delivery after regular business hours, as in the case of the user scenario described with Lisa.

**Objectives:**

The primary objectives of the **Food Ordering App** are to:

* **Enable easy food discovery**: Allow users to browse menus from a variety of restaurants that offer late-night delivery options.
* **Streamline the ordering process**: Make it simple for users to select, customize, and order food with a few taps or clicks.
* **Provide multiple payment options**: Ensure users can pay securely using a variety of methods, including credit/debit cards and digital wallets.
* **Offer real-time updates**: Allow users to track the status of their orders from placement to delivery.
* **Ensure convenience and flexibility**: Make it possible for users to place orders quickly without interrupting their daily activities.

**Key Features:**

1. **Late-Night Restaurant Listings**: The app focuses on **restaurants that are open late at night** or have 24/7 delivery, which is particularly beneficial for users who may need food during unconventional hours. Users can easily filter for **late-night** or **open-now** options, ensuring they only see restaurants available at that time.
2. **Restaurant Menu Browsing**: Users can browse through restaurant menus, view detailed descriptions of dishes, and check ratings and reviews from previous customers. This feature helps users make informed decisions about what to order based on food quality and others' experiences.
3. **Personalized Cart**: The app allows users to **add items to their cart**, modify quantities, and remove items with a few simple clicks. It also supports **customization options** (such as adding extra toppings or choosing spice levels), making the ordering process personalized to each user’s preferences.
4. **User Profiles and Order History**: Users can **create an account**, log in, and manage their profiles. They can view their **order history**, **favorite restaurants**, and **saved addresses** to make repeat orders faster and more convenient.
5. **Secure Payment System**: The app integrates with a secure **payment gateway** to support multiple payment methods, such as credit/debit cards, mobile wallets (e.g., PayPal, Apple Pay), and other common online payment options. This ensures a smooth, secure transaction process for users.
6. **Real-Time Order Tracking**: After placing an order, users can track their **order status** in real time. Notifications inform them when the order is confirmed, when the food is being prepared, dispatched, and when it is out for delivery, giving them an estimated arrival time. This feature improves user satisfaction by keeping them informed throughout the entire delivery process.
7. **Push Notifications**: The app sends timely **push notifications** to users, updating them about their order status. These notifications also include promotional offers, order confirmation, delivery time estimates, and any potential delays.
8. **Search and Filters**: Users can **search for specific dishes** or restaurants by name, cuisine type, or even dietary preferences. Filters can help users narrow down choices based on factors such as price range, food category (e.g., vegetarian, vegan, comfort food), or delivery speed.

**Technological Stack**

* **MongoDB**: A NoSQL database that provides flexibility and scalability, essential for handling diverse and evolving data such as user profiles, orders, and restaurant menus.
* **Express.js**: A minimalist framework for Node.js that facilitates the development of server-side logic, API routes, and database interaction, making the backend development more efficient and organized.
* **React.js**: A powerful JavaScript library for building the frontend user interface, allowing for a responsive and dynamic experience. React makes the app interactive and fast, crucial for a food ordering app that requires frequent updates (e.g., updating order status, adding items to the cart).
* **Node.js**: A JavaScript runtime that enables building fast, scalable, and real-time applications. It handles the backend logic and server-side operations of the app.
* **Stripe/PayPal API (for payment processing)**: A third-party API integrated to handle secure payments through credit cards, debit cards, or digital wallets, ensuring safe financial transactions.

**User Flow:**

1. **Sign Up / Login**: Users can create an account or log in to their existing account to access personalized features such as order history and saved addresses.
2. **Browse and Filter Restaurants**: Once logged in, users can browse available restaurants, filtering by category, price range, and late-night availability.
3. **Add Items to Cart**: Users can browse menus and add items to their cart. They can also customize the order based on their preferences.
4. **Payment and Order Confirmation**: After reviewing their order, users proceed to payment. Once the payment is successful, the order is confirmed, and the user receives an estimated delivery time.
5. **Track Order Status**: Users can track their order in real time. Notifications will update them on the progress of their order, from preparation to delivery.
6. **Delivery and Feedback**: After receiving their order, users are encouraged to leave reviews and feedback on the food and the delivery experience.

**E-R Diagram:**

A diagram of a company

Description automatically generated

**Development Process**

1. **Planning and Design**:  
   The first phase of development involved understanding user needs through the scenario of "late-night cravings" and designing the app to be user-centric, with particular focus on:
   * Ease of navigation
   * Quick order placement
   * Late-night availability
2. **Frontend Development**:  
   Using React, we built the user interface with components for browsing restaurants, adding items to the cart, and managing user profiles. React Router was used for efficient routing and navigation between pages.
3. **Backend Development**:  
   The backend API was created using Node.js and Express.js to manage requests such as fetching restaurant menus, placing orders, and processing payments. We integrated a third-party payment gateway for secure transactions.
4. **Database Schema Design**:  
   The database schema was designed to efficiently store and retrieve information related to users, restaurants, menus, and orders. MongoDB's document model was leveraged for flexible, scalable storage.
5. **Testing and Debugging**:  
   The app underwent several rounds of testing, including functional testing of individual features, integration testing of frontend-backend communication, and user acceptance testing (UAT) to ensure the app met the needs of late-night users like Lisa.
6. **Deployment**:  
   The app was deployed to a cloud server (e.g., AWS or Heroku) to ensure scalability and availability. The frontend was hosted on a service like Netlify or Vercel, while the backend was deployed on Heroku.

**Challenges and Solutions**

* **Late-Night Availability of Restaurants**:  
  Ensuring that the app displayed only those restaurants open late at night required dynamic filtering based on operational hours. This was addressed by adding operational hours as metadata for each restaurant in the database.
* **Payment Security**:  
  Handling secure payment transactions was a critical challenge. We integrated a third-party payment gateway that complies with PCI-DSS standards to ensure data security.
* **Real-Time Order Tracking**:  
  Implementing real-time order tracking required integrating WebSockets to push updates to the frontend as the status of the order changed (e.g., preparing, dispatched, delivered).

**Results and Future Enhancements:**

* **Successful Implementation**:  
  The app successfully meets the needs of users like Lisa, offering a smooth experience for browsing late-night food options, placing orders, and receiving deliveries.
* **Future Features**:  
  In the future, we plan to add:
  + **Ratings and Reviews**: Allow users to leave feedback for restaurants and dishes.
  + **Personalized Recommendations**: Leverage machine learning to suggest food items based on past orders.
  + **Multi-language Support**: Provide a multilingual interface to cater to diverse users.
  + **AI-Driven Chatbot**: A chatbot to assist users with order placement and troubleshooting.

**Conclusion:**

The **Food Ordering App** built using the MERN stack is a robust, scalable solution to address late-night food delivery needs. By focusing on user convenience and using modern web technologies, the app has successfully created a seamless experience for users like Lisa, enabling them to satisfy cravings without interrupting their routine. With further enhancements, this app has the potential to become an indispensable tool for late-night food lovers everywhere.