

# Marketplace Technical Foundation

---

***Table of Contents:***

*Technical Foundation Planning*..... 2

Technical Requirements..... 2

System Architecture..... 3

Detailed Workflows..... 3,4,5

API Requirements..... 5,6,7,8

Sanity Schema Design..... 8,9,10,11

# Technical Foundation Planning

---

## *Technical Requirements:*

### Frontend Requirements:

- User Friendly Interface using [Next.js](#) powered by **Sanity CMS** for browsing products dynamically.
- Responsive design for mobile and desktop users by [Next.js](#) and **TailwindCSS**.
- Essential pages are: Home, Product Listing, Product Detail, Cart, Checkout, order confirmation and Signup / Login.

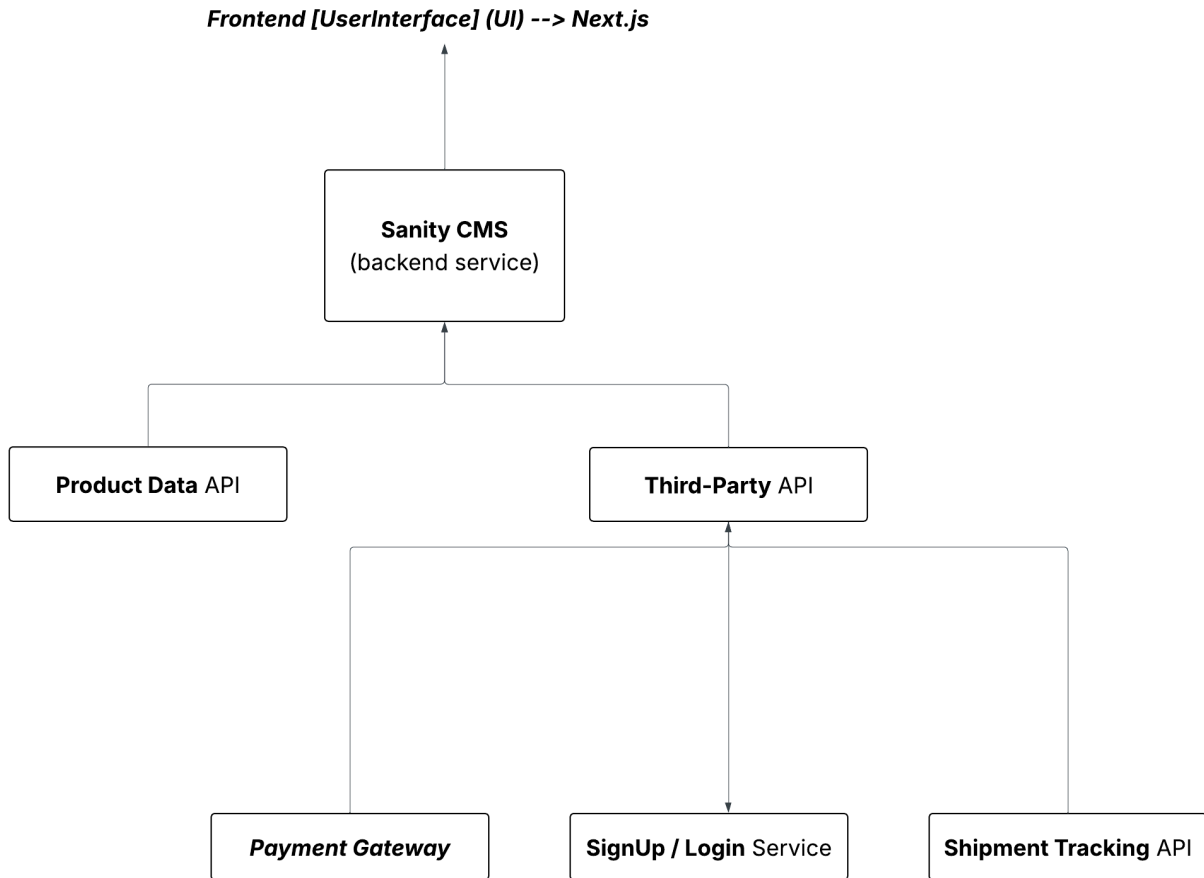
### Backend Requirements:

- Using **Sanity CMS** for managing product data, User Details, Customer Details, Order Records, Shipment Details. **Sanity** will act as the database of the marketplace.
- Designing **Sanity Schemas** aligned with Business goals and requirements.

### Third-Party APIs:

- Integrating **APIs** for Shipment Tracking, Payment Gateways and Login / Signup features.
- Ensuring **APIs** provide the necessary data for frontend functionality

# SYSTEM ARCHITECTURE:



A detailed **System Architecture** showing the overall design and the interaction between different components of a system.

## Detailed WorkFlows:

### User Registration:

- **SignUp:**

- User **SignUp** → Data is stored in **Sanity** → Confirmation sent to the user.
- **Login:**
- User **login** → Data is checked by **Sanity** → Confirmation sent to the user.

## User Browsing Products / Products Details:

- **Browsing Products:**
- User views **Product Categories** → **Sanity API** fetches data → **Products** displayed on frontend.
- **Browsing Products Details:**
- User clicks on **Product** → **Sanity API** fetches product data → **Products details** displayed on frontend.

## User Place Order:

- **Login / SignUp Check:**
- User clicks on **Add to cart** → **Sanity API** checks user data → **User Data** not found, please **Login**.
- **Placing Order:**
- User adds items to the **cart** → Proceeds to **checkout** → Order details saved in **Sanity**.

## User Proceeds to Transaction:

- **Payment Gateway:**
- User clicks on **checkout** → **Transaction** details fetched and displayed to user → User pays the **Payment**.

## User Tracking Shipment :

- **Shipment Tracking:**
- Order status updates fetched via **Third-Party API** -> Displayed to the user in real time.

## API Requirements:

- **For Fetching Products:**
- **Endpoint Name:** /products
- **Method:** GET
- **Description:** Fetch all available products from Sanity.
- **Response:** Product details (ID, name, price, stock, image, rates, discount).
- **For Fetching Filters:**
- **Endpoint Name:** /filters

- **Method:** GET
- **Description:** Fetch filters of a product.
- **Response:** Product ID, colors, sizes, dress style, dress type.

### - For Sending Orders Info:

- **Endpoint Name:** /orders.
- **Method:** POST
- **Description:** Create a new order in Sanity.
- **Payload:** Order ID, customer info, product details, payment status, time stamp.

### - For Sending Orders Summary:

- **Endpoint Name:** /order-summary
- **Method:** POST
- **Description:** Send order summary to sanity.
- **Response:** Order ID, sub total, discount, delivery, total.

### - For Sending Customers Info:

- **Endpoint Name:** /customer-info

- **Method:** POST
- **Description:** sending customer details to sanity.
- **Response:** Customer ID, name, contact info, address.

#### - **For Shipment:**

- **Endpoint Name:** /express-delivery-status
- **Method:** GET
- **Description:** Fetch real-time delivery updates for perishable items.
- **Response:** Shipment ID, order ID, status, ETA.

#### - **For Fetching Delivery Zones:**

- **Endpoint Name:** /delivery-zones
- **Method:** GET
- **Description:** Fetch delivery zones information.
- **Response:** Zone name, coverage area, assigned driver.

#### - **For Sending Payment Details:**

- **Endpoint Name:** /payment-details
- **Method:** POST

- **Description:** Sending payment details to sanity.
- **Response:** Order ID, customer ID, amount, payment gateway, status.

### - For Sending Customer Reviews:

- **Endpoint Name:** /customer-reviews
- **Method:** POST
- **Description:** Sending customer reviews to sanity.
- **Response:** Customer ID, order ID, customer name, rates, comments.

### - For Fetching Customer Reviews:

- **Endpoint Name:** /customer-reviews
- **Method:** GET
- **Description:** Fetching customer reviews from sanity.
- **Response:** Customer ID, order ID, customer name, rates, comments.

## Sanity Schema Design:

### For Product:

```
export default {  
  name: 'product',  
  type: 'document',
```



```
fields: [  
  { name: 'name', type: 'string', title: 'Product Name' },  
  { name: 'price', type: 'number', title: 'Price' },  
  { name: 'image', type: 'string', title: 'Product Image' },  
  { name: 'discount', type: 'number', title: 'Discount' },  
  { name: 'rates', type: 'string', title: 'Rates' },  
  { name: 'rates image', type: 'string', title: 'Rates Image' },  
  { name: 'stock', type: 'number', title: 'Stock Level' }  
]  
};
```

## For Orders:

```
export default {  
  name: 'Order',  
  type: 'document',  
  fields: [  
    { name: 'id', type: 'number', title: 'Order ID' },  
    { name: 'customer id', type: 'number', title: 'Customer ID' },  
    { name: 'customer name', type: 'string', title: 'Customer Name'},  
    { name: 'product names', type: 'string', title: 'Product Names' },  
    { name: 'order status', type: 'string', title: 'Order Status' },  
    { name: 'time stamp', type: 'number', title: 'Time Stamp' }  
  ]  
};
```

## For Orders Summary:

```
export default {  
  name: 'Order Summary',  
  type: 'document',  
  fields: [  
    { name: 'id', type: 'number', title: 'Order ID' },  
    { name: 'subtotal', type: 'number', title: 'Subtotal' },  
    { name: 'discount', type: 'number', title: 'Discount'},  
    { name: 'delivery', type: 'number', title: 'Delivery charges' },  
    { name: 'total', type: 'number', title: 'Total' },  
  ]  
};
```

```
]  
};
```

## For Customer Details:

```
export default {  
  name: 'Customer',  
  type: 'document',  
  fields: [  
    { name: 'id', type: 'number', title: 'Customer ID' },  
    { name: 'name', type: 'string', title: 'Customer Name' },  
    { name: 'contact', type: 'string', title: 'Contact No'},  
    { name: 'email', type: 'string', title: 'Customer Email' },  
    { name: 'address', type: 'string', title: 'Address' },  
  ]  
};
```

## For Shipment Details:

```
export default {  
  name: 'Shipment',  
  type: 'document',  
  fields: [  
    { name: 'id', type: 'number', title: 'Customer ID' },  
    { name: 'order id', type: 'number', title: 'Order ID' },  
    { name: 'status', type: 'string', title: 'Status'},  
    { name: 'delivery date', type: 'string', title: 'Delivery Date' },  
  ]  
};
```

## For Delivery Zones:

```
export default {  
  name: 'Delivery Zone',  
  type: 'document',  
  fields: [  
    { name: 'zone name', type: 'string', title: 'Zone Name' },  
  ]  
};
```

```
{ name: 'coverage area', type: 'string', title: 'Coverage Area'},  
{ name: 'driver', type: 'string', title: 'Assigned Driver' },  
]  
};
```

## For Payments:

```
export default {  
  name: 'Payment',  
  type: 'document',  
  fields: [  
    { name: 'order id', type: 'number', title: 'Order ID' },  
    { name: 'customer id', type: 'number', title: 'Customer ID' },  
    { name: 'amount', type: 'number', title: 'Amount'},  
    { name: 'payment gateway', type: 'string', title: 'Payment Gateway' },  
    { name: 'status', type: 'string', title: 'Status' },  
  ]  
};
```

## For Payments:

```
export default {  
  name: 'Payment',  
  type: 'document',  
  fields: [  
    { name: 'order id', type: 'number', title: 'Order ID' },  
    { name: 'customer id', type: 'number', title: 'Customer ID' },  
    { name: 'customer name', type: 'string', title: 'Customer Name'},  
    { name: 'rates', type: 'string', title: 'Rates' },  
    { name: 'rates image', type: 'string', title: 'Rates Image' },  
    { name: 'comments', type: 'string', title: 'Comments' },  
  ]  
};
```

***‘A DETAILED DOCUMENT OF TECHNICAL FOUNDATION’***