

Marketplace Hackathon

Day # 2

Define Technical Requirements

① Frontend:

* PAGES:

- Home: display featured products
- Product listing: show all products with sorting and filtering options
- Product details: Detailed view of a product with reviews and ratings.
- Cart: Items added to the cart for purchase
- Checkout: Finalize orders with payment options.
- Order confirmation: Show order summary with tracking information.

* DESIGN:

- User-friendly and responsive for mobile/desktop.
- Real-time updates (e.g. reviews or stock changes)

② Sanity CMS (Backend):

Purpose: Manage:

- Products (e.g: name, price, stock)
- Customers (e.g: names, addresses)
- Orders (e.g: items purchased, order status)

③ Third-Party APIs:

- Payment Gateway: Handle secure transaction
- Shipment Tracking: Provide real-time delivery updates.

System Architecture Design

① Flow Diagram:

- * Frontend: user interacts with UI (browse products, add to cart)
- * Sanity CMS: Data fetched/stored via APIs for products, orders, and customers.
- * Third-Party APIs:
 - Shipment tracking fetched for users
 - Payment gateway for secure transaction

② Workflow Examples:

User Browsing:

User → Frontend → Product API → Display Products

Order placement:

User → Frontend → Create Order API

Shipment Updates:

User → Frontend → Shipment API → Real-time updates

API Requirements

① Fetch All Products:

- Endpoint: /products

- Method: GET

- Response:

```
{ "id": 1, "name": "Laptop", "price": 500 }
```

② Create Order,

- Endpoint: /orders

- Method: POST

- Payload:

```
{ "customerId": 123, "productId": 456, "quantity": 2 }
```

- Response:

```
{ "orderId": 789, "status": "confirmed" }
```

③ Track shipment:

- Endpoint: /shipment

- Method: GET

- Response:

```
{ "shipmentId": 1234, "status": "In Transit", "ETA": "2 days" }
```


API Requirements

① Fetch All Products:

- Endpoint: /products
- Method: GET
- Response:

```
{ "id": 1, "name": "Laptop", "price": 500 }
```

② Create Order:

- Endpoint: /orders
- Method: POST
- Payload:

```
{ "customer Id": 123, "product Id": 456,  
  "quantity": 2 }
```

- Response:

```
{ "Order Id": 789, "status": "confirmed" }
```

③ Track shipment:

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

```
{ "shipment Id": 1234, "status": "In Transit", "ETA": "2 days" }
```