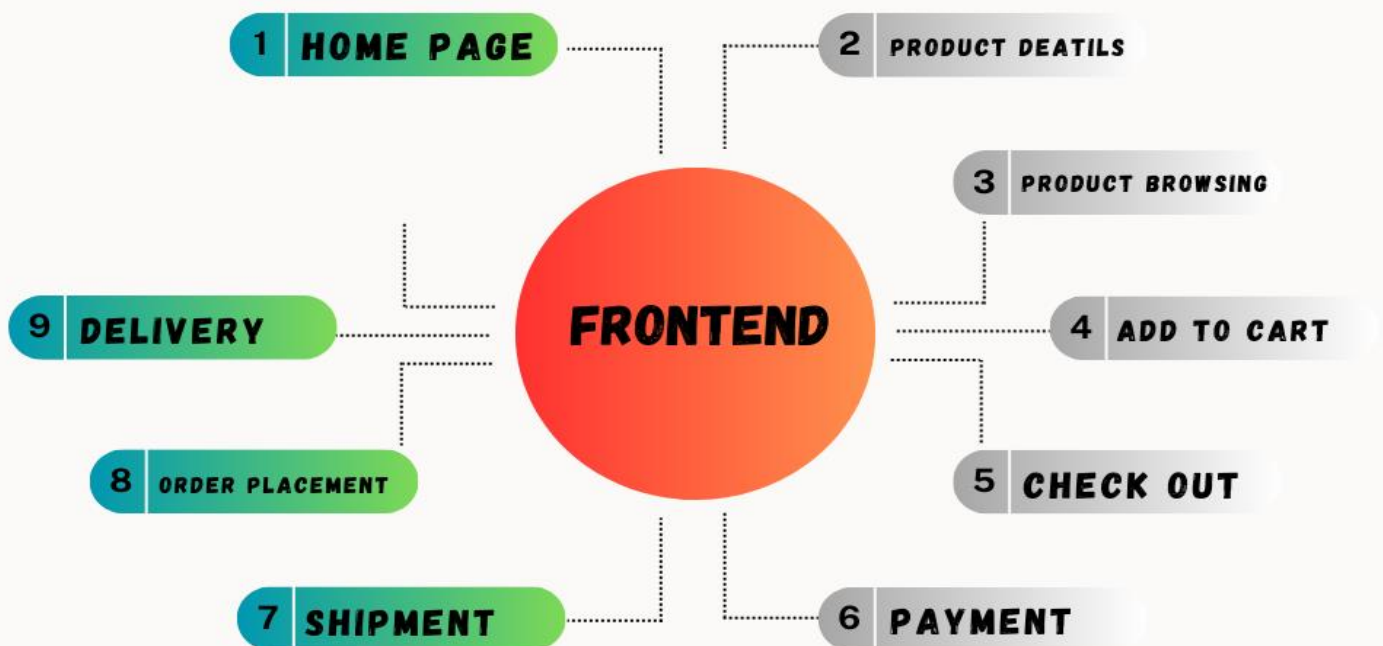**Farwa Batool**

00496911

HACKATHON DAY 02

MARKET PLACE TECHNICAL FOUNDATION

CLOTHING E COMMERCE: (SHOP – CO)

FRONTEND CONTENTS DEFINATIONS

Home Page: User lands on the Home Page. Displays featured categories, popular products and a search Bar.

Product Browsing: User selects a category or uses the search Bar. Products are displayed with options to filter by price, brands or rating.

Product Details: User clicks on a product to view its details. Page includes product description, price, availability and user's reviews.

Add To Cart: User adds the product to the Cart. Cart updates dynamically with quantity and price.

Check Out: User proceeds to the checkout page. Provides shipping address and selects a delivery option.

Order Confirmation: Order details are displayed and sent via email. Order status is updated in the backend.

Shipment Tracking: User visits the "Order History" section. Real-time shipment tracking is enabled via API integration.

Delivery Product: is delivered to the user's address.

REGISTRATION & LOGIN

USER REGISTRATION & LOGIN	
User can register or login while email / password or social media	
PRODUCT DISPLAY (CMS INTEGRATION)	
Show products fetched from a Content Management System (CMS) such as Sanity CMS	
Add to Cart:	Users select products to add to the cart. The cart dynamically updates, reflecting the selected items
Order Payment:	Users place orders with shipping information at payment details.
Shipment Tracking:	Track shipment status after order placement

PROCEDURE

USER COLLECTION

```
{
"UserID": "U002",
"Name": "SALMAN",
"EMAIL": "salman@gmail.com",
}
```

PRODUCT COLLECTION

```
{
"ProductID": "P002",
"Name" : "Jeans",
"Category": "Casual Wear"
"Price"; "2000",
}
```

CART COLLECTION

```
{
"CartID": "C002",
"UserID": "U002",
"Items": [
{
"ProductID": "P002",
"Quantity": "3"
}
]
}
```
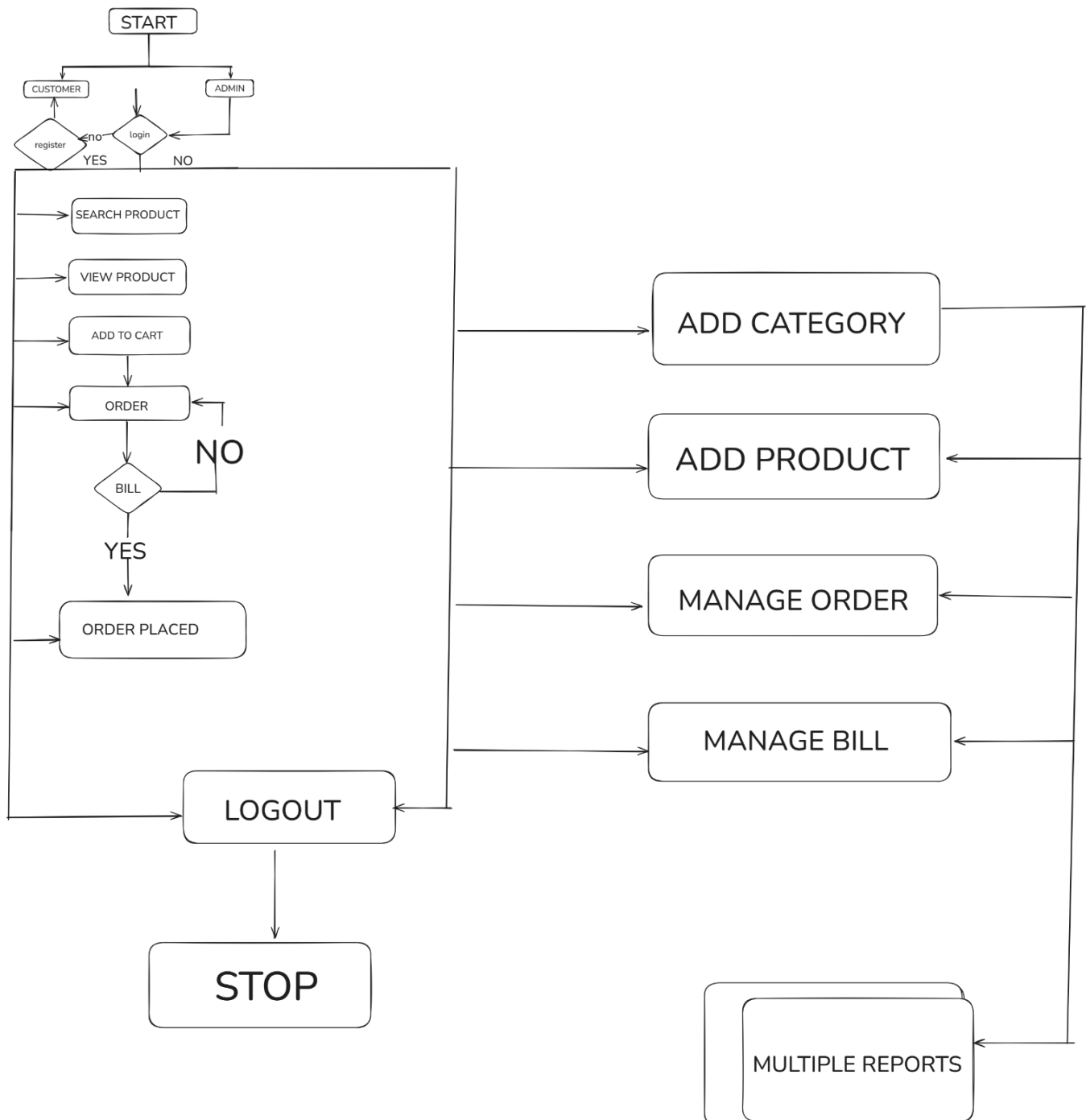
ORDER COLLECTION

```
{
"OrderID": "o001",
"UserID": "U002",
"PaymentStatus": "Paid",
}
```

API ENDPOINTS

S.NO		
1	LOGIN & REGSITRATION	POST/API/auth/register
1		POST/API/auth/login
2	FETCH PRODUCTS	GET/API/products
3	PLACE ORDER	POST/API/ order
4	ADD TO CART	POST/API/cart
5	SHIPMENT TRACKING	GET/API/shipment tracking/{tracking-number}

FRONT END FLOW CHART

START

CUSTOMER

ADMIN

register

no

login

YES

NO

SEARCH PRODUCT

VIEW PRODUCT

ADD TO CART

ORDER

NO

BILL

YES

ORDER PLACED

ADD CATEGORY

ADD PRODUCT

MANAGE ORDER

MANAGE BILL

LOGOUT

STOP

MULTIPLE REPORTS

TECHNICAL DOCUMENTATION

COMPLETE WORK FLOW OF PROJECT

1.USER REGISTRATION:

- ❖ User navigates to the registration page.
- ❖ Next User enters registration details (e.g., Name, Email, Password).
- ❖ User submits the registration form.
- ❖ The Frontend sends the Post request to the backend API to register the User.
- ❖ The backend API validates the input data.
- ❖ If valid, the backend API send a request to Sanity to store the User data.
- ❖ Sanity stores the User data in the database.
- ❖ The backend API receives a confirmation from sanity.
- ❖ The backend API sends a confirmation email to the User.
- ❖ The User receives the confirmation email.

2. PRODUCT BROWSING:

- ❖ User navigates to the product browsing page.
- ❖ The frontend sends a GET request to the Sanity API to fetch product categories.
- ❖ Sanity API retrieves the product categories data.
- ❖ The frontend receives the product categories data.
- ❖ The frontend displays the product categories to the User.
- ❖ User selects the product category.
- ❖ The frontend sends a GET request to the Sanity API to fetch products within the selected category.
- ❖ Sanity API retrieves the product data.

* ❖ **The frontend receives the product data.**
* ❖ **The frontend displays the products to the User.**

3. ORDER PLACEMENT:

* ❖ **User adds product to the cart.**
* ❖ **User navigates to the checkout page.**
* ❖ **User enters shipping and payment details.**
* ❖ **User submits the order.**
* ❖ **The frontend sends a POST request to the backend API to place the order.**
* ❖ **The backend API validates the input data.**
* ❖ **If valid the backend API processes the payment (If applicable).**
* ❖ **The backend API sends a request to Sanity to store the order details.**
* ❖ **Sanity stores the order details in the database.**
* ❖ **The backend API receives a confirmation from Sanity.**
* ❖ **The backend API sends an order confirmation email to the User.**
* ❖ **The User receives the order confirmation email.**

4. SHIPMENT TRACKING:

* ❖ **User navigates to the shipment tracking page.**
* ❖ **The frontend sends a GET request to a third-party API to fetch the order status.**
* ❖ **The third-party API retrieves the shipment status data.**
* ❖ **The frontend receives the shipment status data.**
* ❖ **The frontend displays the shipment status to the User.**
* ❖ **(Optional) the User can refresh the shipment status to GET updates.**