



API Specification Document

1. Fetch All Products

Endpoint: `/products`
Method: GET
Description: Fetch all available products from Sanity CMS.
Request Headers: `Content-Type: application/json`

```
{
  "id": 1,
  "name": "Product A",
  "price": 100,
  "stock": 50,
  "image": "https://example.com/images/product-a.jpg"
},
{
  "id": 2,
  "name": "Product B",
  "price": 80,
  "stock": 10,
  "image": "https://example.com/images/product-b.jpg"
}
```

2. Create a New Order

Endpoint: `/orders`
Method: POST
Description: Create a new order in Sanity CMS.
Request Headers: `Content-Type: application/json`

```
{
  "customer": {
    "name": "John Doe",
    "email": "john.doe@example.com",
    "phone": "1234567890"
  },
  "products": [
    {
      "id": 1,
      "quantity": 2
    },
    {
      "id": 2,
      "quantity": 1
    }
  ],
  "paymentStatus": "Paid"
}
```

Response Example:

```
{
  "orderId": "ORD12345",
  "message": "Order created successfully",
  "totalAmount": 300,
  "status": "Pending"
}
```

3. Track Shipment Status

Endpoint: `/shipment`
Method: GET
Description: Track order status via a third-party API.
Request Headers: `Content-Type: application/json`
Query Parameters: `shipmentId` (Shipment ID to track the order).

```
{
  "shipmentId": "SHIP12345",
  "orderId": "ORD12345",
  "status": "In Transit",
  "expectedDelivery": "2025-01-20"
}
```

Sanity Schema Design

Sanity CMS Schema Definition

1. Product Schema

Fields:

Field Name	Type	Description
<code>id</code>	String	Unique identifier for each product.
<code>name</code>	String	Name of the product.
<code>price</code>	Number	Price of the product.
<code>stock</code>	Number	Current stock quantity.
<code>image</code>	String	URL of the product image.

Example Document:

```
{
  "id": "prod-001",
  "name": "Wireless Headphones",
  "price": 100,
  "stock": 50,
  "image": "https://example.com/images/headphones.jpg"
}
```

2. Order Schema

Fields:

Field Name	Type	Description
<code>id</code>	String	Unique identifier for each order.
<code>customer</code>	Object	Customer information (name, email, phone).
<code>products</code>	Array	Array of product objects with quantity.
<code>totalAmount</code>	Number	Total amount of the order.
<code>status</code>	String	Current status of the order.

Example Document:

```
{
  "id": "order-001",
  "customer": {
    "name": "John Doe",
    "email": "john.doe@example.com",
    "phone": "1234567890"
  },
  "products": [
    {
      "id": "prod-001",
      "quantity": 2
    },
    {
      "id": "prod-002",
      "quantity": 1
    }
  ],
  "totalAmount": 300,
  "status": "Pending"
}
```

3. Shipment Schema

Fields:

Field Name	Type	Description
<code>id</code>	String	Unique identifier for each shipment.
<code>orderId</code>	String	Reference to the order ID.
<code>status</code>	String	Current status of the shipment.
<code>expectedDelivery</code>	String	Expected delivery date.

Example Document:

```
{
  "id": "ship-001",
  "orderId": "order-001",
  "status": "In Transit",
  "expectedDelivery": "2025-01-20"
}
```

