

Marketplace Technical Foundation

Mens Apparel

Introduction

Our e-commerce platform aims to provide exclusive collection of men's clothing, designed for every occasion. From casual wear to sharp formal looks, we've got you covered. This document outlines the technical foundation required to build a scalable, user-friendly marketplace.

Define Technical Requirements:

Frontend Requirements:

1. **User-friendly Interface**
2. **Responsive Design** - Ensure seamless navigation and browsing on mobile, tablet, and desktop devices.
3. **Essential Pages:**

Home: Showcase featured collections, seasonal trends, and a prominent search bar. Include sections for Latest Collection, Top Categories, Featured Products, and Trending Products.

About: Highlight the brand story, mission, values, and what sets your fashion business apart.

Contact Us: Provide a contact form for customer inquiries (fields for name, email, and message). Include phone numbers, email addresses, and a physical address (if applicable).

Product Listing: Offer filtering options (e.g., buy price).

Product Details: Highlight product features such as size, colors variations. Provide customer reviews and ratings.

Cart: Include an estimated shipping cost calculator.

Checkout: Enable guest checkout and account creation options. Include secure payment fields and shipping address validation.

Order Confirmation: Show order summary with delivery timeframes and tracking details.

Sanity CMS as Backend:

Sanity CMS for managing product data, orders, and customer details.

Here's a detailed schema design for **Product**, **Order**, **Customer**, **Payment**, **Shipment**, and **Delivery Zone** using Sanity CMS:

[Product]

```
export interface Product {  
  _id: string;  
  name: string;  
  description: string;  
  price: number;  
  images: { _type: 'image'; asset: { _ref: string; _type: 'reference' } }[];  
  stock: number;  
  category: string;  
  reviews: number  
}
```

[Order]

```
export interface Order {  
  _id: string;  
  productID: string;  
  Qty number  
  Amount: number;  
  orderDate: string  
}
```

[Customer]

```
export interface Customer {  
  _id: string;  
  name: string;  
  email: string;  
  phone: string;  
  address: string;  
}
```

[Payment]

```
export interface Payment {  
  _id: string;  
  order: Order;  
  paymentMethod: 'Credit Card' | 'easypaisa' | 'JazzCash' ;  
  status: 'Pending' | 'Completed' | 'Failed';  
  transactionId: string;  
  Amount: number;  
  paymentDate: string;  
}
```

[Shipment]

```
export interface Shipment {  
  _id: string;  
  order: Order;  
  trackingNumber: string;  
  status: 'Pending' | 'In Transit' | 'Delivered' | 'Cancelled';  
  estimatedDelivery: string;
```

```
}
```

[Delivery Zone]

```
export interface DeliveryZone {  
  _id: string;  
  zoneName: string;  
  Area: string[];  
  shippingCost: number;  
  Driver: string;  
}
```

Third-Party APIs:

Shipment Tracking:

Fetch and display order shipment status in real time.

Payment Gateways:

Securely process payments and return confirmation to the system.

Product Browsing and order Placement

