System Architecture

Overview of System Components

Frontend (Next.js)

- **Role:**
    - Serves as the interactive interface for users to browse products, place orders, and track shipments.
- **Responsibilities:**
    - Implements responsive design for multiple devices.
    - Manages dynamic content rendering based on user activity.
    - Connects with backend APIs to fetch product details, order data, and shipment status.

Sanity CMS (Backend)

- **Role:**
    - Acts as the central database for all marketplace data (products, customers, orders).
    - Provides an API layer to interact with the data dynamically.
- **Responsibilities:**
    - Maintain product catalogs, user profiles, and order history.
    - Provide secure APIs for frontend communication.
    - Enable future scalability with modular schema design.

Third-Party APIs

Shipment Tracking API

- **Role:**
  - Tracks and updates shipment status (e.g., "In Transit", "Delivered").
- **Responsibilities:**
  - Provide accurate delivery information to customers.
  - Integrate seamlessly with backend for real-time status updates.

Payment Gateway

- **Role:**
  - Processes and validates customer payments securely.
- **Responsibilities:**
  - Support multiple payment methods like credit cards, debit cards, and COD.
  - Provide confirmation updates to customers and backend systems.

3. Key Workflows and Data Flow

1. User Registration

- Users submit a registration form.
- Data is securely sent to Sanity CMS for storage.
- Confirmation is displayed on the frontend.

2. Product Browsing

- Users browse the product listing page.
- The frontend fetches product data from the Sanity CMS Product Data API.

- Filters and search refine the product results dynamically.

3. Order Placement

- Users add items to the cart and proceed to checkout.
- Order details are sent to Sanity CMS.
- Payments are processed via the Payment Gateway.
- Successful orders are stored in Sanity CMS and confirmed to users.

4. Shipment Tracking

- Users can track their order in real time.
- Backend communicates with the Shipment Tracking API.
- Tracking status is displayed on the frontend.

4. System Design Features

Scalability

- Modular schemas allow the addition of new product categories.
- Next.js enables efficient handling of a growing user base.

Security

- Secure payment integration via third-party gateways.
- API endpoints are protected to prevent unauthorized access.

Performance

- Optimized server-side rendering for faster page loads.
- Efficient APIs ensure quick data fetching.

5. Detailed Workflow and Interactions

Frontend ↔ Backend

- Users interact with the website, triggering API requests to Sanity CMS.
- Sanity CMS responds with data like product details, order confirmations, and user profiles.

Backend ↔ Third-Party APIs

- Backend interacts with Shipment and Payment APIs to process orders and track shipments.
- Responses from APIs are updated in Sanity CMS and displayed to users.