# **Planning Technical Requirements**

Name: **Muskaan Fayyaz**

Roll No: **00216348  
 "Marketplace Technical Foundation (Comforty)”**

**Technical Requirements:**

* **Frontend:** Provide Best user-experience using Next.js and React for UI components**.**
* **Backend:** Sanity CMS for product, categories and order.
* **Database:** Sanity as the content source with structured schemas**.**
* **Third-Party Integrations:**
* Payment gateway
* shipment tracking API.

**Frontend Requirements:**

* **Home Page:** Display featured products or categories.
* **About Page:** Display the vision and mission with details of our website.
* **Shop Listing Page:** All the available products with filtering options.
* **Product Details Page:** Showcase product images, descriptions, prices, and a "Add to

Cart" button.

* **Cart Page:** It can Display the selected products with options to

update quantities or remove items.

* **Checkout Page:** Handle payment, order and information of customer and shipping.
* **Order Confirmation Page:** Confirm that the order has been placed and show order

Details.

**Backend (Sanity CMS):**

Using Sanity CMS to manage

* Product data
* Customer Detail
* Order Data
* Delivery Zone
* Shipment Data
* Payment Data

**API Requirements (Endpoints):**

* **Product API:**
* GET /api/products → Fetch all products
* GET /api/products/:id → Fetch product by ID
* POST /api/products → Add a new product (Admin only)
* PUT /api/products/:id → Update product details (Admin only)
* DELETE /api/products/:id → Remove a product (Admin only)
* **Order API**
* POST /api/orders → Create a new order
* GET /api/orders/:id → Fetch order details
* PUT /api/orders/:id/status → Update order status (Admin only)

#### **Shipment Tracking API**

* GET /api/shipment/:trackingNumber → Fetch shipment status.

**User Journey**

1. User browses products.
2. User adds items to the cart.
3. User proceeds to checkout.
4. Payment is processed through Stripe/PayPal.
5. Order is created and stored in Sanity.
6. Admin updates order status.
7. Shipment tracking is available.

**Admin Workflow**

1. Admin logs in.
2. Admin adds/updates products.
3. Admin manages orders.
4. Admin updates shipment statuses.

**Product Schema**

export default {

name: 'product',

title: 'Product',

type: 'document',

fields: [

{ name: 'id', type: 'number', title: 'ID' },

{ name: 'name', type: 'string', title: 'Name' },

{ name: 'description', type: 'text', title: 'Description' },

{ name: 'price', type: 'number', title: 'Price' },

{ name: 'category', type: 'string', title: 'Category' },

{ name: 'image', type: 'image', title: 'Image' }

]

};

**Order Schema**

export default {

name: 'order',

title: 'Order',

type: 'document',

fields: [

{ name: 'orderId', type: 'number', title: 'Order ID' },

{ name: 'customerId', type: 'number', title: 'Customer ID' },

{ name: 'product', type: 'array', of: [{ type: 'reference', to: [{ type: 'product' }] }], title: 'Product' },

{ name: 'totalAmount', type: 'number', title: 'Total Amount' }

]

};

**Customer Schema**

export default {

name: 'customer',

title: 'Customer',

type: 'document',

fields: [

{ name: 'customerId', type: 'number', title: 'Customer ID' },

{ name: 'name', type: 'string', title: 'Name' },

{ name: 'email', type: 'string', title: 'Email' },

{ name: 'address', type: 'string', title: 'Address' },

{ name: 'phoneNo', type: 'string', title: 'Phone Number' },

{ name: 'order', type: 'array', of: [{ type: 'reference', to: [{ type: 'order' }] }], title: 'Orders' }

]

};

