Day 2 Assignment: Planning the Technical Foundation for the E-Commerce

Web
Overview:
The General E-CommerceWebsite will allow users to browse, Cart and manage Orders booked efficiently. The website should prioritize scalability, responsiveness, and ease of use.
Key Technical Requirements:
Frontend Framework: React.js (with Next.js for SSR and routing)
Styling: Tailwind CSS for a clean, responsive UI
Sanity: Sanity for storage and API
Authentication: Next auth for authentication (e.g., Google Login)
Payment Integration: Stripe for secure payments
Deployment Platform: Vercel for frontend
Version Control: Git (GitHub for repository hosting)
Design System Architecture
Frontend:
○ React components for modularity
○ Context API for state management
○ API calls to backend services for data retrieval
Backend:
○ RESTful APIs
○ Middleware for input validation

Submission Title

"Marketplace Technical Fundation E-Commerce"

This Documents describes the user journey flow for an eCommerce marketplace, Focusing on key steps and their technical implementation its is tailored for building arobust and user-freindly platform

Flowchart Overview.

Below is the complete user journey flow for an eCommerce marketplace.

Home Page.

User lands on the homepage.
Displays featured categories, Running products, and a search bar

Products Browsing.

User selects a category or uses the search bar.
Products are displayed with options to filter by price , Brand or rating and stock

Product Details:

User Clicks on cart to view uts details.
Page includes product desription price availability and user reviees

Add to Cart:

User add the product to cart
Cart updates dynamically with quantity and price

Checkout:

User proceds to the checkout page
Provides shipping address and selects a delivery option

Payments.

User enters payment details.
Secure payment gateway processes the transaction

Order Confirmation.

Orders details are displayed and sent via email
Order status is updates in the backend

Shipment Tracking.

User visits the (Order History) Section
Real-Time Shipment tracking is enable via API intergration

Delivery.

Product is delivered to the user address

Data Schema.

User Table:
- UserID (Type: Number:Required:True)
- Name (Type: String : Required :True)
- Email (Type: String/Number: Required :True)
- Password (hashed)
- Address (Type: String/ Boolean/Number : Required :True)

Products Table:
- ProductID (Type: Number : Required :True)
- ProductName (Type: String : Required :True)- ProductDescription (Type: String : Required :True)
- Price (Type: Number : Required :True)
- ImageURL
- CategoryID (foreign key referencing Categories table)

- Category Table*:
- CategoryID (Type: Number : Required :True)
- CategoryName (Type: String : Required :True)
- CategoryDescription (Type: String : Required :True)

- Orders Table:
- OrderID (Type: Number: Required :True)
- UserID (foreign key referencing Users table)
- OrderDate (Type: Number : Required :True)
- TotalCost (Type: Number : Required :True)
- Status (e.g., "pending", "shipped", "delivered")
- *Order Items Table*:
- OrderItemID (Type: Number : Required :True)
- OrderID (foreign key referencing Orders table)
- ProductID (Type: Number : Required :True)
- Quantity (Type: Number : Required :True)

- Payments Table:
- PaymentID (Type: Number : Required :True)
- OrderID (foreign key referencing Orders table)
- PaymentMethod (EasyPaisa.Jazzcash.Bank transfer)
- PaymentDate (Type: Number : Required :True)
- Status (e.g., "pending", "successful", "failed")