

# MY TASK FOR DAY 2

## PLANNING THE TECHNICAL FOUNDATION

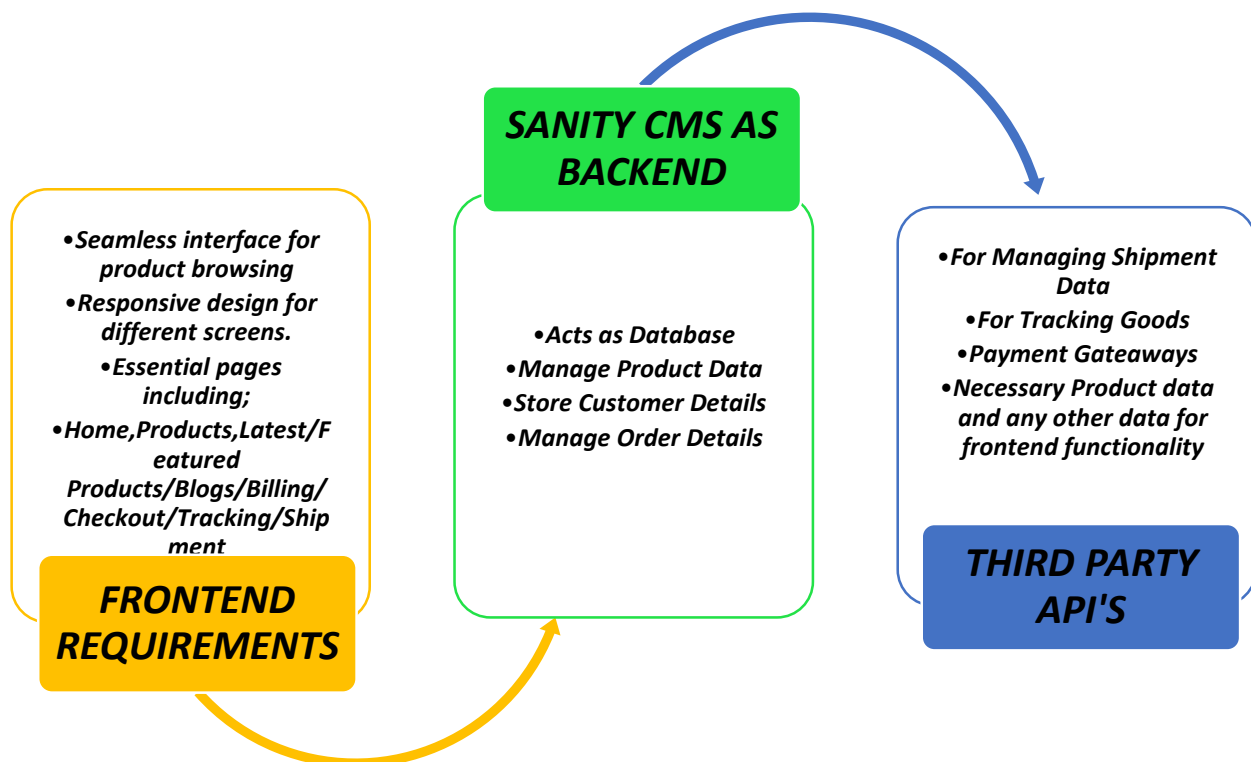
### Day 2 Goal:

The primary goal of Day 2 is to transition from the business-oriented planning of Day 1 to the technical preparation required to build your marketplace which includes;

- ❖ create a high-level technical plan including;
  1. System architecture,
  2. Workflows,
  3. API requirements

### Blueprint for The Implementation Phase.

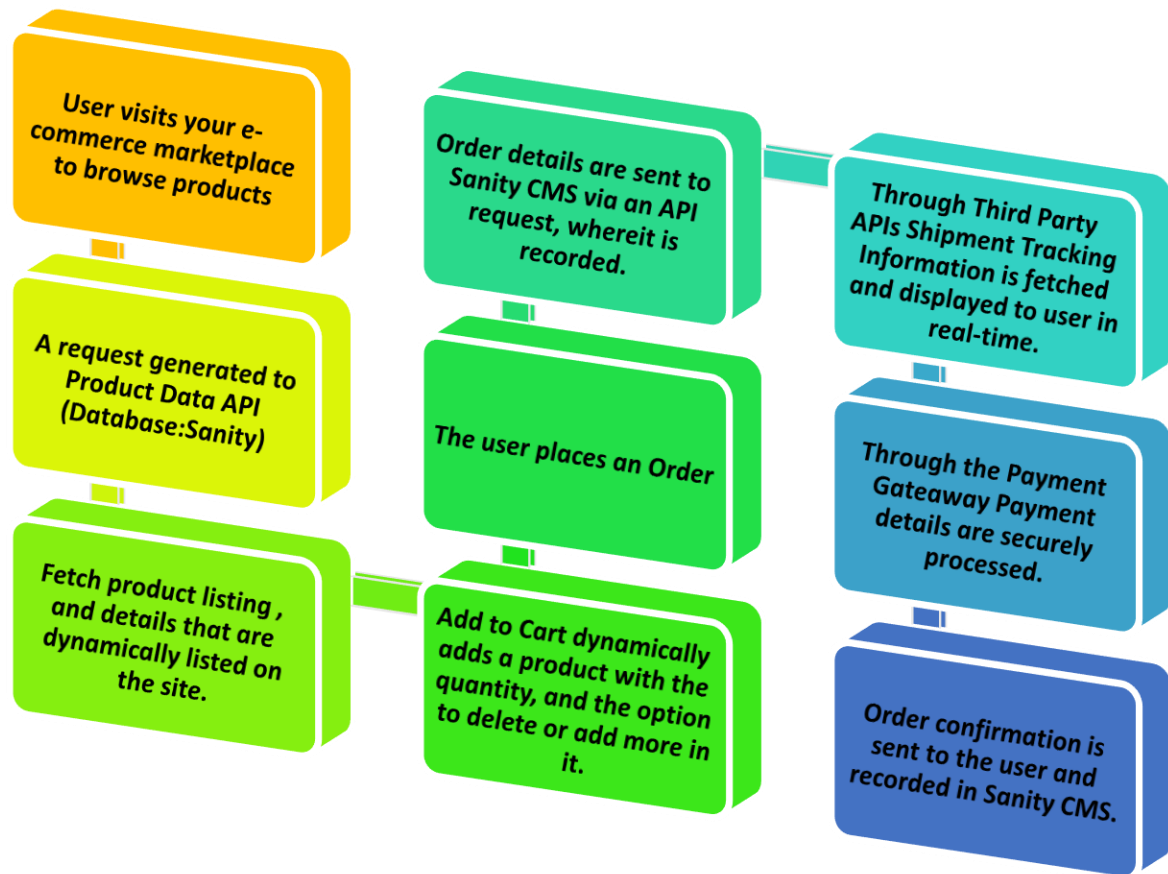
#### 1. Define Technical Requirement



## 2. Design System Architecture

This detailed workflow demonstrates how components interact in a real world scenario and how data flows seamlessly between them.

This demonstrates how frontend interactions are supported by Sanity CMS for content management, third-party APIs for logistics, and payment gateways for transaction processing.



### 3. Plan API Requirements

#### API ENDPOINTS IN TABULAR FORM

❖ <i>/products</i>	<i>GET</i>	<i>Fetch all the details of the product</i>	<i>{ "id": 1, "name": "Product A", "price": 100 }</i>
❖ <i>/orders</i>	<i>POST</i>	<i>Create a new order in Sanity</i>	<i>{ "id": 1, "customer-name": "XYZ", "Order-id": 000, "product details": abc, "payment status": success }</i>
❖ <i>/shipping</i>	<i>GET</i>	<i>Track order status via third-party API.</i>	<i>{ "Shipment-id": 000, "Order-id": "000", "Status": processing, "product details": abc, "estimated delivery date": 25-01-2025 }</i>
❖ <i>/tracking</i>	<i>GET</i>	<i>Order status updates fetched via 3rd-party API</i>	<i>{ "Order-id": "000", "Status": processing, "product details": abc, "estimated delivery date": 25-01-2025 }</i>

Source: Fictitious data, for illustration purposes only

## 4. Sanity Schema Example

### ▪ **Product Schema:**

```
export default {  
  name: 'product',  
  type: 'document',  
  fields: [  
    { name: 'name', type: 'string', title: 'Product Name' },  
    { name: 'price', type: 'number', title: 'Price' },  
    { name: 'stock', type: 'number', title: 'Stock Level' },  
    { name: 'category', type: 'string', title: 'Category' },  
    { name: 'description', type: 'text', title: 'Description' },  
    { name: 'image', type: 'image', title: 'Product Image' },  
  ],  
};
```

### ▪ **Customer Schema:**

```
export default {  
  name: 'customer',  
  type: 'document',  
  fields: [  
    { name: 'firstName', type: 'string', title: 'First Name' },  
    { name: 'lastName', type: 'string', title: 'Last Name' },  
    { name: 'email', type: 'string', title: 'Email' },  
    { name: 'phone', type: 'string', title: 'Phone Number' },  
    { name: 'address', type: 'text', title: 'Address' },  
  ],  
};
```

### ▪ **Order Schema:**

```
export default {  
  name: 'order',  
  type: 'document',  
  fields: [  
    { name: 'orderNumber', type: 'string', title: 'Order Number' },  
    { name: 'customer', type: 'reference', to: [{ type: 'customer' }], title: 'Customer' },  
    { name: 'products', type: 'array', of: [{ type: 'reference', to: [{ type: 'product' }] }], title: 'Products' },  
    { name: 'totalAmount', type: 'number', title: 'Total Amount' },  
    { name: 'status', type: 'string', title: 'Order Status' },  
    { name: 'date', type: 'datetime', title: 'Order Date' },  
  ],  
};
```

### ▪ **Billing Schema:**

```
export default {  
  name: 'billing',  
  type: 'document',  
  fields: [  
    { name: 'order', type: 'reference', to: [{ type: 'order' }], title: 'Order' },  
    { name: 'billingAddress', type: 'text', title: 'Billing Address' },  
    { name: 'paymentMethod', type: 'string', title: 'Payment Method' },  
    { name: 'paymentStatus', type: 'string', title: 'Payment Status' },  
  ],  
};
```

### ▪ **Shipping Schema:**

```
export default {  
  name: 'shipping',  
  type: 'document',  
  fields: [  
    { name: 'order', type: 'reference', to: [{ type: 'order' }], title: 'Order' },  
    { name: 'shippingAddress', type: 'text', title: 'Shipping Address' },  
    { name: 'deliveryDate', type: 'datetime', title: 'Expected Delivery Date' },  
    { name: 'shippingStatus', type: 'string', title: 'Shipping Status' },  
    { name: 'courier', type: 'string', title: 'Courier Service' },  
  ],  
};
```

### ▪ **Tracking Schema:**

```
export default {  
  name: 'tracking',  
  type: 'document',  
  fields: [  
    { name: 'order', type: 'reference', to: [{ type: 'order' }], title: 'Order' },  
    { name: 'trackingNumber', type: 'string', title: 'Tracking Number' },  
    { name: 'status', type: 'string', title: 'Delivery Status' },  
    { name: 'lastUpdated', type: 'datetime', title: 'Last Updated' },  
    { name: 'location', type: 'string', title: 'Current Location' },  
  ],  
};
```

## FLOWCHART REPRESENTING WORKFLOW

Products["Products"]

Customers["Customers"]

Orders["Orders"]

Billing["Billing"]

Shipping["Shipping"]

Tracking["Tracking"]

-----

Customers -->|Places| Orders

Orders -->|Contains| Products

Orders -->|Has| Billing

Orders -->|Has| Shipping

Shipping -->|Tracked by| Tracking

