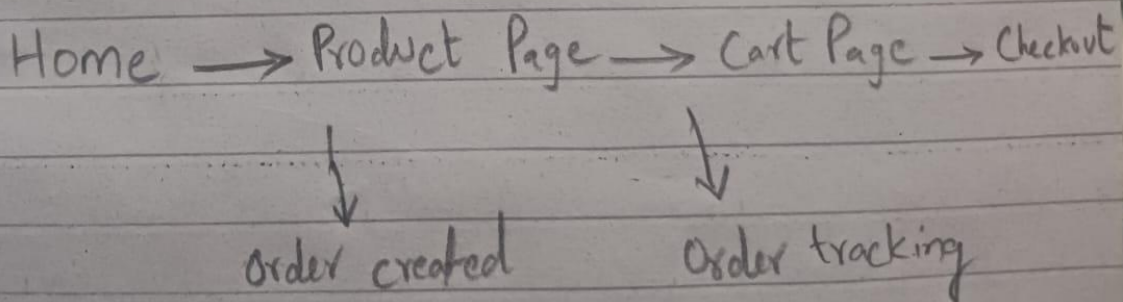


Day 2

1) TECHNICAL REQUIREMENTS:

FRONTEND (Next.js)	BACKEND (Sanity.io)	THIRD-PARTY API
Home	Product Data	Payment Gateway
Product listing	Customer Data	Shipment Tracking
Product detail	Order Records	AR Integration
Cart		
Checkout		
Order Confirmation		

2) DESIGN DATA ARCHTECTURE:



The above diagram shows how user move through the app.

③ PLAN API ENDPOINTS:

SANITY API ENDPOINTS:

a) Customer Schema (/customer):

- ① Create (POST): Add a new customer
- ② Get (GET): Retrieves customer details
- ③ Update (PUT): Modify existing information
- ④ Delete (DELETE): Removes customer

b) Order Schema (/order):

- ① Create (POST): Add a new order
- ② Get (GET): Retrieves order details
- ③ Update (PUT): Modify existing order
- ④ Delete (DELETE): Delete an order

c) Cart Schema (/cart):

- ① Create (POST): Add items to cart
- ② Get (GET): Retrieves cart content
- ③ Update (PUT): Modify items in cart
- ④ Delete (DELETE): Clear the cart

WORKFLOW:-

PRODUCT DETAIL:-

- ① Function: Shows information of selected item.
- ② User Interaction: User can add product to cart.

CART PAGE:

- ① Function: Displays items which added to cart.
- ② User Interaction: User can modify the cart by adding, editing or delete.

CHECKOUT PAGE:-

Function: Allow users to enter their details and finalize the order.

Backend actions:

- ① Create a new customer record and saves in sanity.
- ② Save the order in sanity.
- ③ Generate the shipping id once the order is successfully created.

4) TECHNICAL DOCUMENTATION:

(a) Processing: The order is received & being prepared.

(b) Shipped: The order is dispatched and tracking id assigned.

(c) Delivered: The order is successfully delivered to customer.

(d) Function: Allow customer to track their order by tracking ID.

(e) Data Management: Fetch order details and status from sanity.