

1. Technical Roadmap

1. Requirement Analysis

- a. Define user journeys (browse, add to cart, order, track shipment, make payments).
- b. Identify core features: product listing, customization options, payment integration, order management.

2. Frontend Development

Tech Stack: Next.js with responsive design.

Pages:

- i. Home
- ii. Categories (Corporate Giveaways, Promotional Apparel, Customized Items)
- iii. Product Detail Page (e.g., Office Essential Kit)
- iv. Cart Manager
- v. Wishlist
- vi. Shipment/Tracking Order
- vii. About Us
- viii. Payment Methods
- ix. FAQs
- x. Privacy Policy
- xi. Contact Us (with a form)
- xii. Login/Signup

Features:

- xiii. Dynamic categories and new arrivals sections.
- xiv. Customer reviews section.

3. Backend Development

Tech Stack: Sanity CMS for content and data management.

Data Schemas:

Product Schema:

```
{
  "id": "01",
  "name": "Office Essential Kit",
  "description": "Includes keychain, bottle, diary, and box",
  "SKU": "t12",
  "price": 199,
  "category": "Corporate Giveaways",
  "customizable": true
}
```

Customer Schema:

```
{
  "id": "c01",
  "name": "John Doe",
  "contactInfo": {
    "email": "johndoe@example.com",
    "phone": "123-456-7890"
  },
  "orders": ["o01"]
}
```

Order Schema:

```
{
  "id": "o01",
  "customerId": "c01",
  "productId": "01",
  "shipmentId": "s01",
  "paymentStatus": "Paid",
  "totalPrice": 199
}
```

Shipment Schema:

```
{
  "id": "s01",
  "orderId": "o01",
  "zone": "Karachi",
  "deliveryTime": "2 Days",
  "courierDetails": "XYZ Courier"
}
```

4. API Development

a. Endpoints:

- i. GET /products: Fetch all products.
- ii. GET /products/:id: Fetch details of a specific product.
- iii. GET /categories: Fetch all categories.
- iv. POST /orders: Place an order.
- v. GET /orders/:id: Fetch order details.
- vi. POST /shipment: Create shipment details.
- vii. GET /shipment/:id: Fetch shipment status.

5. Integration of Third-Party APIs

- a. **Shipment Tracking API:** Integrate with a courier service for live tracking.

b. Payment Gateways:

- i. Bank transfer.
- ii. COD (Cash on Delivery).

6. Testing & Optimization

- a. Perform end-to-end testing for all user journeys.
- b. Optimize for speed and scalability.

7. Deployment

- a. Host the frontend on Vercel or Netlify.
- b. Deploy the backend and APIs on cloud platforms like AWS or DigitalOcean.

2. User Journey Workflow

1. Browsing Products:

User visits the homepage, navigates through categories, and views new arrivals and popular items. *n ****Product Selection:**

User views product details, selects options (e.g., customization), and adds to cart.

Cart Management:

User reviews items in the cart, applies discounts if any, and proceeds to checkout.

Order Placement:

User enters contact and shipping details, selects payment method, and confirms the order.

Shipment Tracking:

After confirmation, users receive a shipment ID to track delivery.

Order Completion:

Order status updates to "Delivered," and users can leave reviews.

3. Example Schema Definitions

a. Categories Schema:

```
[
  {
    "id": "cat01",
    "name": "Corporate Giveaways"
  },
  {
    "id": "cat02",
    "name": "Promotional Apparel"
  },
  {
    "id": "cat03",
    "name": "Customized Items"
  }
]
```

b. Customized Items Example Schema:

```
{
  "id": "citem01",
  "name": "Customized Giveaway Box",
  "items": ["bottle", "pen", "diary"],
  "price": 49.99
}
```

4. API Endpoints

Endpoint	Method	Description
/products	GET	Fetch all products
/products/:id	GET	Fetch product details
/categories	GET	Fetch all categories
/orders	POST	Place an order
/orders/:id	GET	Get order details
/shipment	POST	Create shipment details
/shipment/:id	GET	Get shipment tracking details

5. Technical Implementation Workflow

Frontend Implementation:

Build responsive UI components for all pages.

Integrate API endpoints for dynamic content.

Backend Setup:

Configure Sanity CMS for product and customer data.

Implement required schemas and relationships.

API Development:

Develop RESTful APIs for product, order, and shipment management.

Ensure secure and scalable API architecture.

Third-Party Integration:

Configure shipment tracking and payment gateway APIs.

Testing:

Validate all workflows from product browsing to order confirmation.

Deployment:

Deploy frontend and backend.

Monitor performance and resolve any issues post-launch.