InfinoHop

🛍️ E-Commerce Platform – Developer Overview

Hey there, 👋 welcome to our **E-Commerce Platform** — a complete end-to-end solution that connects **Admins**, **Vendors**, and **Users** together under one powerful system! 🚀

This project isn't just an online store — it's a **multi-role management ecosystem** designed for real-world scalability, transparency, and automation. ⚙️

🧠 Project Summary

This project enables:

- **Users** 🧑 to browse and buy products.
- **Vendors** 🏪 to register, list products, track sales, and manage deliveries.
- **Admins** 👨‍💼 to control vendor access, manage subscriptions/commissions, monitor performance, and handle the business backend.

The project includes:
✅ Website (for users and vendors)
✅ Admin panel (for management)
✅ API backend (with JWT authentication 🔐)
✅ Real-time delivery status updates using WebSocket ⚡

🧩 Core Modules Overview

👨‍💼 1. Admin Panel

- Manages vendor approvals & subscriptions 💰
- Sets commission % or monthly fee during onboarding
- Handles vendor payments, user disputes & product monitoring
- Maintains a hierarchy of roles:
  - 👨‍💻 **Admin (Owner)** → full control

    - 👔 **Manager** → manages vendors & employees
    - 🧾 **Employee** → manages vendor requests only
- Can suspend vendors or discontinue access if payment/subscription expires

---

🏪 2. Vendor Management System

- Vendors register via “Sell Now” form ✍️
- Their request first goes to Admin for approval 🔎
- Admin sets commission/subscription fee and sends an approval email 📩
- Vendor accepts → officially added to vendors table ✅
- Vendors can:
    - Add/edit/delete products 🧾
    - Manage orders & update delivery status 📦
    - Track monthly sales, profit/loss, and payment reports 📊

🛒 3. User System

- Users can register/login and browse products 🧑🏿
- Add items to cart 🛍️
- Place orders and track delivery in real-time via WebSocket 🔔
- Can rate products, request returns, or cancel orders 🗳️
- Each order can have multiple products (handled via orderDetails table).

🧾 4. Product Management

- Each product belongs to a **Category → Subcategory → Vendor** hierarchy:
- Category → Subcategory → Product
- Admin controls all categories & subcategories.

- Vendors choose where their product belongs while adding listings.
- Products store:
  - Name, price, stock, image, and description
  - Linked vendor, subcategory, and optional category
  - Created date (createdAt TIMESTAMP)

💳 5. Order & Delivery System

- Each user can place multiple orders.
- Each order has multiple items stored in orderDetails table.
- Delivery tracking (delivery table) linked 1:1 with each order:
  - Delivery status, tracking number, shipped date, etc.
- Real-time delivery updates sent to users via WebSocket 🔄

💸 6. Payments & Subscriptions

- Vendors can operate under:
  - 🏢 **Monthly Subscription Model**
  - 💰 **Commission per Sale Model**
- Admin tracks payments using vendor_payments table.
- Non-paying vendors get temporarily deactivated 🚫.

🔒 Authentication – JWT Tokenization

- Every user (Admin, Vendor, or User) logs in and receives a **JWT token**.
- The token is required to access protected APIs.
- Middleware validates token before serving any request.
- Secure, stateless, and scalable 🔐.

🗂️ Database Overview (Simplified)

Table	Purpose
users	Stores user details
administration	Stores admin/manager/employee data with priority
vendors	Stores active vendors
vendor_requests	Temporary table for vendor onboarding requests
vendor_payments	Tracks subscription/commission payments
categories	Stores main product categories
subcategories	Linked to categories
products	Linked to subcategories & vendors
orders	Main order data per user
orderDetails	Products inside each order
delivery	Delivery info linked 1:1 to orders

⚙️ Tech Stack

Layer	Technology
Frontend	React.js / Next.js + Material UI 💅
Backend	Node.js / Express.js or Spring Boot ☕
Database	MySQL 🗄️
Auth	JWT (JSON Web Token) 🔐
Realtime	WebSocket / Socket.io ⚡

Layer	Technology
Cloud	AWS / Render / Railway (for deployment) ☁️

🧠 Future Enhancements

✨ Search Demand Counter:

Track user search frequency for products in specific areas and notify vendors of trending demands.

📱 Mobile App:

Build Android/iOS apps using React Native or Flutter.

🤖 AI Recommendation:

Suggest trending or high-demand products to vendors and users.

📊 Analytics Dashboard:

Show insights for admins and vendors using charts & graphs.

🌟 Developer Notes

“Every role in the system is connected but independent.”

- Admin handles vendors.
- Vendors handle products & orders.
- Users handle buying & reviewing.

This modular architecture keeps the system **secure, scalable, and easy to maintain** 🧩.

💬 Final Words

This e-commerce system is designed for **real-world scalability**, not just theory.
It focuses on automation, vendor profitability, and admin control while giving users a smooth shopping experience.

🚀 Smart. Secure. Scalable. Future-Ready.

That’s the motto of this project. 💪