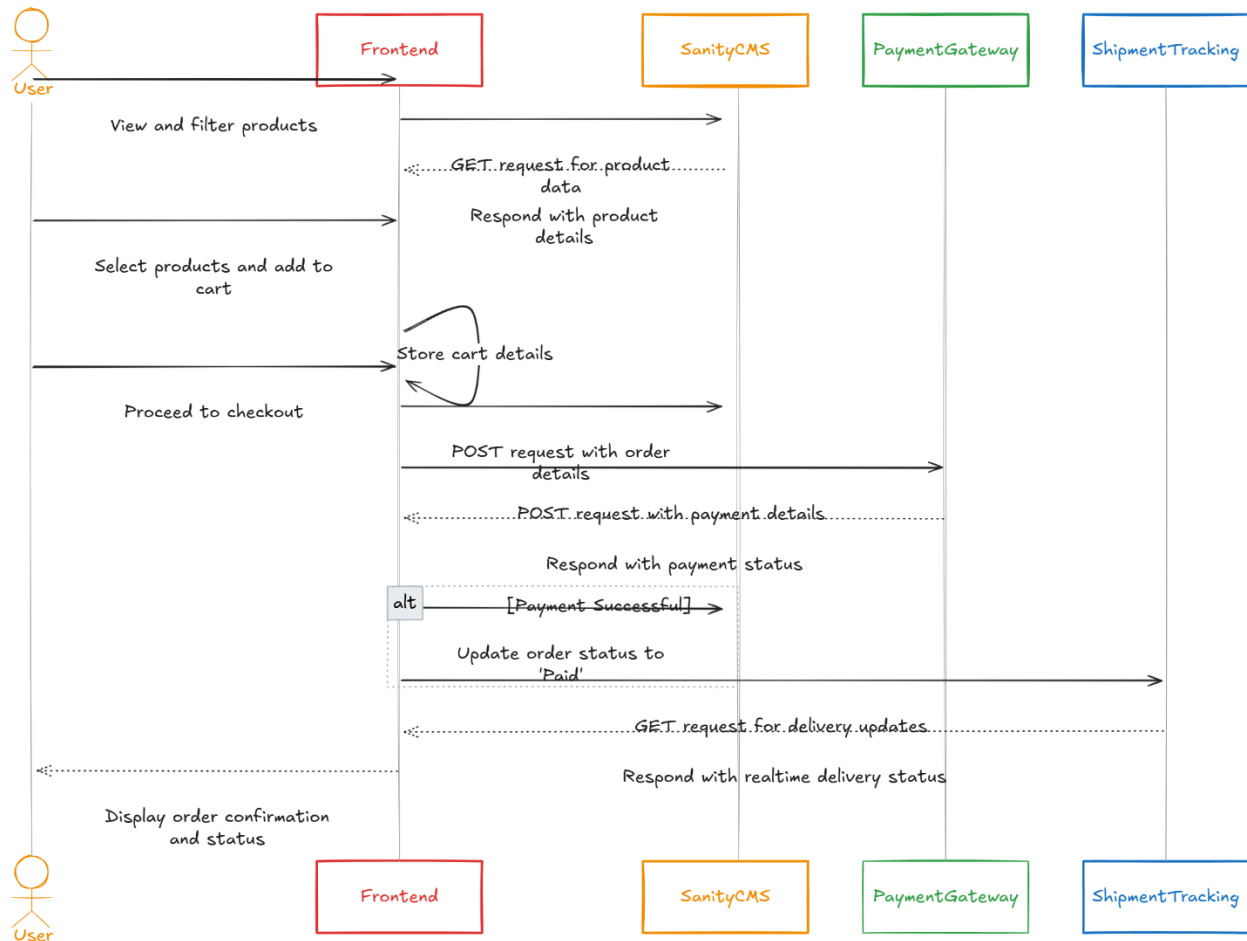


# Marketplace Technical Foundation - [Hybrid Q-commerce For Rental]

## 1. System Architecture Overview

### Architecture Diagram



### Component Roles

- **Frontend (Next.js):**
  - Provides an intuitive and responsive user interface for browsing, purchasing, and renting products.
  - Sends API requests to backend services for data retrieval and order processing.
- **Sanity CMS:**
  - Manages product inventory, customer details, and order records.

- Acts as a centralized database, providing APIs to the frontend.
  - **Third-Party APIs:**
    - **Shipment Tracking API:** Provides real-time delivery updates.
    - **Payment Gateway API:** Handles secure payment processing for purchases and rentals.
- 

## 2. Key Workflows

### 1. User Registration

- **Steps:**
  1. User registers via the frontend.
  2. Data is sent to Sanity CMS via API and stored.
  3. A confirmation email or message is sent to the user.

### 2. Product Browsing

- **Steps:**
  1. User visits the product listing page.
  2. Frontend sends a GET request to the `/products` endpoint of the Product Data API (Sanity CMS).
  3. Products are dynamically displayed with filtering options (e.g., "Buy" or "Rent").

### 3. Order Placement

- **Steps:**
  1. User adds items to the cart and proceeds to checkout.
  2. Frontend sends a POST request to the `/orders` endpoint in Sanity CMS.
  3. Sanity CMS records the order with status `Pending`.

### 4. Shipment Tracking

- **Steps:**
  1. Frontend sends a GET request to the `/shipment` endpoint of the Shipment Tracking API with the order ID.
  2. API responds with real-time shipment status (e.g., "In Transit").
  3. Status is displayed on the order confirmation page.

### 5. Payment Processing

- **Steps:**
  1. Frontend sends a POST request to the `/process-payment` endpoint of the Payment Gateway API with payment details.

2. Payment is processed securely, and the API returns a success or failure response.
3. Sanity CMS updates the order status to `Paid` upon success.

---

## 3. Category-Specific Instructions

### Q-Commerce:

- Focus on real-time inventory updates and delivery SLA tracking.
- **Example Endpoint:** `/express-delivery-status`
  - **Method:** GET
  - **Description:** Fetch real-time tracking for express deliveries.
  - **Response Example:**

```
{ "orderId": 123, "status": "In Transit", "ETA": "15 mins" }
```

### Rental eCommerce:

- Workflows for rental-specific features:
  - Rental duration management.
  - Deposit tracking.
  - Condition reports for returned items.
- **Schema Example Fields:** `rentalDuration`, `depositAmount`, `conditionStatus`.

### General eCommerce:

- Standard workflows like product browsing, cart management, and order placement.
- **Example Endpoint:** `/products`
  - **Method:** GET
  - **Description:** Fetch all product details.
  - **Response Example:**

```
{ "id": 1, "name": "Product A", "price": 100 }
```

---

## 4. API Endpoints

Endpoint	Method	Purpose	Response Example
<code>/products</code>	GET	Fetch all product details	<pre>[ { "id": 1, "name": "Product A", "price": 100 } ]</pre>
<code>/products/:id</code>	GET	Fetch specific product details	<pre>{ "id": 1, "name": "Product A", "price": 100, "stock": 20 }</pre>
<code>/orders</code>	POST	Create a new order	<pre>{ "orderId": 456, "status": "Order Placed" }</pre>
<code>/rental-duration</code>	POST	Add rental details	<pre>{ "confirmationId": 789, "status": "Success" }</pre>

/shipment	GET	Track order shipment status	{ "orderId": 123, "status": "In Transit", "ETA": "15 mins" }
/process-payment	POST	Process payment securely	{ "transactionId": "txn_001", "status": "Success" }

## 5. Sanity Schema Example

### 1. Product Schema Example

```
{
  "name": "Party Balloons",
  "description": "High-quality balloons suitable for birthday parties.",
  "price": 20,
  "rentalPrice": 5,
  "stock": 100,
  "category": "Party Decoration",
  "rentalDuration": 7,
  "deposit": 50
}
```

### 2. Order Schema Example

```
{
  "orderId": 123,
  "customer": {
    "name": "Alice Smith",
    "email": "alice@example.com",
    "phone": "123-456-7890"
  },
  "products": [
    {
      "productId": 1,
      "name": "Party Balloons",
      "type": "rental",
      "quantity": 10,
      "duration": "7 days",
      "deposit": 50
    },
    {
      "productId": 2,
      "name": "Catering Table",
      "type": "purchase",
      "quantity": 2
    }
  ],
  "totalPrice": 120,
  "paymentStatus": "Paid",
  "orderStatus": "In Progress"
}
```

### 3. Customer Schema Example

```
{
  "customerId": 456,
  "name": "Bob Johnson",
  "email": "bob@example.com",
  "phone": "987-654-3210",
  "address": {
    "street": "123 Main St",
    "city": "Los Angeles",
    "state": "CA",
    "zip": "90001"
  },
  "orders": [
    { "orderId": 123, "orderDate": "2025-01-15", "totalPrice": 120 },
    { "orderId": 124, "orderDate": "2025-01-16", "totalPrice": 50 }
  ]
}
```

---

### 4. Rental Schema Example

```
{
  "rentalId": 789,
  "productId": 1,
  "customerId": 456,
  "rentalDuration": 7,
  "deposit": 50,
  "conditionStatus": "Good",
  "returnDate": "2025-01-22",
  "rentalPrice": 5
}
```

---