

E-Commerce Marketplace Development

Technical Planning Documentation

Key Technologies

- Frontend: Next.js
- Content Management System (CMS): Sanity
- Order Tracking and Shipment: ShipEngine
- Database: MongoDB (for authentication)
- Hosting and Deployment: Vercel (for frontend) and AWS (for backend)
- Payment Gateway: Stripe

Technical Architecture

1. Frontend (Next.js)
2. Backend APIs
3. CMS (Sanity)
4. Order Tracking (ShipEngine)
5. Deployment: Frontend on Vercel,

API Endpoints

User Management

- POST /api/auth/register: Register a new user.
- POST /api/auth/login: User login.
- GET /api/users/profile: Fetch user profile (requires authentication).
- PUT /api/users/update: Update user details.

Product Management

- GET /api/products: List all products.
- GET /api/products/:id: Fetch product details by ID.
- POST /api/products: Add a new product (requires seller role).

Products Schema

- product_id: Unique identifier for the product.
- name: Name of the product.
- price: Rental cost per day/hour.
- stock: Availability status of the product.
- description: Detailed description of the product.
- image_url: URL of the product image.
- sizes (optional): Available sizes for the product.
- user_id (mandatory): ID of the seller who listed the product.

Orders Schema

- `order_id`: Unique identifier for the order.
- `customer_id`: Reference to the customer placing the order.
- `product_id`: Reference to the rented product.
- `quantity`: Number of products rented.
- `status`: Current status (e.g., Pending, Confirmed, Completed).
- `order_date`: Timestamp of when the order was placed.

Integration Details

Sanity CMS: Manages dynamic content.

ShipEngine: API for shipping labels and tracking.

Stripe Integration: Secure payment processing.

Deployment Plan

Frontend (Next.js)

- Hosting: Vercel.
- CI/CD: Automatic deployment from GitHub.

Backend

- Sanity handling the backend