

# Marketplace Technical Foundation - Komal's Shoes E-commerce Store

## Milestone 1 - Project Setup & Frontend Development

- Initialize Next.js 14 with Tailwind CSS.
  - Create dynamic routes for:
    - /male, /female, /kids /product/[id], /cart, /checkout
  - Homepage (/)
    - Display products categorized by Male, Female, and Kids.
    - Implement a product search feature.
  - Product Detail Page (/product/[id])
    - Show product image, name, price, stock status, and size options.
    - Implement an Add to Cart button.
  - Cart Page (/cart)
    - Display selected products with quantity and total price.
    - Include a Proceed to Checkout button.
  - Checkout Page (/checkout)
    - Collect customer details and address.
    - Display order summary.
- 

## Milestone 2 - Backend Development & API Integration

### Sanity CMS Setup

- Product Management
  - Store product details (name, price, stock, category, image, size).
  - Fetch product data using GROQ queries.
- Order Management

- Store customer orders, including:
    - Order ID, Customer Name, Email, Address.
    - Products Ordered (IDs & Quantities).
    - Payment Status (Pending by default).
- 

## Milestone 3 - API Specification Document

### Products API

#### Fetch All Products

- **Endpoint:** /products
- **Method:** GET
- **Description:** Fetch all available products from Sanity.
- **Response Example:**

```
[  
  {  
    "id": 1,  
    "name": "Product 1",  
    "price": 120,  
    "stock": 60,  
    "category": "Female",  
    "size": "XL",  
    "image": "https://example.com/images/product1.jpg"  
  }  
]
```

### Orders API

#### Create Order

- **Endpoint:** /orders
- **Method:** POST

- **Description:** Create a new order in Sanity.
- **Request Payload:**

```
{
  "customer": {
    "order_Id": 2544,
    "name": "Daniel",
    "email": "dani@example.com",
    "address": "ABC road",
    "timestamp": "5 days"
  },
  "products_Detail": [
    { "productId": 1, "quantity": 2 },
    { "productId": 2, "quantity": 1 }
  ],
  "paymentStatus": "pending"
}
```

- **Response Example:**

```
{
  "orderId": 2541,
  "totalPrice": 2540,
  "timestamp": "6 days",
  "status": "pending"
}
```

## Shipment API

### Track Order Status

- **Endpoint:** /shipment
- **Method:** GET
- **Description:** Track order status via third-party API.
- **Request Parameters:** orderId
- **Response Example:**

```
{
```

```
"shipmentId": "ABC123",  
"orderId": 1234,  
"status": "Shipped",  
"expectedDeliveryDate": "2025-01-20"  
}
```

## Payment API

### Process Payment

- **Endpoint:** /payment
- **Method:** POST
- **Description:** Process payment for an order.
- **Request Payload:**

```
{  
  "orderId": 1234,  
  "amount": 280,  
  "paymentMethod": "Credit Card",  
  "cardDetails": {  
    "cardNumber": "4521158741",  
    "expiryDate": "19/26",  
    "cvv": "123"  
  }  
}
```

- **Response Example:**

```
{  
  "paymentId": "PAY12345",  
  "status": "Success",  
  "transactionDate": "2025-01-16",  
  "amount": 280  
}
```

---

## Milestone 4 - Order Processing & Payment

- **Order Placement**
    - Allow users to place orders from the frontend.
    - Send order data to **Sanity CMS**.
  - **Shipment Tracking**
    - Fetch real-time order status using a **third-party shipment API**.
  - **Payment Processing**
    - Process payments via **API** (e.g., Credit Card, Cash on Delivery).
    - Update **payment status** in Sanity.
- 

## Milestone 5 - Deployment & Optimization

- Optimize product images using Next.js Image Optimization.
  - Improve performance with Lazy Loading & Caching.
  - Deploy on Vercel.
-