

Amazon-like Inventory Management System - Full Stack Project Plan

1. Understanding the Project Scope

Building a platform like Amazon includes:

- Inventory and stock management
 - E-commerce features (search, cart, checkout)
 - Order and user account management
 - Admin dashboard with analytics
 - Stock manager panel for warehouse control
-

2. Tech Stack (Modern Full-Stack)

Frontend:

- **Framework:** React.js + Next.js
- **UI:** Tailwind CSS, ShadCN, Framer Motion
- **State Management:** Redux Toolkit / Zustand
- **Authentication:** NextAuth / Firebase Auth

Backend:

- **Framework:** Node.js with Express.js / NestJS
- **API:** REST (or GraphQL for advanced)
- **Auth:** JWT or OAuth2

Database:

- PostgreSQL (Primary relational DB)
- MongoDB (Optional for reviews, metadata)

DevOps & Hosting:

- Frontend: Vercel
 - Backend: Render / Railway
 - Database: Supabase / PlanetScale / Neon
 - Cloud Storage: AWS S3 / Cloudinary
-

3. Core Features

User Side:

- Product browsing & searching
- Product details page
- Add to cart / Wishlist
- Checkout & Order placement
- Track orders
- Ratings & Reviews
- Account management

Admin Panel:

- Manage products (Add/Edit/Delete)
- Category and inventory control
- Track and update order status
- View sales and user analytics

- Role-based access management

Stock Manager Panel:

- View and update product stock in real-time
- Alert system for low inventory
- Assign products to warehouses/locations
- Monitor incoming shipments and restock status
- Role-based restricted access

System Modules:

- Inventory Management
 - Order Management
 - User & Admin Management
 - Product Catalog
 - Notification System
 - Cart & Wishlist
 - (Optional) Payment Gateway (Stripe / Razorpay)
-

4. Workflow (Like Amazon)

User Flow:

1. Signup/Login
2. Browse/Search Products
3. Add to Cart/Wishlist
4. Checkout -> Payment -> Place Order

5. Inventory updates, order stored, user notified
6. Track delivery, review product

Admin Flow:

1. Admin login
2. Add/Edit Products
3. Update stock
4. Track and manage orders
5. View insights and analytics

Stock Manager Flow:

1. Login with stock manager role
2. View all inventory and low-stock alerts
3. Add or update stock quantities
4. Assign or transfer stock to specific warehouses
5. Mark items as restocked or damaged

5. Database Design (ERD Overview)

Key Tables:

- **Users:** id, name, email, password, role
- **Products:** id, name, category_id, price, stock, image_url
- **Categories:** id, name
- **Orders:** id, user_id, total, status, payment_status

- **OrderItems**: order_id, product_id, quantity, price
 - **CartItems**: user_id, product_id, quantity
 - **Wishlist**: user_id, product_id
 - **Reviews**: user_id, product_id, rating, comment
 - **Warehouses**: id, name, location
 - **InventoryLogs**: id, product_id, warehouse_id, quantity, action_type, updated_by, timestamp
-



6. Software Engineering Process

SDLC Steps:

1. Requirement Gathering (features, user stories)
 2. UI/UX Design (Figma)
 3. System Design (API, DB schema)
 4. Development (Agile sprints)
 5. Testing (Unit + Integration)
 6. Deployment (CI/CD)
 7. Monitoring & Maintenance
-



7. Project Management Tools

- Figma (Wireframes)
- Trello/Jira (Tasks, sprints)
- Notion (Docs, tracking)

- GitHub Projects (Kanban)



8. Milestone-Based Timeline

Phase	Task	Duration
1. Planning	Features, DB design, wireframes	1 week
2. Setup	Dev env, GitHub, structure	2 days
3. Frontend	Pages: Home, Product, Cart	1.5 weeks
4. Backend	Auth, Products, Orders APIs	2 weeks
5. Integration	FE + BE connect, fetch APIs	1 week
6. Admin Panel	Product/order mgmt	1 week
7. Stock Panel	Inventory features, restock alerts	1 week
8. Testing & Deployment	Hosting, testing	1 week



9. Extra Suggestions

- AI product recommendations (ML model)
- Multi-vendor marketplace support
- Admin analytics dashboard with Recharts or Power BI