

Day 2 Hackathon:

Transitioning to Technical Planning:

Frontend Requirements:

- 1) Firework: **Next JS**
- 2) Responsive design for Mobile, Tablet & Laptop by using Features Navigation, Grid Layouts, Search and Filters, Content Priority, Images in **Tailwind CSS**.
- 3) **Essential Pages** which I use in my Website Home, About us, Products, Contact us. And Products child pages, Product Category, Add to Chart, Order Confirmation.

Backend Requirements:

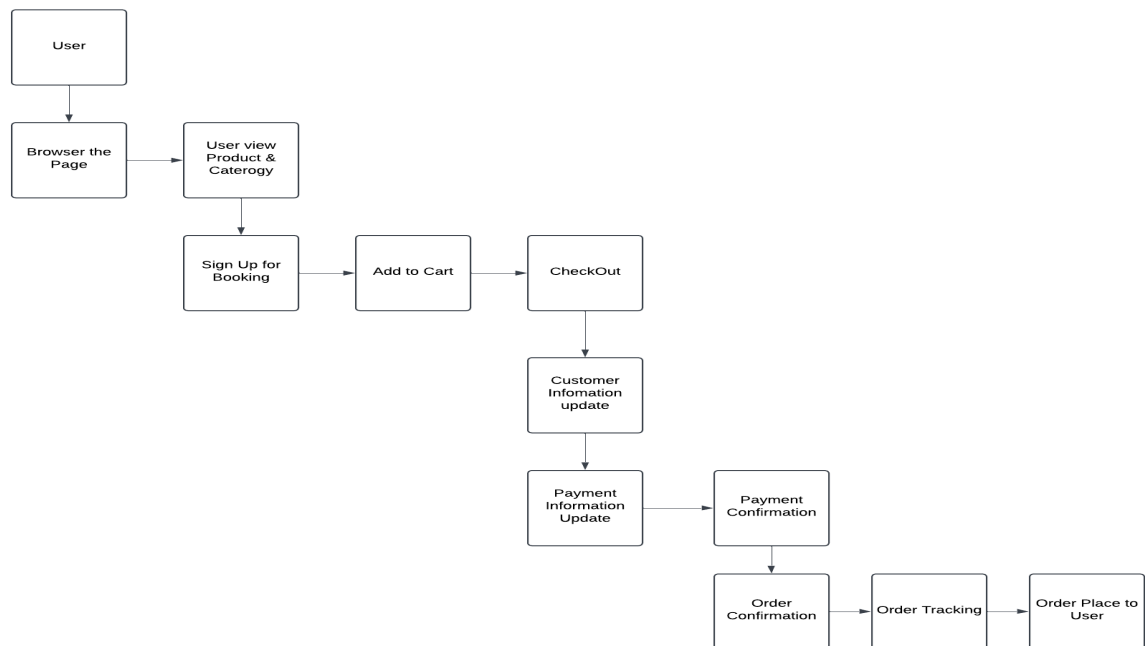
- 1) **Sanity** use for Real-time collaboration and flexible content modeling, Studio (admin panel) can be hosted or integrated into apps, Offers a generous free tier for small projects.

Third Party API:

- 1) Shipment Tracking using **DHL, TCS, Leopard**.
- 2) Payment Gateway: **EasyPaisa, Jazz Cash, HBL Connect**.
- 3) Email Services: **Cloud Server**.

Design System Architecture

Work Flow Chart



Work Flow details

- 1) User visit Rental Website and send request to browser.
- 2) API received the request and send to Sanity CMS getting data.
- 3) Sanity CMS send the Product data to API and API send the result to User.
- 4) User view the Page and Select the product which received from Sanity.
- 5) User add the Product in ADD to Cart and send request to API and API send request to Data Base and Post order.
- 6) Data Base request to API for collect customer data and information for Post Order to address.
- 7) API ask request to User to provide information, payment process for post the order and tracking shipment.
- 8) If any details of customer provide wrongly API send error to User as failure.

1. Plan API requirements

- a. Endpoint name: **/Products**
 - **Method:** **GET**
 - **Description:** **Fetch all** products from Sanity
 - **Response:** **Product** (ID, Name, Price, Description, Image, Rating, Category, Stock).
- b. Endpoint name: **/Customers**
 - **Method:** **POST**
 - **Description:** Get **customer's data** and add it in sanity
 - **Response:** **Customer** (ID, Name, Email, Phone, Address).
- c. Endpoint name: **/Orders**
 - **Method:** **POST**
 - **Description:** Create a **new order** in Sanity.
 - **Response:** **Order** (ID, Customer ID, Order, Date, Item, Amount, Status)
- d. Endpoint name: **/Shipment**
 - **Method:** **GET**
 - **Description:** **Track order** status via Third-party API (MOCK API)
 - **Response:** **Shipment ID**, Order ID, Status, expected delivery date, carrier, tracking number.

2. Sanity Data Schema for all endpoints:

a. Products:

```
export default {
  name: 'product',
  type: 'document',
  fields: [
    { name: 'id', type: 'number', title: 'Product Id' },
    { name: 'name', type: 'string', title: 'Product Name' },
    { name: 'price', type: 'number', title: 'Price' },
    { name: 'Description', type: 'string', title: 'Description' },
    { name: 'Image', type: 'string', title: 'Image' },
    { name: 'Rating', type: 'number', title: 'Rating' },
    { name: 'Stock', type: 'number', title: 'Stock' }
  ]
}
```

b. Customers:

```
export default {
  name: 'customer',
  type: 'document',
  fields: [
    { name: 'id', type: 'number', title: 'Customer Id' },
    { name: 'name', type: 'string', title: 'Customer Name' },
    { name: 'email', type: 'string', title: 'Email' },
    { name: 'phone', type: 'number', title: 'Phone Number' },
    { name: 'address', type: 'string', title: 'Address' },
  ]
}
```

c. Orders:

```
export default {
  name: 'order',
  type: 'document',
  fields: [
    { name: 'id', type: 'number', title: 'Order Id' },
    { name: 'customerId', type: 'number', title: 'Customer Id' },
    { name: 'email', type: 'string', title: 'Email' },
    { name: 'phone', type: 'number', title: 'Phone Number' },
    { name: 'address', type: 'string', title: 'Address' },
  ]
}
```

d. Shipment:

```
export default {  
  name: 'shipment',  
  type: 'document',  
  fields: [  
    { name: 'id', type: 'number', title: 'Shipment Id' }  
    { name: 'orderId', type: 'number', title: ' Order Id' },  
    { name: 'status', type: 'string', title: 'Order Status' },  
    { name: 'date', type: 'number', title: 'Delivery Date' }  
  ]  
}
```

e. Payment:

```
export default {  
  name: 'payment',  
  type: 'document',  
  fields: [  
    { name: 'id', type: 'number', title: 'Payment Id' }  
    { name: 'customerId', type: 'number', title: 'Customer Id' },  
    { name: 'date', type: 'number', title: 'Payment Date' },  
    { name: 'amount', type: 'number', title: 'Paid Amount' }  
    { name: 'status', type: 'string', title: 'Payment Status' }  
    { name: 'method', type: 'string', title: 'Payment Method' }  
    { name: 'gateway', type: 'string', title: 'Payment Gateway' }  
  ]  
}
```