Marketplace Technical Foundation

1. System Architecture Overview

The system consists of the following key components:

- **Frontend (Next.js):** Handles user interactions and displays marketplace data.
- **Sanity CMS (Backend):** Manages products, orders, and customer data.
- **Third-Party APIs:**
  - Payment Gateway: Processes payments securely.
  - Shipment API: Fetches real-time order delivery status.

Workflow:

1. Users browse products on the frontend.
2. The frontend requests product data from Sanity CMS.
3. Orders are placed and stored in Sanity CMS.
4. Payment is processed via the Payment Gateway.
5. Shipment tracking updates are fetched from the Shipment API.

2. Key Workflows

1. **Product Browsing:**
   - User selects a category -> Fetch data from Sanity -> Display products.

2. **Order Placement:**
   - User adds products to cart -> Checkout -> Order is saved in Sanity CMS.

3. **Payment Processing:**
   - Order details sent to Payment Gateway -> Payment confirmed -> Order finalized.

4. **Shipment Tracking:**
   - Order status retrieved from Shipment API -> Displayed on frontend.

3. Category-Specific Instructions

- **Q-Commerce:** Focus on express delivery workflows. Example API: /express-delivery-status.

- **Rental eCommerce:** Includes rental duration, deposit, and return condition tracking.
- **General eCommerce:** Standard product browsing, cart management, and order placement.

4. API Endpoints

Endpoint	Method	Purpose	Response Example
/products	GET	Fetch all product details	{ "id": 1, "name": "Product A", "price": 100 }
/orders	POST	Create a new order	{ "orderId": 123, "status": "Pending" }
/shipment	GET	Track shipment status	{ "orderId": 123, "status": "Shipped" }

5. Sanity Schema Example

```
export default {
  name: 'product',
  type: 'document',
  fields: [
    { name: 'name', type: 'string', title: 'Product Name' },
    { name: 'price', type: 'number', title: 'Price' },
    { name: 'stock', type: 'number', title: 'Stock Level' }
  ]
};
```