HACKATHON_DAY_2_TASK

PLANNING THE TECHNICAL FOUNDATION

Name: Sumayya fazal hussain
rollno.00137046.

E-COMMERCE -WEBSITE

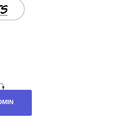
1.Technical requirements:

1. System Architecture Overview
Our e-commerce platform consists of three main components:

*Frontend: Built with Next.js for server-side rendering and optimal performance
*Backend: Utilizing Sanity CMS for content management and data storage
*Third-party APIs: Integrated for payment processing, shipment tracking, and other

2. Key Workflows
The platform supports the following key workflows:

.User Registration and Authentication
. Product Browsing and Searching
.Shopping Cart Management
.Checkout and Order Placement
.Order Tracking and Management
3. API Endpoints
The platform exposes RESTful API endpoints for various operations. Key endpoints include:

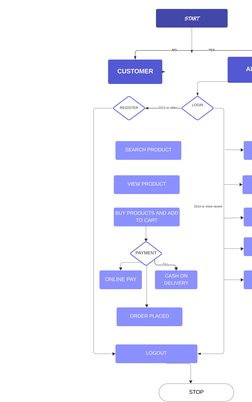


//products: For product-related operations
//orders: For order management
//users: For user account management
//cart: For shopping cart operations
4. Sanity Schema Design
Sanity CMS is used to manage the following data models:

. Products: Including details like name, price, description, and inventory
. Orders: Tracking order details, status, and associated customer
. Customers: Managing user profiles and authentication details
. Categories: Organizing products into browsable categories
5. Implementation Guidelines
When implementing the e-commerce platform, consider the following:

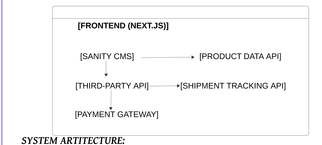
Ensure responsive design for mobile and desktop compatibility
Implement proper error handling and validation in both frontend and backend
Use appropriate caching strategies to optimize performance
Follow security best practices, especially for handling user data and payments
Implement analytics to track user behavior and platform performan

FRONTEND REQUIREMENTS

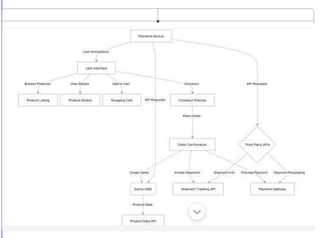


2 SYSTEM ARCHITECTURE OVERVIEW

DIAGRAM:

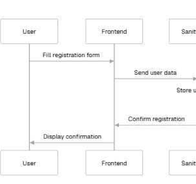


SYSTEM ARTITECTURE:

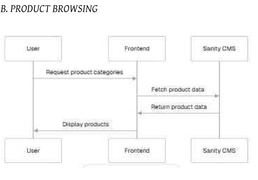


This diagram illustrates the flow of data and interactions between different components of the e-commerce system.
The frontend, built with Next.js, interacts with Sanity CMS for content management and third-party APIs for additional functionalities like shipment tracking and payment processing.

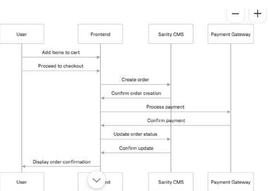
A. USER REGISTRATION WORKFLOW



B. PRODUCT BROWSING



C. ORDER PLACEMENT WORKFLOW

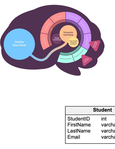


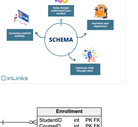
3. API Requirements

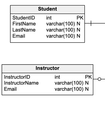
ENDPOINTS	Method	Description	Payload/Response Example
/products	GET	Fetch all available products	{ "id": 1, "name": "Product A", "price": 100, "stock": 50 }
orders	POST	Create a new order	{ "customerId": 1, "products": [{"id": 1, "quantity": 2}], "totalAmount": 200 }
/shipment/:orderId	GET	Track order shipment status	{ "orderId": 1, "status": "In Transit", "expectedDelivery": "2023-06-15" }

5. SANITY SCHEMA :

```
{
  "User": {
    "id": "string",
    "name": "string",
    "email": "string"
  },
  "Product": {
    "id": "string",
    "name": "string",
    "price": "number",
    "category": "string"
  },
  "Order": {
    "id": "string",
    "userId": "string",
    "products": ["productId"],
    "total": "number"
  }
}
```







Task 2 has been completed..!