

Technical Foundation Planning

➤ **System Architecture:**

- Use Next.js to improve performance and SEO optimization.
- Integrate Sanity CMS for admin profile management, including adding, deleting, and editing products.
- Implement a payment gateway API for secure transactions.
- Set up user authentication to allow product orders only when users are logged in.

➤ **Frontend Requirements:**

● **User Interface:**

- Design an intuitive and user-friendly interface for easy browsing of products and incorporate smooth navigation and interactive elements for a seamless user experience.

● **Responsive Design:**

- Ensure the marketplace is fully functional across both mobile and desktop platforms.

● **Essential Pages :**

- **Home Page:** Showcase featured products or categories.
- **About Page:** Provide information about the Business.
- **Listing Page:** Display available products.
- **Product Details Page:** Present product images, descriptions, prices, and an "Add to Cart" button.
- **Sign-Up Page:** Allow users to sign up and log in. Only logged-in users can order products or add items to the cart.
- **Cart Page:** Display selected products with options to adjust quantities or remove items for logged-in users.
- **Checkout Page:** Handle customer payment and shipping details.

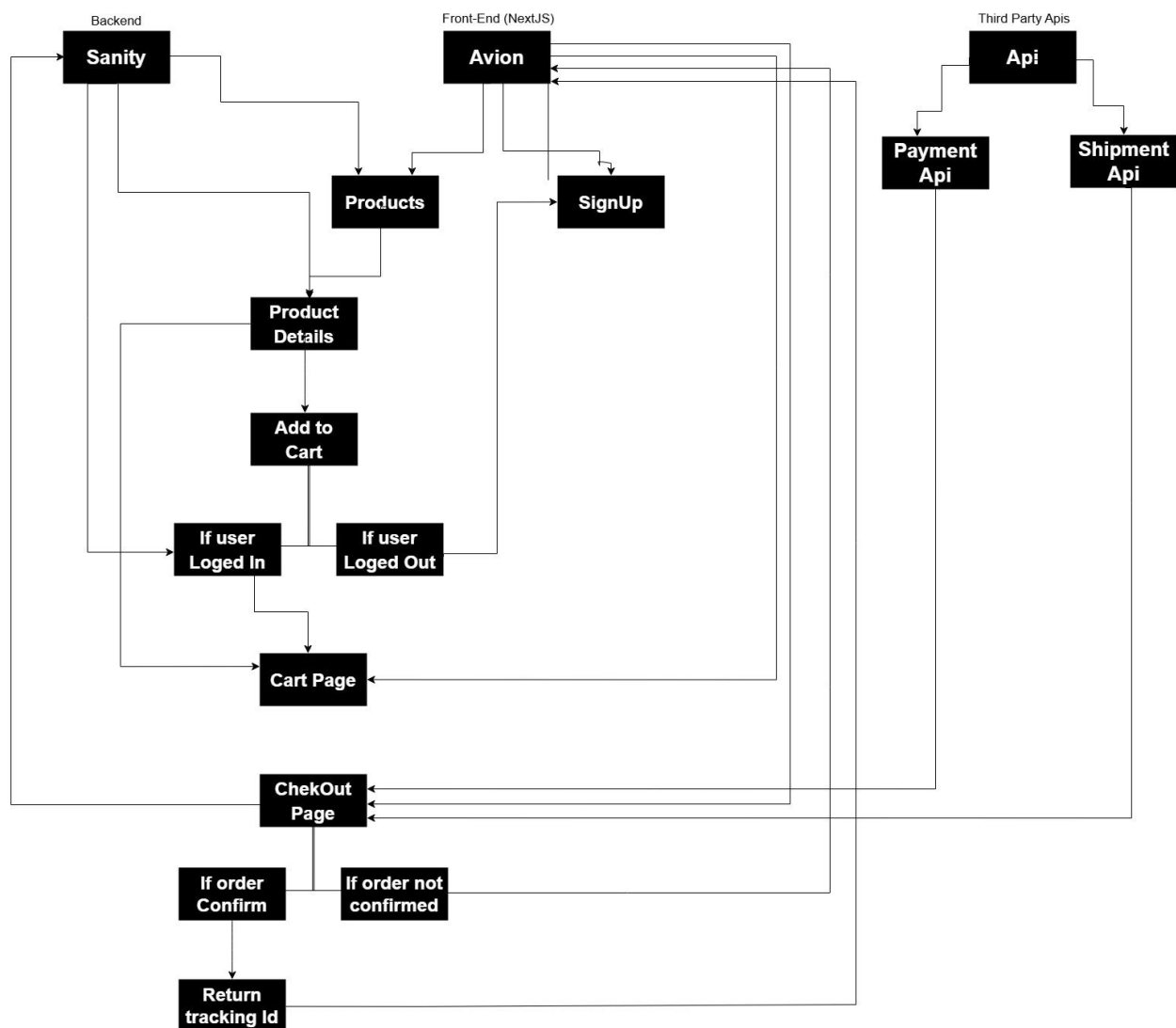
➤ **Backend Requirements :**

● Use Sanity to manage:

- **Product Data:** Including name, description, price, category, and images.
- **Customer Information:** Manage profiles (name,contact,address, email etc) and orders.
- **Order Records:** Track order status, payment confirmations, and shipping details.

- Schema Design:
 - Set up schemas in Sanity to organize data based on the business needs, such as products, orders, customers, and delivery zones.
- **Implement Third-Party APIs:**
 - **Shipment Tracking API:** Enable customers to easily track the status and progress of their orders.
 - **Payment Gateway API** (e.g., Stripe, Easypaisa, JazzCash): Provide secure, fast, and reliable payment processing for customer transactions.

System Architecture Diagram



Work Flows

i. User Registration Process:

- **Flow:**
 - User registers → Form information is sent to Sanity CMS → User account is created → A confirmation message or email is sent to the user.
- **Purpose:**
 - Facilitates smooth user registration and account management.

ii. Product Browsing Process:

- **Flow:**
 - User explores the marketplace → The frontend makes an API call to Sanity CMS → Product information (name, description, price, stock, images) is retrieved → The data is displayed dynamically on the website.
- **Purpose:**
 - Allows users to view product details and make informed purchasing decisions.

iii. Order Placement Process:

- **Flows:**
 - User adds items to the cart on the frontend.
 - At checkout, order details (product info, total cost, and user info) are sent to Sanity CMS through an API call.
 - The order is logged in Sanity CMS with a unique order identifier.
 - The user is directed to the payment gateway to process the payment.
 - Once payment is successful, a confirmation is sent to the user and the order status is updated in Sanity CMS.
- **Purpose:**
 - Ensures smooth order recording and secure payment processing.

iv. Shipment Tracking Process:

- **Flows:**
 - The user checks the status of their order on the frontend.
 - The frontend fetches real-time tracking info from a Third-Party Shipment API.

- Shipment details (current location, expected delivery time) are displayed to the user.
- Updates are stored in Sanity CMS for record-keeping.
- **Purpose:**
 - Keeps users updated on the status of their shipments.

API Endpoints

1. Get All Products

- **Endpoint:** /products
- **Method:** GET
- **Description:** Retrieve all available products from Sanity CMS, including perishable food items.
- **Sample Response:**

```
{
  "id": 1,
  "name": "Dandy Chair",
  "price": 50,
  "stock": 200,
  "category": "Chairs",
  "image": "dandy_chair.jpg"
}
```

2. Create a New Order

- **Endpoint:** /orders
- **Method:** POST
- **Description:** Submit a new food order to Sanity CMS.
- **Sample Payload:**

```
{
  "customerId": "00232232",
  "customerName": "Taha",
  "address": "Karachi, Pakistan",
  "products": [
    {
      "productId": 1,
      "quantity": 2,
      "price": 50
    }
  ]
}
```

```
    },  
    {  
      "productId": 2,  
      "quantity": 1,  
      "price": 35 }  
  ],  
  "totalAmount": 85,  
  "paymentStatus": "Pending"  
}
```

Conclusion:

- This platform is designed to offer a scalable and efficient solution for Furnisher sales.
- The frontend is developed using **Next.js**.
- **Sanity CMS** is used as the backend to manage and organize the website content.
- **Shipping Api** is integrated for order tracking and real-time delivery updates.