

①

Day: 2

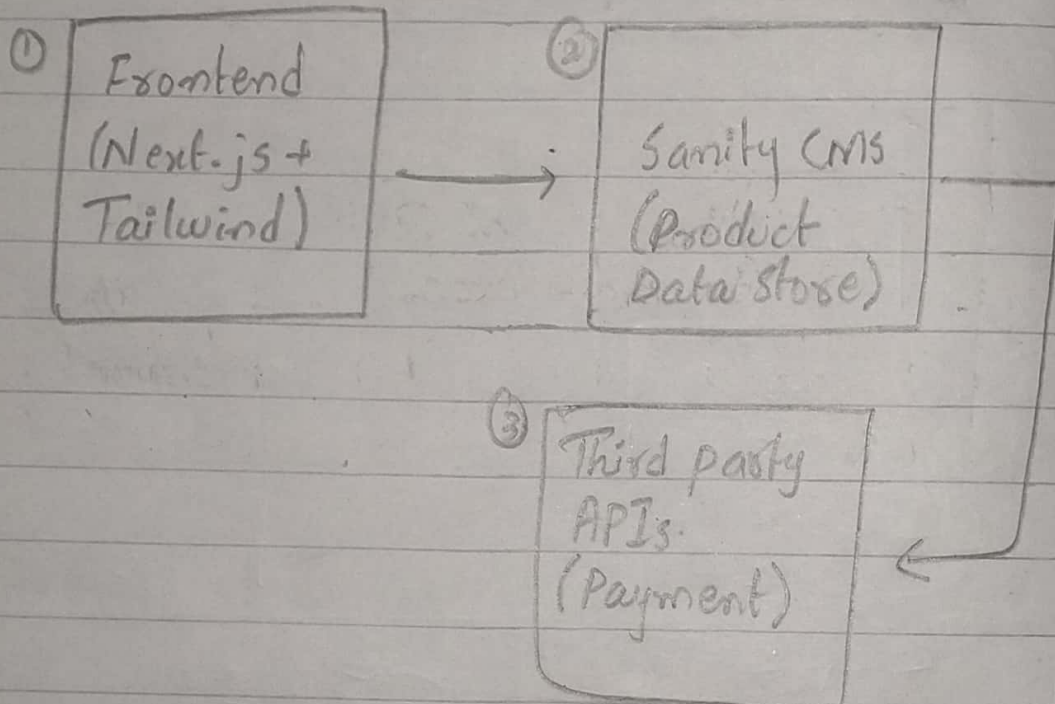
Date: 16-01-2025 Hackathon: 3

① Name:- Mahnoor

Roll No:- 00070019

Slot: Sunday, Morning 9 to 12

① System Architecture Overview:-



→ Briefly Describe each components:-

① Frontend (Next.js and Tailwind).

→ It helps users browse products, add them to the cart, and payment for them. It links to the Sanity to get data.

→ user friendly.

→ Responsive all Device.

(2)

② Sanity CMS:-

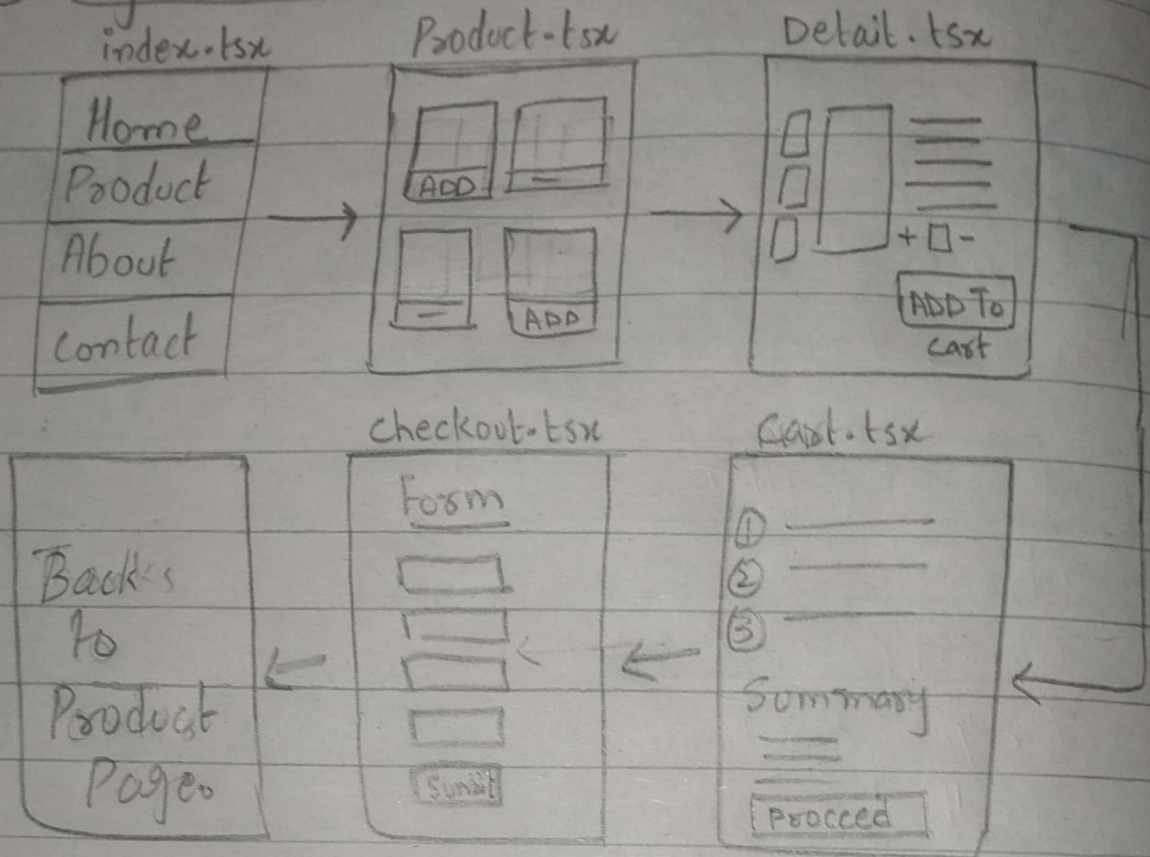
→ Sanity is where the product information like:- product Name, prices, descriptions, category, stock, etc is stored. It helps the backend fetch and display this data on the Frontend.

③ Third - Party APIs:-

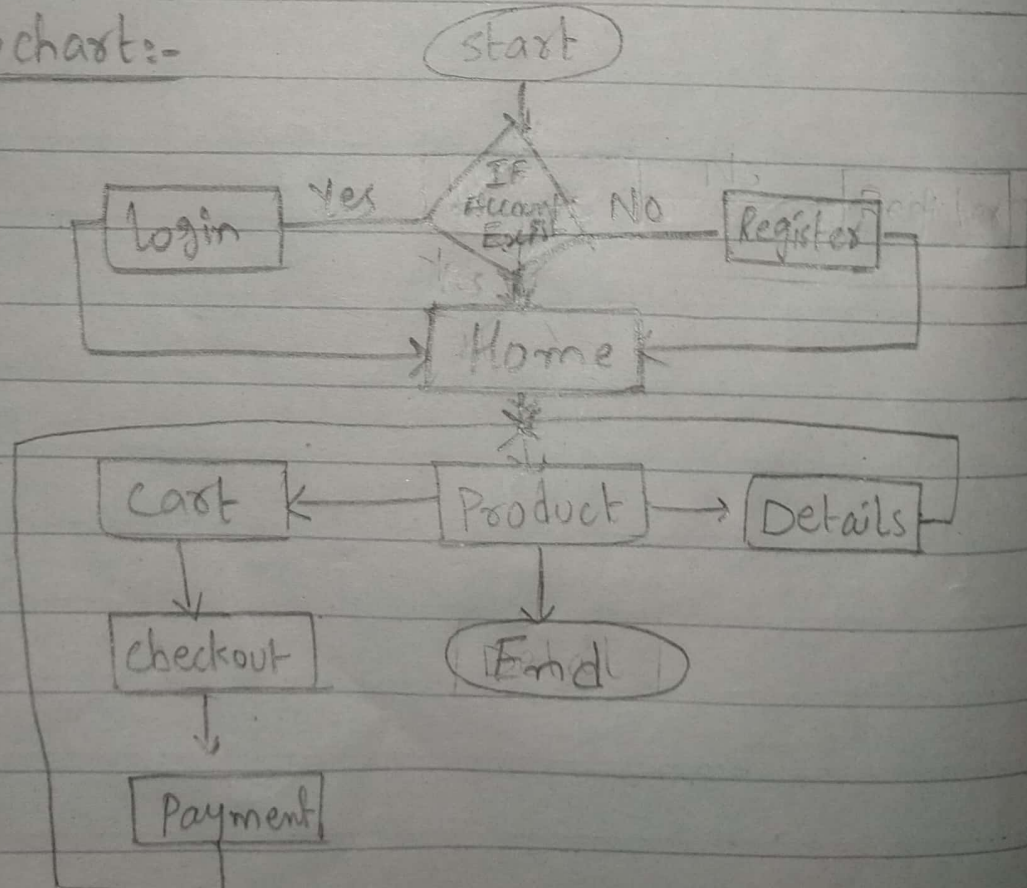
→ Handles the payment process when users pay for their orders. Like, paypal, Easy Paisa, Jazzcash and other banking method.

③

② Key Work Flows:-



* Flowchart:-



(4)

(3) Workflows Detail:-

(1) User Adds Product to Cart:-

- A user browse products and adds them to their cart.
- The Frontend updates the cart with the selected products.
- The backend checks Sanity CMS for Product details (name, price, stock*, category, quantity, etc). and updates the cart.
- The cart page shows the current products the user added.

(2) User Makes payment Detail:-

- The user enters payment detail form and submit the order.
- The backend process the payment using the payment API like; paypal, easypaisa and Jazzcash. etc.
- The user see update on the delivery status.

* Stock:- is bcz use horaha qk sometimes event day pe product khatam hojati hai like:- 14 August pe Bakery wegera mai cake ki shortage hojati hai.

(5)

④ API Requirements.

① Authentication.

Method 8- POST /login

Description:- Sign in Already register users.

Example:- $\{ "email": "abc@gmail.com", "password": "Password123" \}$

→ Same as registration but Data schema are different.

②

② Product:-

Method 9- GET /products

Description:- Fetch all available Products

Example:- $[\{ "id": 1, "name": "Pizza", "Price": 1200, "category": "Fast Food" \}]$

③ Product Detail:-

Method:- GET /products/{id}

Description:- Fetch details of a single Product.

Example:- $\{ "id": 1, "name": "Pizza", "Price": 1200, "Description": "Delicious cheese Pizza" \}$.

⑥

④ Order /cart:-

Method:- POST /cart

Description:- New order Places.

Example:- { "userId": 1, "cartItems":
[{ "productId": 1, "Quantity": 2 }] }

→ GET method se old history fetch ho jae gi user ki.

7

⑤ Sanity Schema Examples:-

① Product:-

export default {

name: 'product',

type: 'document',

fields: [

{name: 'name', type: 'string', title: 'Product Name'},

{name: 'Price', type: 'number', title: 'Price'},

{name: 'description', type: 'text', title: 'Description'},

{name: 'Category', type: 'string', title: 'Category'},

]};

② Cart:-

export default {

name: 'Cart',

type: 'document',

fields: [

{name: 'customerID', type: 'number', title: 'CID'},

{name: 'Products', type: 'array',

fields: [{name: 'ProductID', type: 'number'},

{name: 'quantity', type: 'number'},

],

title: 'Product'}

]};

(8)

③ Order:-

export default {

name: 'order',

type: "document",

fields: [

{ name: "customerId", type: 'number', },

{ name: 'cartItem'; type: 'array',

fields: [{ name: 'ProductID', type: 'number', }

{ name: 'quantity', type: 'number' }],

title: "cart Item",

{ name: 'status', type: 'string', title: "order status",

{ name: 'payment', type: 'string', title: "Payment",

] };

ER Diagram

