1. Fronted Requirements.

1. Home Page.

- Display Featured products os Categories.
- Include a search bar and navigition to other essential pages.

2. Product Listing Page.

- List products dynamically Fetched from the Sanity CMS.
- Filters (e.g. price, category) and sorting options.

3. Product Details Page.

- Display detailed product information (title, description, price, inventory, images).
- Include an "Add to Cart" button.

4. Cart Page.

- Show selected Products quantities and total Price.
- Option to remove items or adjust quantities.

5. Checkout Page.

Form for user information (name, address, payment details).
Integration with a payment geteway.

6. Order Confirmation Page.

Display order summary and Confirmation message.

Order Schema:

```
{ name: "order",
  Fields: [
    {name": "customer", type: "reference", to: "customer"}
    {name: "product", type: "array" oF: [{type: referen
    ,to: "product"}]},
  ]
}.
```

Third Party API Integration
Shipment Tracking API
Real-time tracking updates of user
orders.
API Endpoint: /track/{order Id}.

Payment Gateway API
Secure payement processing for order.
API Endpoint: /Payement/process.

Other API
Currency conversion, notifications, etc as needed.

End point Name /express-delivery-status.
Method: GET
Response:
```
{ order: 123,
  Status: "In Transit"
  ETA: 15mins
}
```

2 Backend (Sanity CMS)

Scope:

Sanity CMS will handle the following

- Product Data:
Title, price, inventory, image, description, etc

- Customer Details:
Name, address, Phone Number.

- Order Records:
Product ordered, total price, use details

1- Product Schema:

```
{ "name": "product";
  "Fields": [
    { "name": "title", "type": "string" },
    { "name": "price", "type": "number" };
  ]
}
```

2. Customer Schema:

```
{ name: "customer",
  "Fields": [
    { "name": "name", "type": "string" };
    { "name": "Phone", "type": "number",
      "name", "address", "type": "text" };
  ]
}
```

Fronted Requirements.

1. Home Page.

- Display Featured products or Categories.
- Include a search bar and navigition to other essential pages.

2. Product Listing Page.

- List products dynamically Fetched from the Sanity CMS.
- Filters (e.g. price, category) and sorting options.

3. Product Details Page.

Display detailed product information (title, description, price, inventory, images).

Include an 'Add to Cart' button.

4. Cart Page.

Show selected Products quantities and total Price.

Option to remove items or adjust quantities.

5. Checkout Page.

Form for user information (name, address, payment deta..

Integration with a payment gateway.

6. Order Confirmation Page.

Display order summary and Confirmation message.

2 | Backend (Sanity CMS)

Store.

Sanity CMS will handle the following

- Product Data:
  Title, price, inventory, image, description, etc

- Customer Details:
  Name, address, Phone Number.

- Order Records:
  Product ordered, total price, user details.

1- Product Schema:

```
{ "name": "product";
  "Fields": [
  { "name": "title", "type": "string" };
  { "name": "price", "type": "number" };
  ]
}
```

2- Customer Schema:

```
{ name: "customer";
  "Fields": [
  { "name": "name", "type": "string" };
  { "name": "phone", "type": "number";
  { "name": "address", "type": "text" };
  ]
}
```

Order Placement

User adds products to the cart and proceed to checkout.

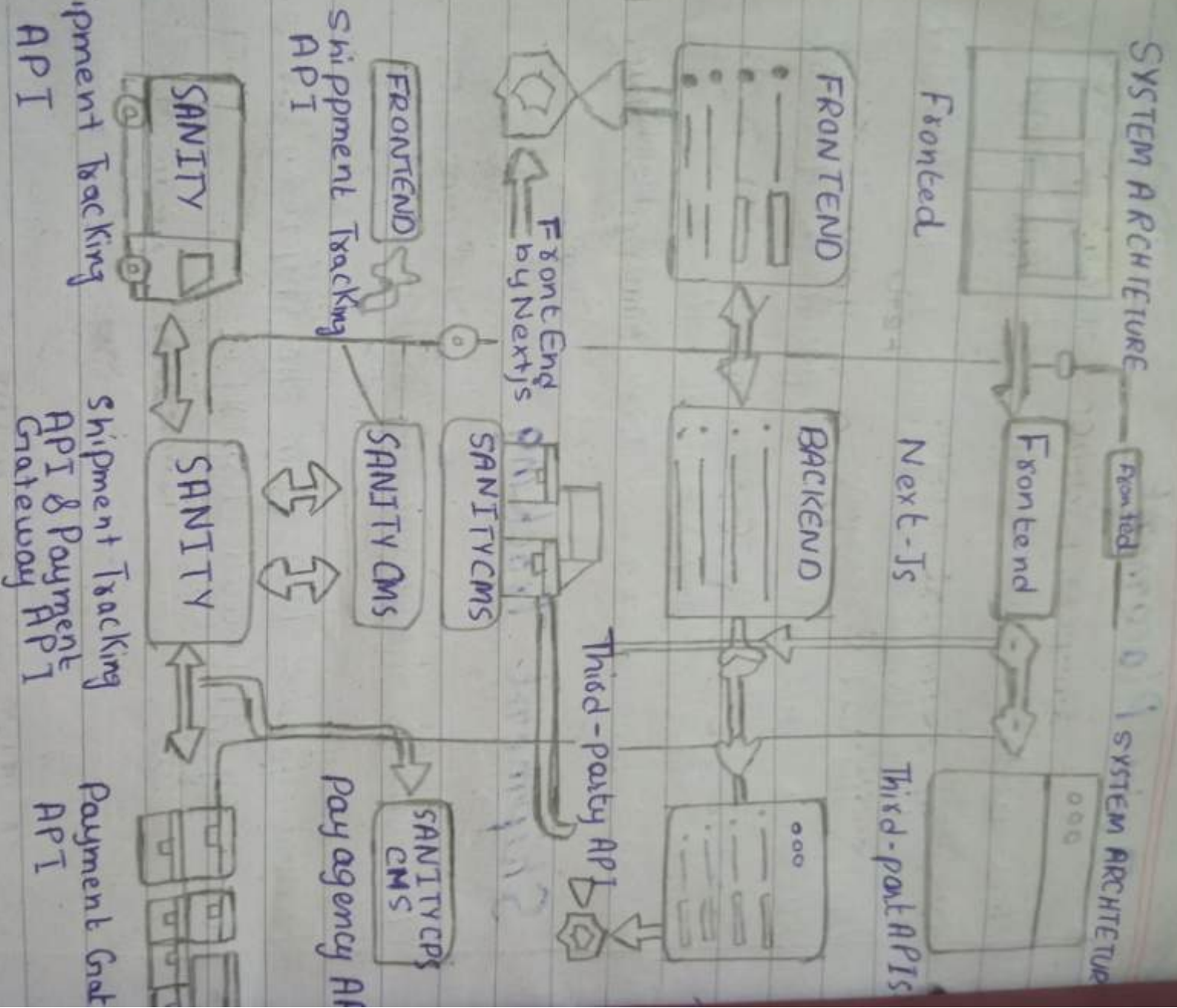Fronted sends the order details to the backend (using the order Schema.

Backend interacts with the payment getway to process the payment.

Upon successful payement, the backened creates an orders record in Sanity CMS.

Shipment Tracking

User checks the status of their order.

Fronted fetches data from the shipment tracking API and display it.

SYSTEM ARCHITECTURE

Fronted

Fronted

Next-Js

Third-part APIs

FRONTEND

FRONTEND

BACKEND

Third-party API

Front End
by Nextjs

SANITYCMS

SANITY CMS

SANITYCMS
CMS

SANITY

SANITY

Pay agency A

Shipment Tracking
API

Shipment tracking
API

Shipment Tracking
API & Payment
Gateway API

Payment Gat
API

Payment
Gateway API

Product Browsing:

Uses visits the Fronted.

Fronted sends a request to fetch the products
from Sanity CMS.

Products are displayed dynamically on the side