"Marketplace Technical Foundation - [ee-commerce-furniture-website-furino]"
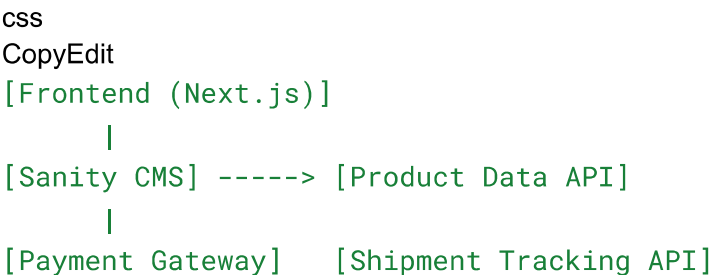
---

1. System Architecture Overview:

- **Frontend (Next.js):** Handles the user interface and interacts with Sanity CMS and third-party APIs.
- **Sanity CMS:** Manages product data, customer details, and order records.
- **Third-Party APIs:**
  - **Payment Gateway:** Processes payments securely.
  - **Shipment Tracking API:** Fetches real-time order tracking data.

Architecture Diagram:

css
CopyEdit
```
[Frontend (Next.js)]
      |
[Sanity CMS] -----> [Product Data API]
      |
[Payment Gateway]   [Shipment Tracking API]
```

---

2. Key Workflows:

1. User Registration:

- User signs up on the frontend.
- Data is stored in Sanity CMS.
- Confirmation sent to the user.

2. Product Browsing:

- User browses product categories on the frontend.
- Frontend makes a request to the Sanity API to fetch product data.
- Products are displayed dynamically.

3. Order Placement:

- User adds items to cart and proceeds to checkout.
- Order details are sent to Sanity CMS and recorded.
- Payment is processed via the Payment Gateway.
- Confirmation sent to the user.

4. Shipment Tracking:

- User tracks the status of their order via a third-party Shipment Tracking API.
- Shipment status is displayed to the user.

---

3. 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 in Sanity	{ "customerInfo": "John", "items": [...], "status": "pending" }
/shipment	GET	Track order status via third-party API	{ "shipmentId": 123, "status": "In Transit", "ETA": "15 mins" }

---

4. Sanity Schema Example:

Product Schema:
javascript
CopyEdit

```javascript
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' },
    { name: 'image', type: 'image', title: 'Product Image' }
  ]
};
```