HACKTHOON DAY 2

TECHICAL FOUNDATION

Frontend

Technologies :

Next.js for server _side rendering and faster page loads . thailwind css for styling and responsive design.

Key features :

Build a user-friendly interface for browsing products and placing orders.

Implement core pages

Home ,product listing , product details , cart and checkout, order confirmation

Sanity CMS as Backend

Purpose:

. use sanity cms to manage product data custromer Records, and order information

. define structure content schemas for:

. product Details (ID, name , price, stock, images)

. orders (order ID, customer ID, products IDs , quantity, status)

. customer (customer Id , contact info , and purchase History)

. Real-time updates to reflect stock changes and order processing

Third-party ApIs

shipEngine:

for shipment tracking and logistics.

Enables assigning delivery riders and tracking shipment zones .

Payment gateway:

Support cash on Delivery and digital payments (stripe ,paypal, or jazzcash)

Notification Api :

Use services like twilio or firebase for SMs/Email

Notofications about order status.

Api Requirements

1) Product management api

Endpoint Name : /products

Method: Get

Description: Fetch all available products from sanity CMS .

Response Example :

[{ “id” : 1   “name”: “product A”, “prices”: 100}

{ “id”:2, “name”: “product B “   “prices”:200}

]

Api Requirements

2)order and management api

Endpoint Name : /orders

Method: post

Description: create a new order in the system

Response Example :

{“order”_id : 456, “satus”: “succcess”,

“message”: “order created successfully”

Api Requirements

3) shipment management Api

Endpoint Name : /shipment

Method: Get

Description: track the order status via a third-party api

Response Example :

{ “shipment-id”: 789 , “order-id”: 456 , “status”: “In transit”, “expected -delivery _dates ” : “2025-01-20”}

Api Requirements

1) Express Delivery status Api

Endpoint Name : /express-dfelivery-status

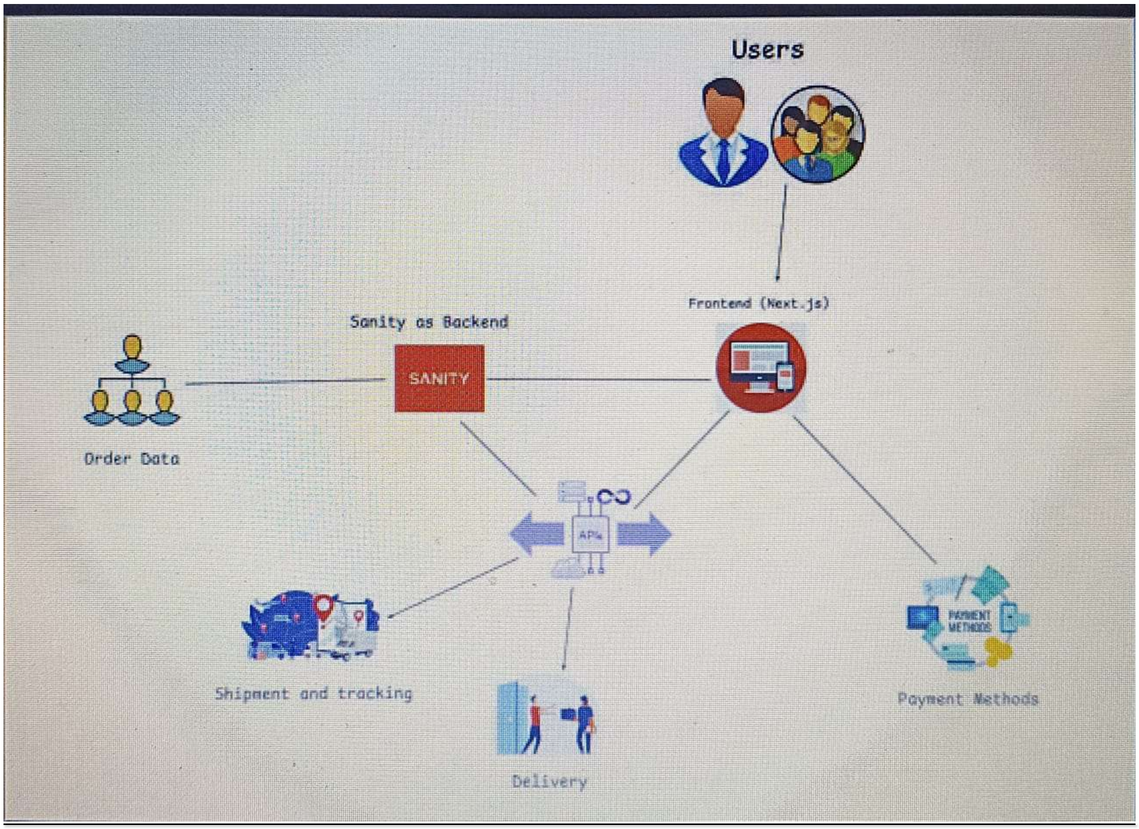
Method: Get

Description: fetch real-time delivery updates for persiable items,

Response Example :

{ “order-id” : 123, “status”: “in transit”,

“ETA”: “45 minutes”}



MARKETPLACE TECHNICAL FOUNDATION FOR MY E-COMMERCE WEBSITE

1 . system Architecture overview

This section provides an overview of how the different components of your e_commerce platform interact .

Frontend:

. responsive and user friendly design

. core pages

. Home: show featured product and categories

. Product Listing: display all product in category

. Cart and checkout: Add / remove products , view total prices and finalize orders

Backend;

. Technology : sanity CMS

. purpose : manaage all content and data

. products, order, customers, and shipment details

Third-party Api

. shipment : Track shipment assign riders and fetch delivery zone coverage

. payment getway : jazzcash/stripe for secure payment

. notification Api : Twilllio /firebase for SMS/email anout orders

product schema:

. ID; unique identify for all products

. name: name of the product

. price : price of the producy

. images: array of images url

order schema:

. order Id : unique identify for the order

. customer id : ID for the customers who placed the order

. products

. product Id: ID of the product

. Quantity : Quantity of the order

. status : current status of the order (eg. Pending shipped delivery)

. payment status : status of payment (eg. Paid unpaid)

. Delivery address: Address for the order delivery

Customer schema:

• customer Id : unique identifier for bthe customer

. name : name of the product

. contact info: contact information for customer (eg . phone email)

. purchase history : array of orders Ids representing the puirchase history

. Delivery ID : unique identify for the delivery

. order ID : Associated order ID

. Assigned Rider : name or id fore the rider responsible for delivery

. Actual Delivery : Time : Time stamp of when the delivery was completed

. delivery notes : optional notes about the delivery (eg . left at the dooostyrap)

Key backend feature :

1) Order management

. Handle orders creations , status , updates and collection

. payment integration for COD for online payment

. 2) inventory Management

. update stock level based on orders

. alters for lows stock level

3) Returns Refunds

. simplify returns with easy to use workflow

. automate refunds process

4) customers review

. enable uploading of photos ans real feedback

5) secuitry

. Authentications use JWT for login and sign in process

. Encryptions : HTTPS for secure communication

6) Deplyment and monitoring

1 hosting

. deploy frontend on vercel

. host backend (sanity CMS) on sanity.io

2) Monitoring Tools

. use logrockets or sentry for performance monitoring

. enable goggle Analyttics to track user enagment