

System Architecture

Role of Each Component:

- **Frontend (Next.js):** The user-facing interface where customers browse products, add items to their cart, checkout, and view their order status.
 - **Sanity CMS:** The content management system where all the product data (name, price, stock, etc.) and user data (orders, customer info) are stored.
 - **Product Data API:** A set of RESTful API endpoints to fetch product listings, product details, and manage the shopping cart.
 - **Third-Party API:** Used for fetching external data such as shipment tracking and delivery status.
 - **Payment Gateway:** Securely handles the payment process, including transactions, refunds, and payment confirmations.
-

Key Workflows

1. User Registration:

- **Action:** A new user signs up using an email and password.
- **Backend Process:** The user data is stored in the Sanity CMS under a "Users" schema.
- **Response:** A confirmation email is sent to the user, and the user is logged into the system.

2. Product Browsing:

- **Action:** User browses different product categories like electronics, clothing, etc.
- **Backend Process:** Product data is fetched via the **Product Data API**, connecting to the Sanity CMS.
- **Response:** A list of products with details such as name, price, image, and availability is shown to the user.

3. Order Placement:

- **Action:** User selects products, adds them to the cart, and checks out.
- **Backend Process:** The cart and order details are saved in Sanity CMS. The payment gateway is called to process the payment.
- **Response:** After payment success, the order is confirmed, and the user receives an order confirmation email.

4. Shipment Tracking:

- **Action:** User wants to track the shipment status of their order.
- **Backend Process:** The **Third-Party API** is called to fetch the real-time shipment status for the order.
- **Response:** Shipment status (e.g., "Shipped", "In Transit", "Delivered") is displayed to the user with an estimated delivery date.

API Endpoints

Endpoint	Method	Description	Response Example
/products	GET	Fetch all product details from Sanity	{ "id": 1, "name": "Smartphone", "price": 500, "stock": 20, "image": "url" }
/orders	POST	Create a new order in Sanity	{ "orderId": 123, "status": "Success" }
/shipment	GET	Track order status via shipment API	{ "shipmentId": 789, "status": "In Transit", "ETA": "5 mins" }
/checkout	POST	Process payment and finalize order	{ "paymentStatus": "Success", "orderId": 123, "totalAmount": 1000 }

Sanity Schema Example

Here's a **Product Schema** example for Sanity CMS:

```
1  export default {  
2    name: 'product',  
3    type: 'document',  
4    fields: [  
5      { name: 'name', type: 'string', title: 'Product Name' },  
6      { name: 'price', type: 'number', title: 'Price' },  
7      { name: 'stock', type: 'number', title: 'Stock Level' },  
8      { name: 'image', type: 'image', title: 'Product Image' },  
9      { name: 'category', type: 'string', title: 'Product Category' },  
10     { name: 'description', type: 'text', title: 'Product Description' }  
11   ]  
12 };  
13
```

This schema will store the essential data for products in the marketplace, such as product name, price, stock, category, and description.

Technical Roadmap and Milestones

Milestone 1: Setup the Development Environment

- **Task:** Set up Next.js frontend.
- **Task:** Set up Sanity CMS and create schemas for products and users.

- **Task:** Implement the basic navigation and layout for product pages.

Milestone 2: Product Management

- **Task:** Implement API for fetching product data from Sanity CMS.
- **Task:** Create a product listing page and ensure that data is displayed correctly.
- **Task:** Create the product detail page with image and description.

Milestone 3: User Authentication & Cart System

- **Task:** Set up user registration and login system.
- **Task:** Implement shopping cart functionality.
- **Task:** Display cart items and calculate total price dynamically.

Milestone 4: Checkout & Payment Integration

- **Task:** Implement the checkout page and integrate with the payment gateway (Stripe or PayPal).
- **Task:** Save the order details in Sanity CMS after payment confirmation.

Milestone 5: Shipment Tracking Integration

- **Task:** Integrate Stripe for payments. • Integrate ShipEngine for shipping and tracking.
- **Task:** Display shipment status on the user profile/order page.

Milestone 6: Final Testing & Deployment

- **Task:** Test the full marketplace flow (product browsing, checkout, payment, shipment tracking).
- **Task:** Deploy the project on Vercel (for frontend) and Sanity (for CMS).
- **Task:** Prepare the production environment and finalize documentation.

Collaboration and Version Control

- **GitHub Repository:** All development will be version-controlled using GitHub. I'll ensure the code is modular, with each feature in a separate branch, and merge branches for key milestones.
- **Collaborative Tools:** Communication will happen through Slack or Discord to discuss tasks and troubleshoot issues in real time.
- **Peer Review:** Code will be peer-reviewed to ensure it follows best practices and is scalable.