Day 02 Of Hackathon

Submission Title:

Marketplace Technical Foundation — Avion General E-Commerce

System Architecture Overview:

- Front End: Next.js
- Sanity CMS → Product Data API
- Third Party APIs:
  - Shipment Tracking API
  - Payment Gateway

Component Description:

1. Front End (Next.js):

- Provides a seamless user interface for browsing, purchasing, and managing orders.
- Dynamic pages include:
  - Home
  - Product Listing
  - Product Details
  - Cart
  - Checkout
  - Order Confirmation

2. Sanity CMS:

- Functions as the backend for managing products, orders, and customer data.
- Customizable schemas to efficiently handle various data entities.

3. Third-Party APIs:

- **Shipment Tracking API:** Retrieves real-time shipment status.
- **Payment Gateway:** Securely processes payments.

Key Workflows:

1. User Registration:

- **User signs up through a frontend form.**
- **Data is validated on the frontend and sent to Sanity.**
- **Sanity stores the user details securely.**

2. Product Browsing:

- **Users navigate through product listings.**
- **Frontend fetches product data via Sanity's API.**
- **Products are dynamically displayed based on availability.**

3. Order Placement:

- **Users add items to their cart and proceed to checkout.**
- **Checkout details are sent to Sanity CMS via API.**
- **Sanity processes and stores the order details.**
- **Users receive order confirmation notifications.**

4. Shipment Tracking:

- **Users request their order status.**
- **Frontend fetches shipment details via the third-party API.**
- **Real-time shipment status is displayed to users.**

API Requirements:

1. Fetch Products:

- Endpoint: /products
- Method: GET
- Description: Retrieves all available products from Sanity.
- Response Example:

```
[
 {"id": 1, "name": "Product 1", "price": $120, "stock": 70},
 {"id": 2, "name": "Product 2", "price": $180, "stock": 40}
]
```

2. Create an Order:

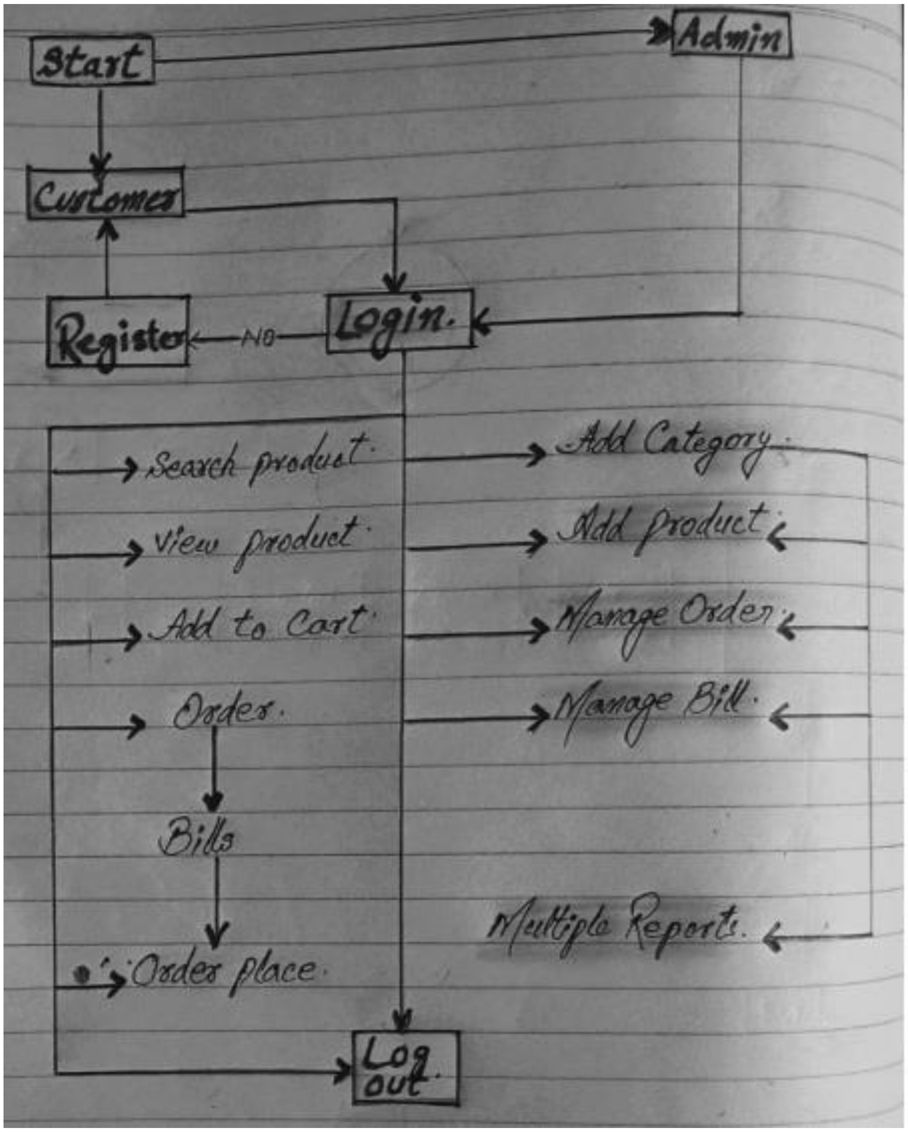
- Endpoint: /order
- Method: POST
- Description: Creates a new order with customer details and product information.
- Payload Example**:**

```
{
 "Customer Info": {
  "name": "Abbas",
  "email": "Abc000@gmail.com" },
 "products": [
  {"id": 1, "quantity": 5},
  {"id": 2, "quantity": 3}
 ],
 "Payment Status": "Paid"
}
```

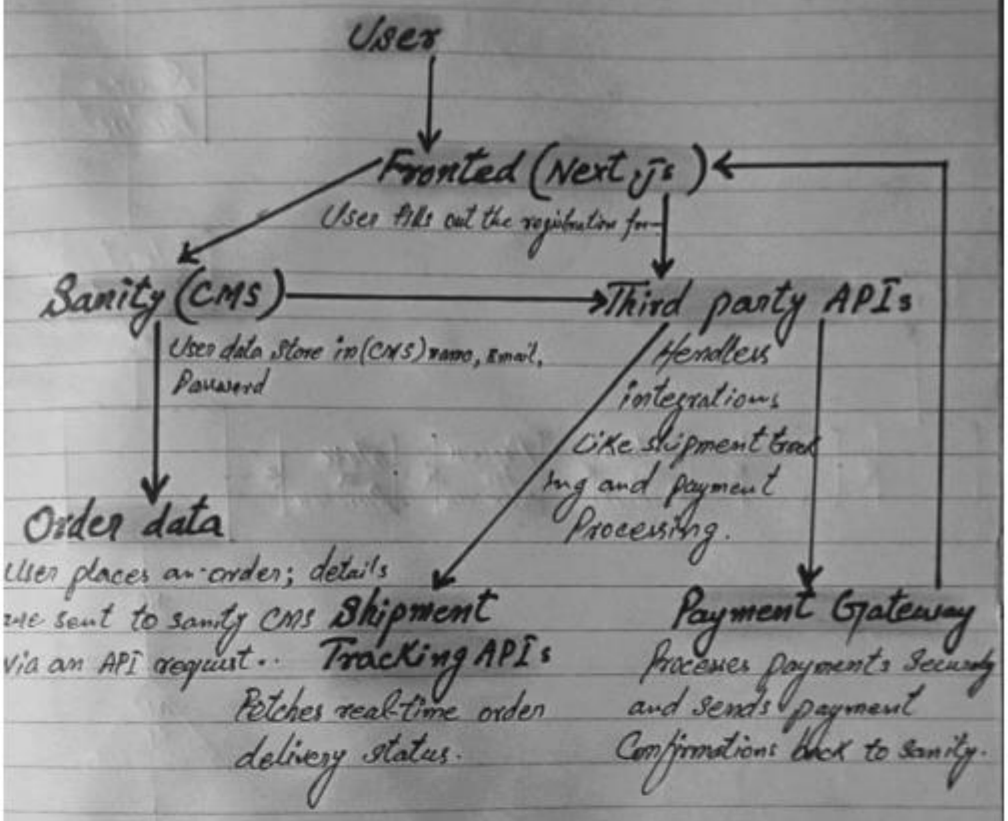
3. Shipment Tracking:

- Endpoint: /shipment
- Method: GET
- Response Example:

```
{
  "Shipment Id": "55555",
  "Order Id": "8765",
  "status": "In Process",
  "Expected Delivery Date": "2025-02-17"
}
```


Start
Admin
Customer
Register
NO
Login.
Search product.
View product.
Add to Cart.
Order.
Bills
Order place.
Log out.
Add Category.
Add product.
Manage Order.
Manage Bill.
Multiple Reports.

System Architecture Overview



User fills out the registration form

User data Store in (CMS) name, Email, Password

Hendless integrations like shipment tracking and payment Processing.

User places an order; details are sent to sanity CMS via an API request..

Fetches real-time order delivery status.

Processes payments Securely and sends payment Confirmations back to sanity.