**Online Food Ordering Solution**

**Project submitted to the**

**APSCHE**

**Bachelor of Technology**

**In**

**Computer Science and Engineering**

**School of Engineering and Sciences**

**Submitted By**

**Punyamantula D S Aravind Kumar**

**Vykuntam Eswara rao**

**Yadla Mahima Rani**

**M Sriya Vaishnavi**

**Under the guidance of**

**M.Anji Sir**

**July 2025**

# INTRODUCTION

In today’s digital age, the demand for convenient and quick food delivery has surged, especially in urban and suburban regions. With changing lifestyles, longer work hours, and a growing dependence on technology, food delivery platforms have become a vital service for millions. **Orderonthego** is a comprehensive food ordering and delivery solution built using the MERN stack (MongoDB, Express.js, React.js, and Node.js) aimed at bridging the gap between restaurants and customers through a seamless digital experience.

The core idea of Orderonthego is to simplify the food ordering process—from discovering restaurants to placing an order and tracking it in real time. Whether users want to browse local cuisine, place group orders, or reorder their favorite meals, the application ensures a hassle-free experience. It integrates real-time notifications, secure payments, and live tracking to deliver a modern, user-centric platform.

For restaurant owners, the application serves as a powerful business tool. They can manage their menu, track orders, control availability, and gain insights into customer preferences. For delivery partners, the system provides efficient order allocation, delivery status updates, and route tracking—resulting in faster and more organized logistics.

With the scalability of the MERN stack, **Orderonthego** is designed not just for small-scale use but also for large-scale deployment where performance, reliability, and user satisfaction are paramount.

# DESCRIPTION

**Orderonthego** is a robust, scalable, and intuitive food ordering platform designed to meet the dynamic needs of modern consumers, restaurant owners, and delivery personnel. Built using the powerful **MERN stack** (MongoDB, Express.js, React.js, Node.js), the application streamlines the entire food ordering lifecycle—from discovering restaurants to delivering the meal at the user’s doorstep.

At its core, Orderonthego is a three-sided platform:

#### 1. ****For Customers****

* Users can sign up, browse menus, apply filters (e.g., vegetarian, price, ratings), and place orders from a wide range of local restaurants.
* Features like **real-time order tracking**, **digital wallets**, **discount coupons**, and **order history** enhance the user experience.
* Customers are notified about order status through SMS, email, or in-app push notifications.

#### 2. ****For Restaurant Owners****

* Restaurant partners have access to a dedicated **dashboard** to manage their profile, menu items, pricing, availability, and incoming orders.
* They receive notifications for new orders, can update status (accepted, preparing, ready), and manage delivery coordination.
* Analytics tools provide insights into **best-selling dishes**, **peak order times**, and **customer reviews**, enabling data-driven decisions.

#### 3. ****For Delivery Partners****

* Delivery agents are assigned orders based on location and availability.
* They can view pickup and drop locations, estimated delivery time, and directions.
* Real-time updates help ensure efficient logistics and delivery success.

#### 4. ****For Admin****

* The admin panel provides a bird’s-eye view of the platform.
* Admins approve or verify restaurant and delivery registrations, monitor user activity, handle disputes, and generate reports.
* They can manage system configuration, broadcast announcements, and ensure data integrity and compliance.

# FEATURES

**Orderonthego** offers a wide array of features tailored to improve user experience, optimize restaurant operations, and simplify food delivery logistics. These features are designed with scalability, usability, and real-time functionality in mind.

#### 1. ****Customer Panel Features****

Customers interact with the platform through an intuitive and visually appealing interface that makes ordering food simple and efficient.

* **Personalized Dashboard**  
  View restaurants nearby based on location and filters like cuisine, rating, and delivery time.
* **Cart Management**  
  Add, edit, or remove items; update quantities; view total pricing including taxes and delivery charges.
* **Multiple Payment Options**  
  Supports UPI, debit/credit cards, net banking, and wallets with secure transaction gateways.
* **Real-Time Order Tracking**  
  Track each stage of your order from "Preparing" to "Out for Delivery" on a map interface.
* **Feedback System**  
  Submit reviews and ratings for restaurants and individual menu items.

#### 2. ****Restaurant Panel Features****

Designed for restaurant managers/owners to gain full control over their digital presence.

* **Order Notification System**  
  Receive and accept new orders instantly with sound/vibration alerts.
* **Order Status Update**  
  Update status: “Accepted,” “Preparing,” “Ready,” etc., which reflects instantly on the customer’s screen.
* **Sales Analytics**  
  View daily/weekly/monthly sales, popular items, customer feedback, and peak hours.
* **Profile & Store Settings**  
  Update operating hours, contact info, delivery zones, and promotions.

#### 3. ****Delivery Partner Panel****

Streamlines the process of food delivery from restaurants to customers.

* **Assigned Deliveries View**  
  Shows pickup and drop locations, along with optimized route suggestions.
* **Status Management**  
  Update order status: “Picked Up,” “On the Way,” “Delivered.”
* **Performance Metrics**  
  Shows delivery success rate, on-time rate, and customer feedback.

#### 4. ****Admin Panel Features****

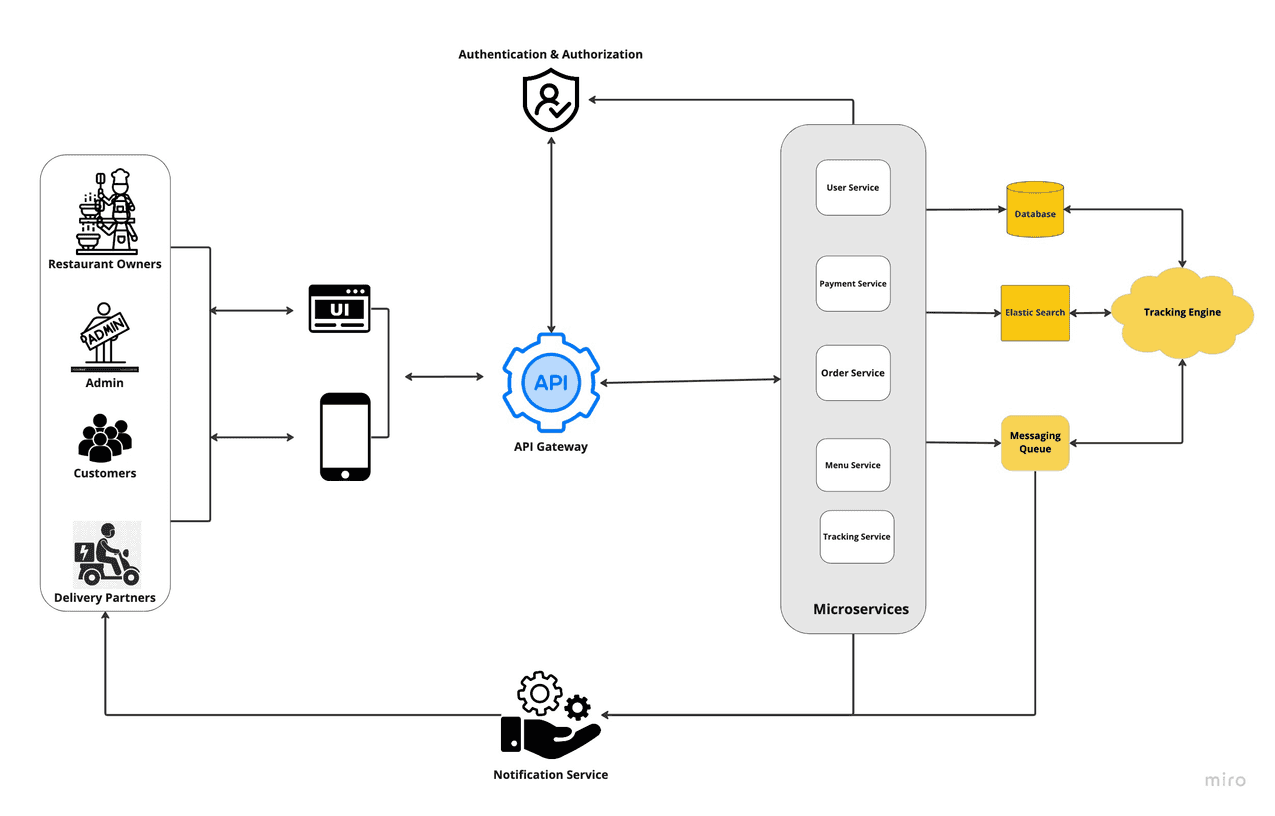
Centralized dashboard for managing all operations, users, and analytics.

* **User Verification**  
  Approve/ban restaurants or delivery partners based on documents and activity.
* **Complaint Resolution Center**  
  View, respond, and log user complaints or service issues.
* **Broadcast Notifications**  
  Send promotional or system-wide announcements to all users or specific groups.
* **System Configuration**  
  Manage platform settings such as commission percentages, refund policies, and peak time charges.

#### 5. ****General System Features****

* **Responsive Design:** Works seamlessly across smartphones, tablets, and desktops.
* **Search and Filter Options:** Instant search by dish, restaurant name, or location.
* **Favorites and Wishlists:** Users can mark restaurants and items for easy access later.
* **Offers and Coupons:** Apply promo codes during checkout for discounts and cashback.

# TECHNICAL ARCHITECTURE

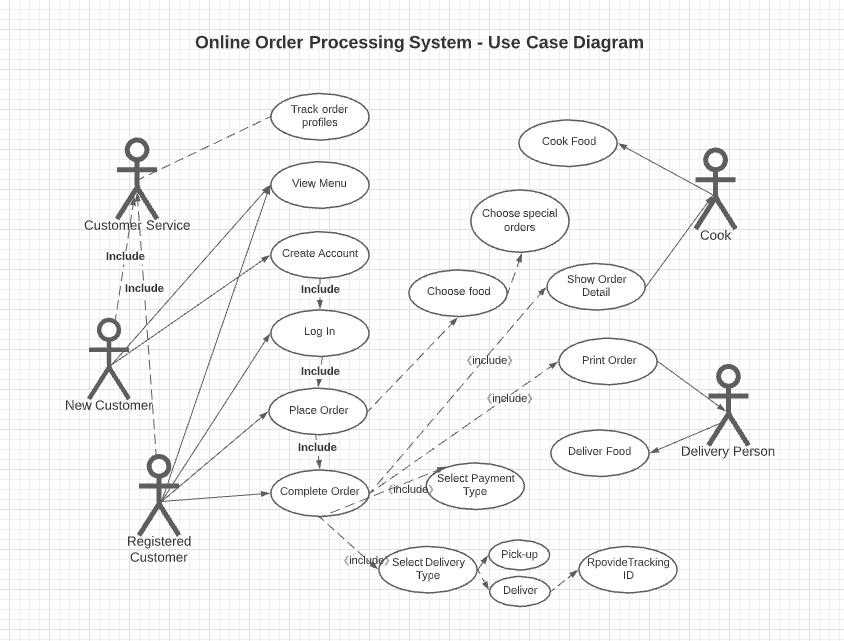


The architecture of **Orderonthego** is designed using a microservices-based approach that enables independent scalability and modular functionality. Users—including customers, restaurant owners, admins, and delivery partners—interact with the system via web and mobile interfaces, collectively referred to as the UI. This UI communicates directly with the **API Gateway**, which acts as the centralized entry point for all service requests. The API Gateway routes incoming requests to appropriate backend services and ensures secure communication between the frontend and backend systems.

To control access, the **Authentication & Authorization** module verifies user identity and determines permissions using secure tokens. This security layer prevents unauthorized access to services such as ordering, payments, and data management. Once

The **User Service** manages all user-related data and operations, including account creation, profile management, and role assignments. The **Payment Service** handles payment transactions, allowing users to pay for their orders using secure gateways. It supports multiple payment methods and handles refunds, invoices, and confirmations. The **Order Service** is central to processing and tracking food orders. It communicates with other services to update order status and route assignments in real time. Meanwhile, the **Menu Service** allows restaurant owners to manage their listings, update item availability, and set pricing dynamically.

# ER DIAGRAM



# PREREQUISITES

To build and deploy Orderonthego, a robust food ordering platform, the following technologies, tools, and skills are required:

### ****Development Environment****

* **Node.js & npm** – JavaScript runtime and package manager  
  [Node.js — Download Node.js®](https://nodejs.org/en/download/)
* **Visual Studio Code (VS Code)** or equivalent IDE
* [Visual Studio Code - Code Editing. Redefined](https://code.visualstudio.com/)
* **Git & GitHub** – Version control and collaboration  
  [Git - Downloads](https://git-scm.com/downloads)
* Sure! Here's a **50-line, easy-to-understand version** of the **Prerequisites** section, written in a clear and beginner-friendly way for your **Orderonthego: Food Ordering System** documentation:

## ****Prerequisites for Developing the Food Ordering System****

* To build the Orderonthego food ordering platform, we need certain tools, technologies, and some basic knowledge in web development. This system is created using the **MERN stack**, which includes MongoDB, Express.js, React.js, and Node.js.
* First, we need to install **Node.js**. It lets us run JavaScript on the server (backend). Along with it comes **npm**, a tool that helps install other required packages. You'll also need a code editor like **Visual Studio Code** to write and manage your code.
* To keep track of code changes and work with a team, we use **Git** and upload our projects to **GitHub**. This is important for version control and collaboration.
* For the frontend (what users see), we use **React.js**. It helps build fast and dynamic web pages. To make the website look nice, we can use design libraries like **Bootstrap**, **Material UI**, or **Ant Design**. To send and receive data from the backend, we use **Axios**.
* The backend (where the logic happens) is built using **Express.js**, a simple and powerful tool for creating APIs. These APIs help connect the frontend with the backend.
* We store all the user, food, order, and payment data in **MongoDB**, which is a NoSQL database. To make it easier to work with MongoDB, we use a tool called **Mongoose**. It helps us structure the data.

# ROLES & RESPONSIBILITIES

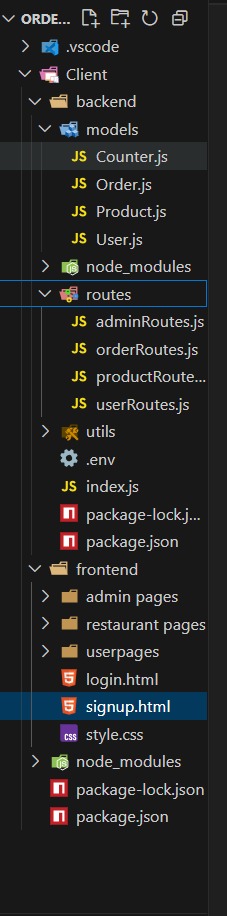
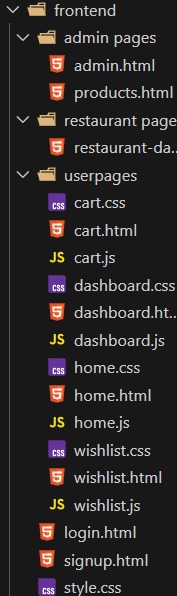
The Orderonthego food ordering platform includes four key roles: **Customer**, **Restaurant Owner**, **Delivery Partner**, and **Admin**, each playing a distinct part in the system’s functionality.**Customers** use the app to register, browse restaurants, place orders, make payments, and track their delivery. They can also rate their experience, manage order history, and cancel orders when necessary.

**Restaurant Owners** manage their menu, receive and process orders, update order status, and maintain item availability. They are responsible for ensuring quality service and responding to customer feedback.**Delivery Partners** are responsible for picking up food from restaurants and delivering it to customers. They update the delivery status in real time and ensure timely and accurate delivery.

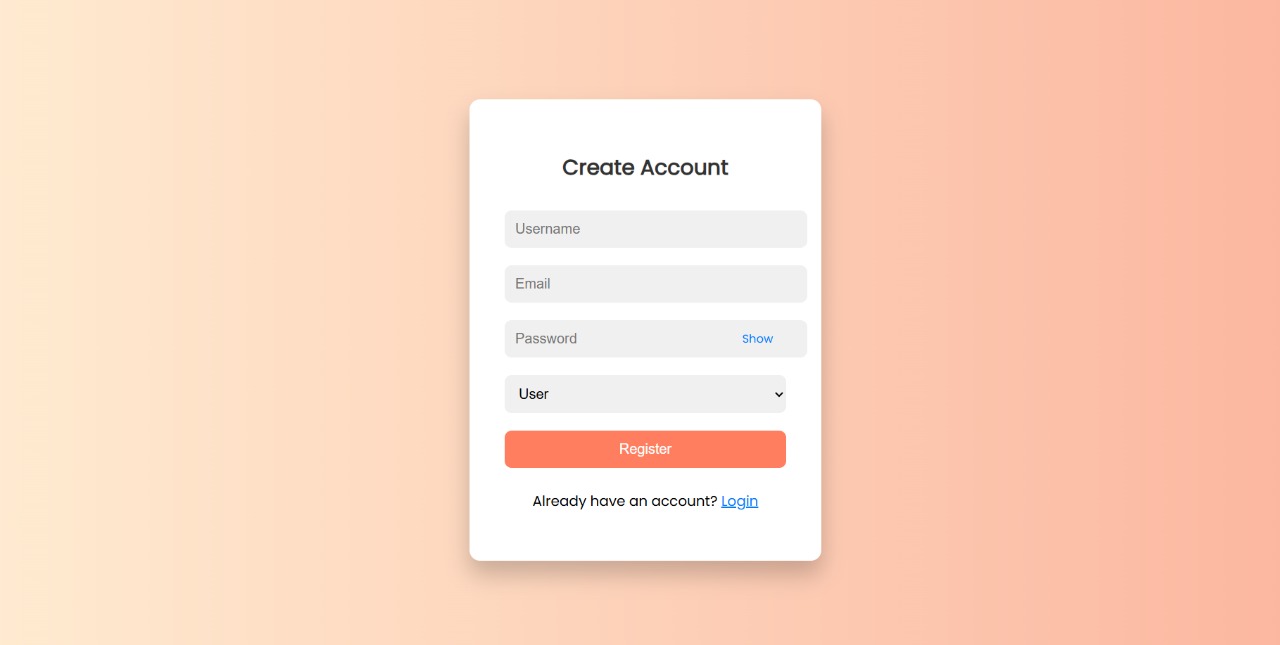
**Admins** oversee the entire platform, approve user registrations, manage restaurants and delivery agents, resolve disputes, and monitor system performance. They ensure security, compliance, and operational efficiency.

Each role contributes to the platform’s seamless workflow, enabling a complete and user-friendly food ordering experience from order placement to final delivery.

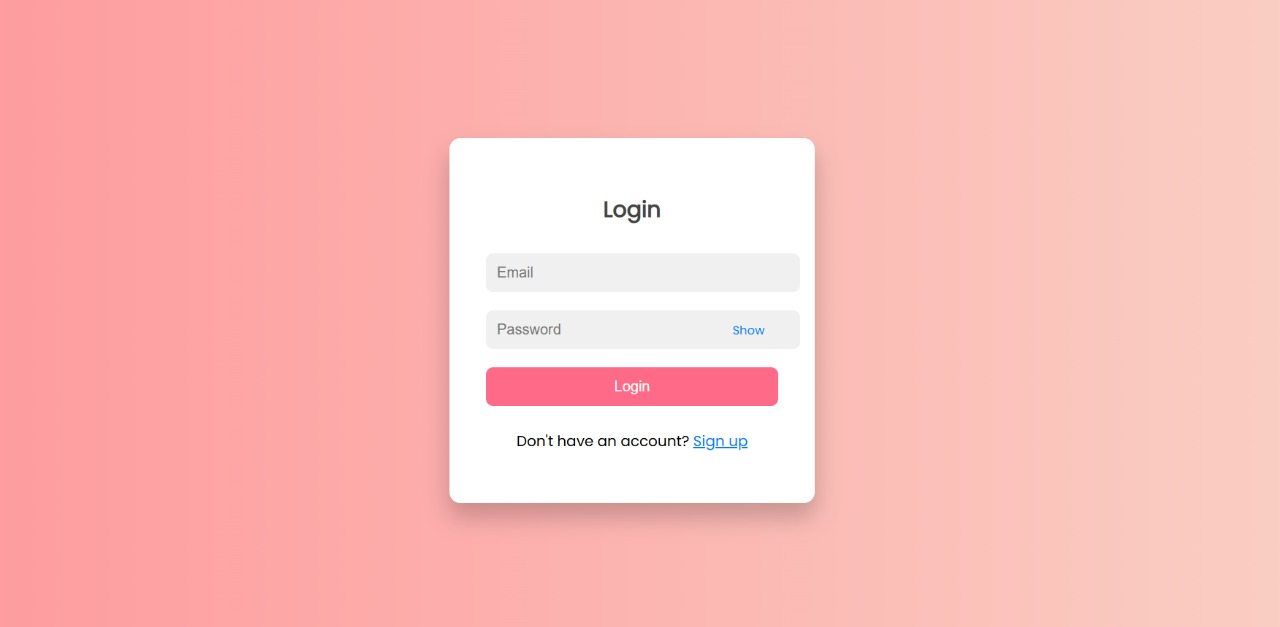
**PROJECT STRUCTURE:**



# Register or Signup:



# Login:



# REFERENCES

- https://www.swiggy.com/  
- https://www.zomato.com/  
- https://www.mongodb.com/docs/  
- https://www.section.io/engineering-education/building-a-mern-stack-app/