**Documentation**

**Aim :**

Design and implement RESTful APIs for a simple Product Management Module, using which the system can manage (create, read, update, and delete) the following entities.

**Objective :**

1. **Product Management :**

Add new products with details such as name, price, stock quantity, and category.

View a list of all products with relevant information.

1. **CRUD Operations:**

Enable users to perform CRUD (Create, Read, Update, Delete) operations on products.

Add/Edit product details through a modal dialog.

Delete products from the list.

1. **Order Management (in the extended version):**

Implement a section for managing orders.

View a list of orders, their details, and items associated with each order.

Allow CRUD operations on orders.

1. **Error Handling:**

Error Handling Implement error handling mechanisms to gracefully handle issues such as network errors or API failures.

1. **State Management:**

Utilize state management libraries or patterns (like Redux) for efficient data handling and communication between components.

**Business Needs :**

1. **Efficient Product Management:**

Manage diverse product details accurately.

1. **Improved User Experience:**

Enhance satisfaction with a user-friendly interface.

1. **Effective Search and Filtering:**

Facilitate quick product search and filtering.

1. **Integration with Backend Systems:**

Ensure seamless communication between frontend and backend.

**Layout Forms :**

1. **Home Page**

Logo Home Product Category Order OrderItem CartList CatSearch OrderSearch

**Product Management App**

Footer

1. **Product Page**

Logo Home Product Category Order OrderItem CartList CatSearch OrderSearch

Search Bar For Product

Add Product

ProductId ProductName Price StockQuantity Category Action AddToCart

Update/Delete AddToCArt

Footer

1. **Category Page**

Logo Home Product Category Order OrderItem CartList CatSearch OrderSearch

Category List

Add Category

CategoryId CategoryName Action

Update/Delete

Footer

1. **Order Page**

Logo Home Product Category Order OrderItem CartList CatSearch OrderSearch

Add Order

OrderId CategoryDate Total Amount Action

Update/Delete

Footer

1. **OrderItem Page**

Logo Home Product Category Order OrderItem CartList CatSearch OrderSearch

Create Order

Id orderid OrderDate TAmount ProdId ProdName Quantity UnitPrice Action

Update/Delete

Footer

1. **CartList Page**

Logo Home Product Category Order OrderItem CartList CatSearch OrderSearch

Cart

Id ProductName Price Stock Quantity Action

Update/Delete

Footer

1. **CategorySearchByCartId Page**

Logo Home Product Category Order OrderItem CartList CatSearch OrderSearch

Enter Cart ID

Footer

1. **ProductSearchByOrderId Page**

Logo Home Product Category Order OrderItem CartList CatSearch OrderSearch

Enter Order ID

Footer

**Real Time Events :**

1. **Product Creation:**

Event: Triggered when a new product is added.

Use Case: Updates the product list in real-time for all users.

1. **Product Update:**

Event: Triggered when there is a modification to product details.

Use Case: Ensures that updated product information is immediately reflected throughout the system.

1. **Product Deletion:**

Event: Triggered when a product is removed from the inventory.

Use Case: Updates the system to reflect the removal of the product in real-time.

1. **Search and Filter:**

Event: Triggered when a user performs a search or applies filters.

Use Case: Dynamically updates the displayed products based on the search criteria.

1. **Order Placement:**

Event: Triggered when a new order is placed.

Use Case: Updates stock levels and order history in real-time.

1. **Inventory Update:**

Event: Triggered when there is a change in stock quantity.

Use Case: Ensures that stock levels are synchronized across the system.

1. **Error Handling:**

Event: Triggered when an error occurs during product management operations.

Use Case: Provides real-time notifications for users and logs errors for system

administrators.

1. **Price Changes:**

Event: Triggered when there is a change in product pricing.

Use Case: Updates displayed prices for users and maintains accurate financial records.

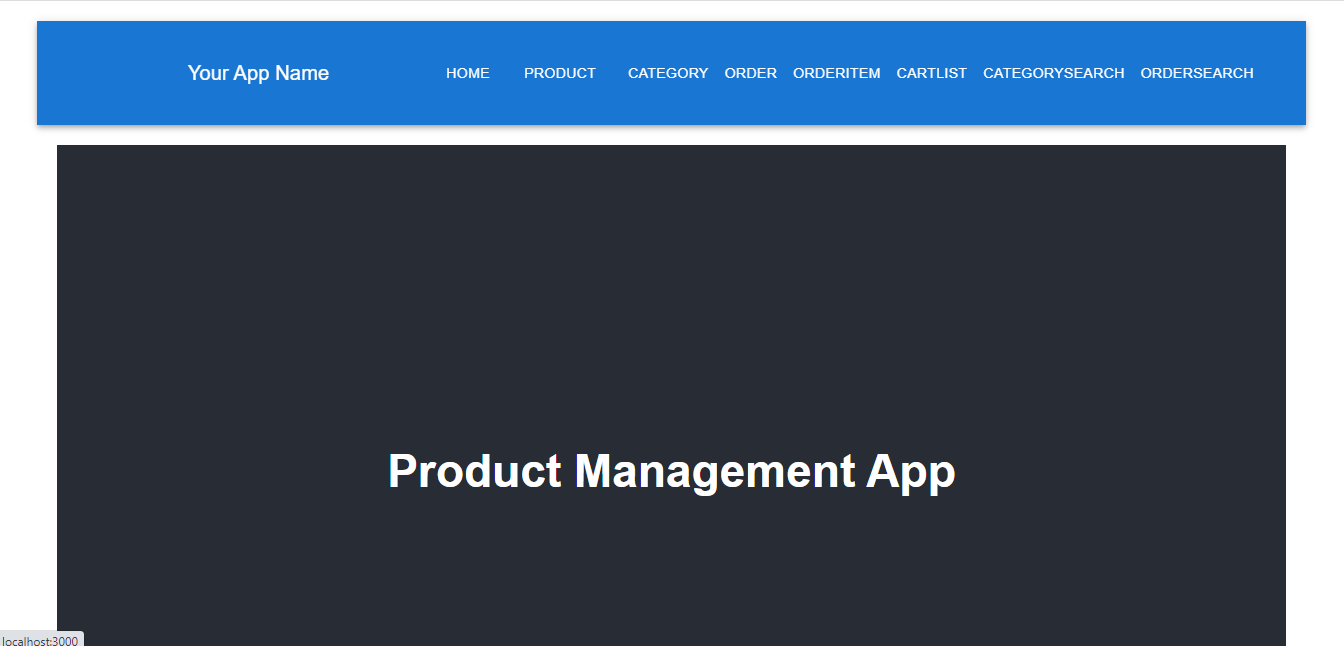
1. **Integration with External Systems:**

Event: Triggered during data exchanges with external systems (e.g., payment gateways, shipping services).

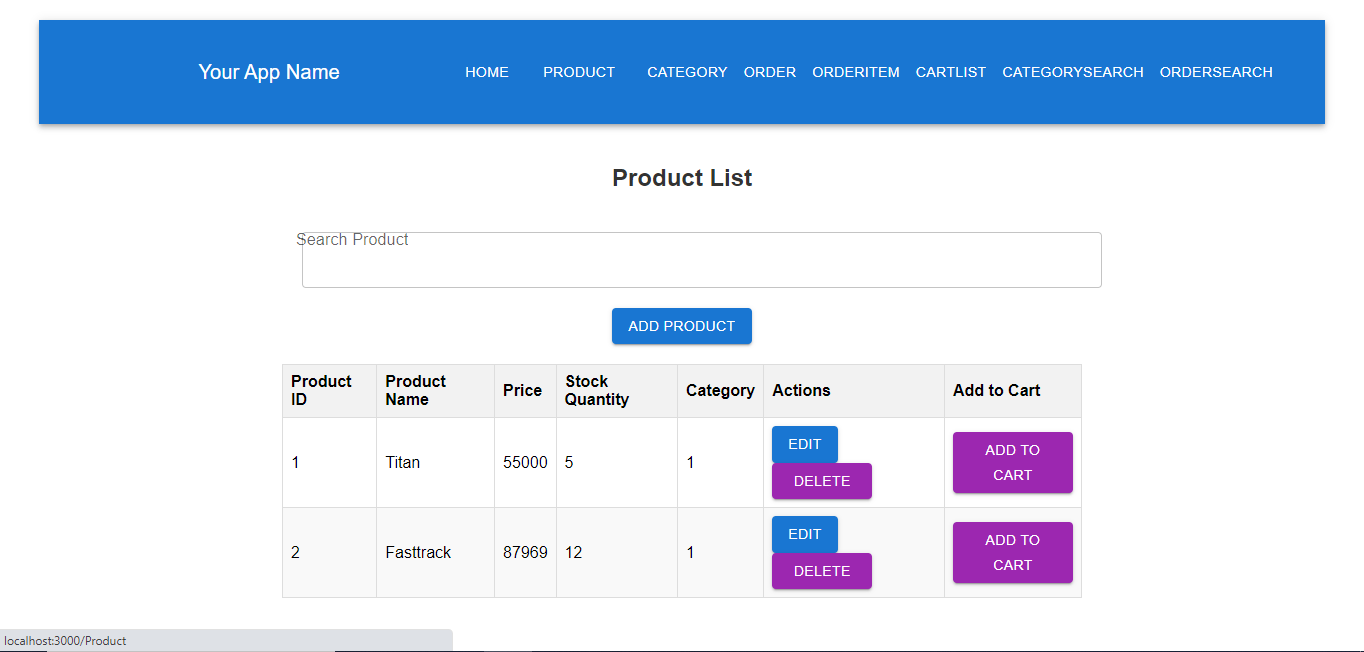
Use Case: Ensures seamless communication and synchronization with external partners.

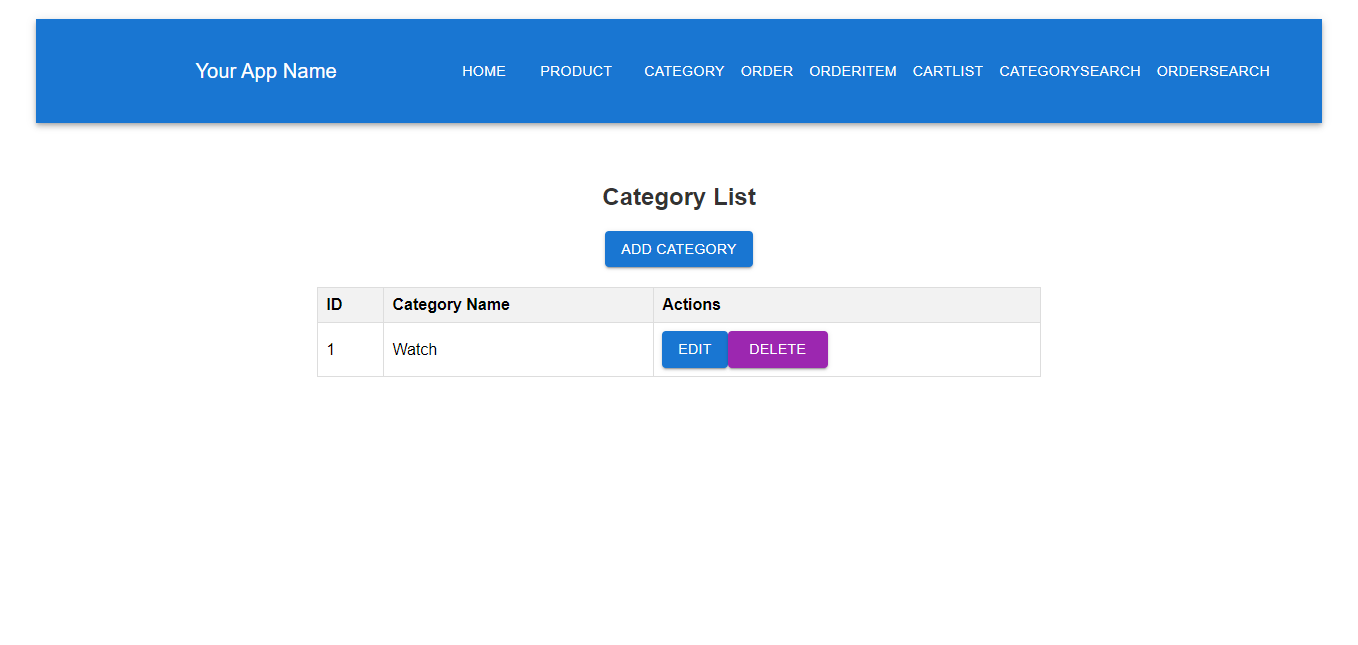
**Output :**

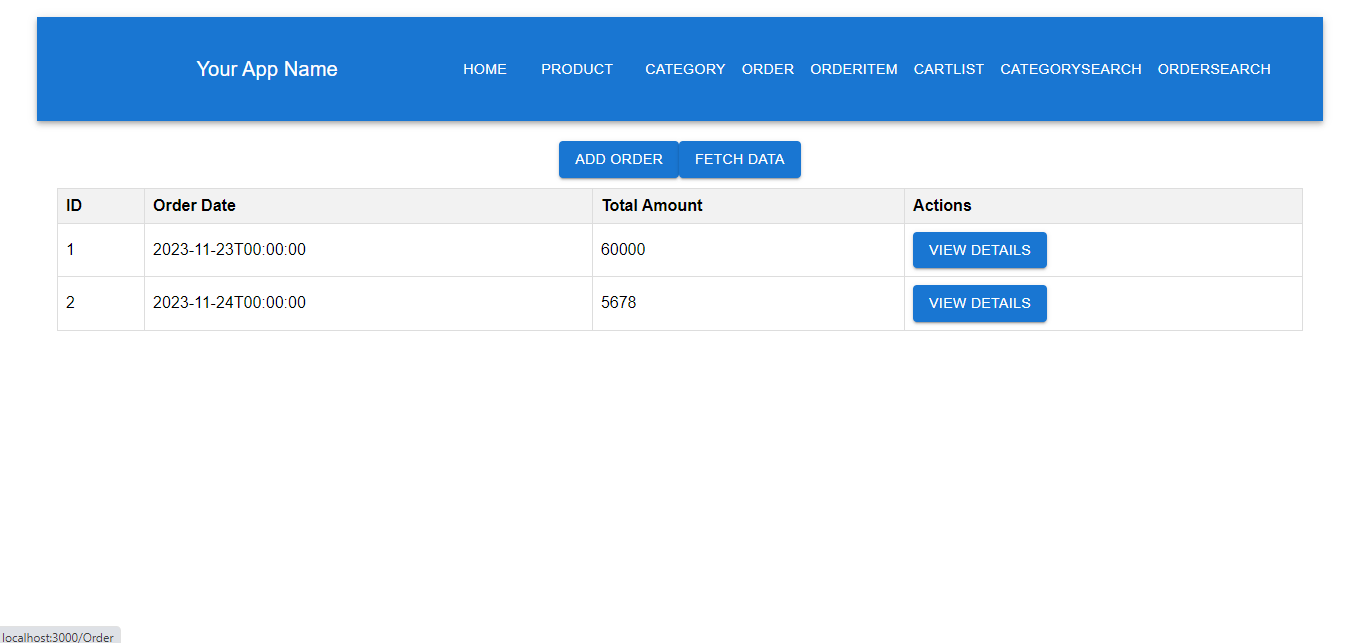
1. **HomePage**

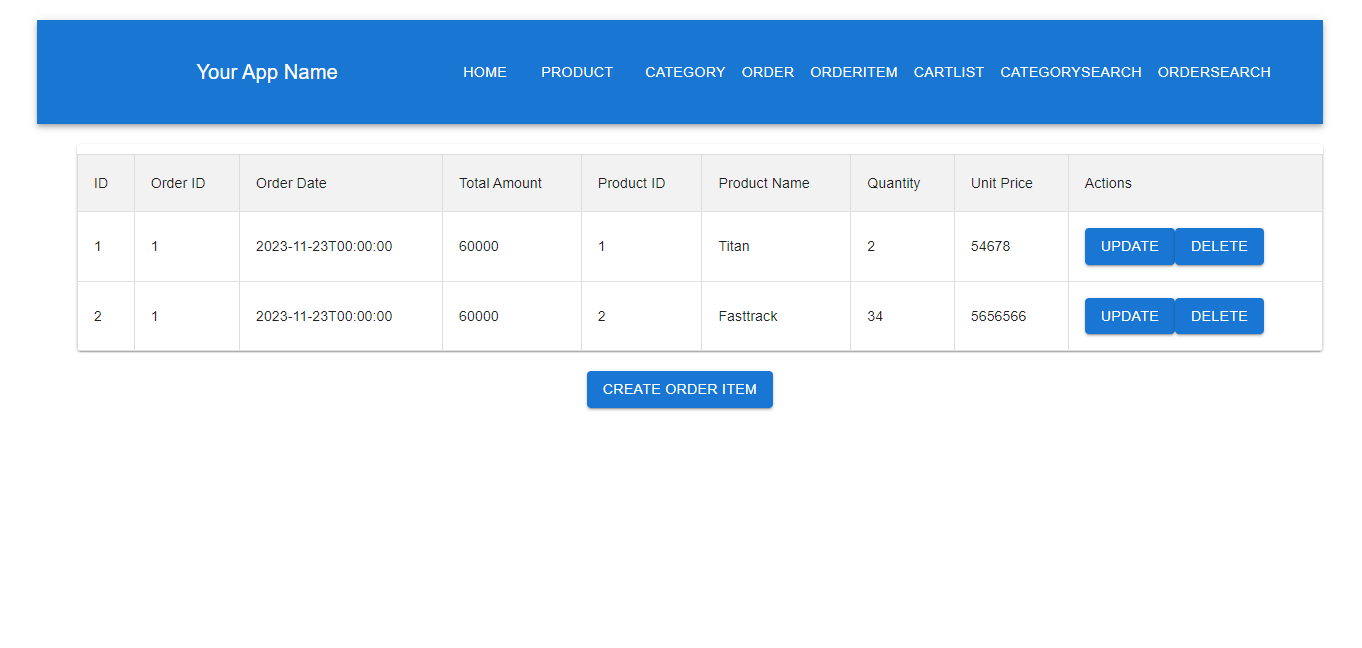
****

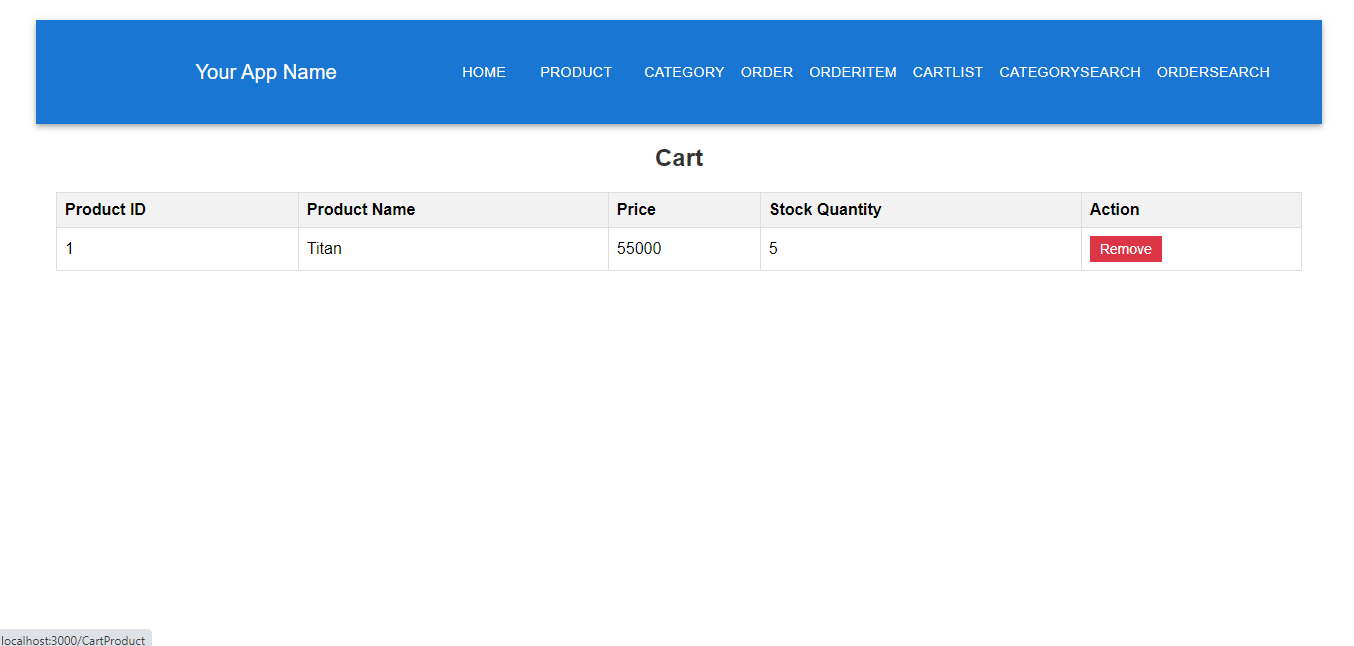
1. **Product Page**

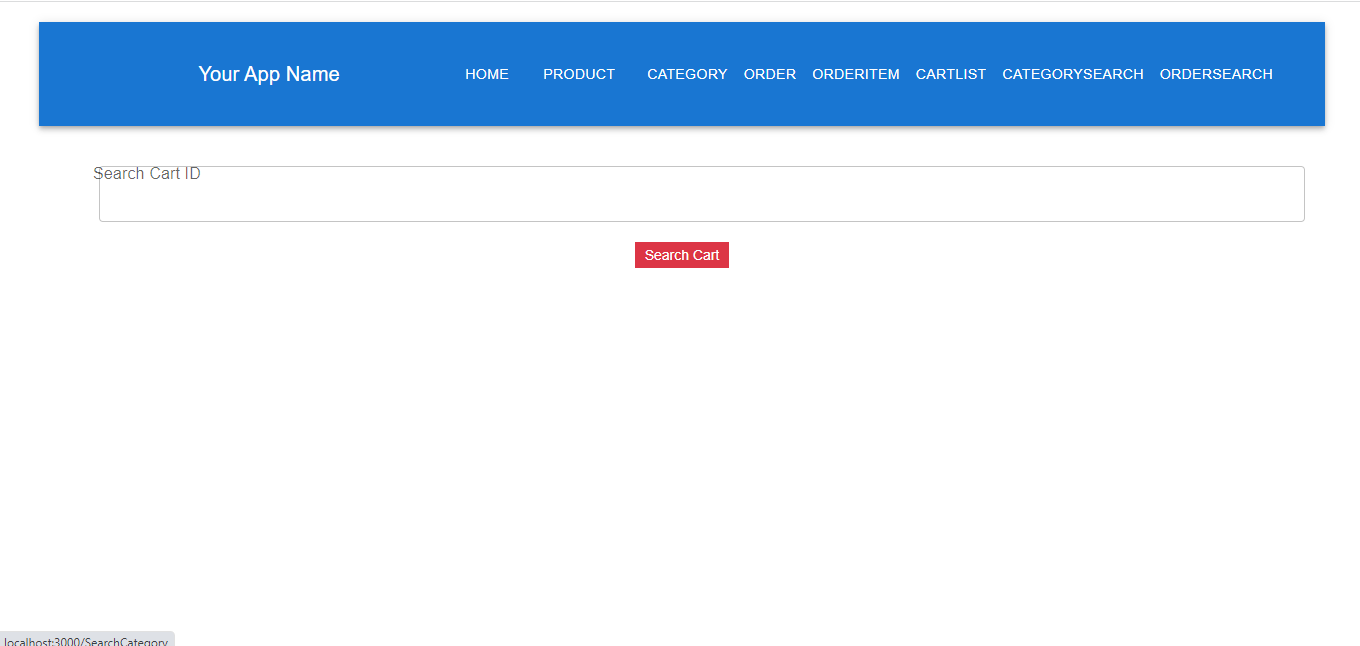
****

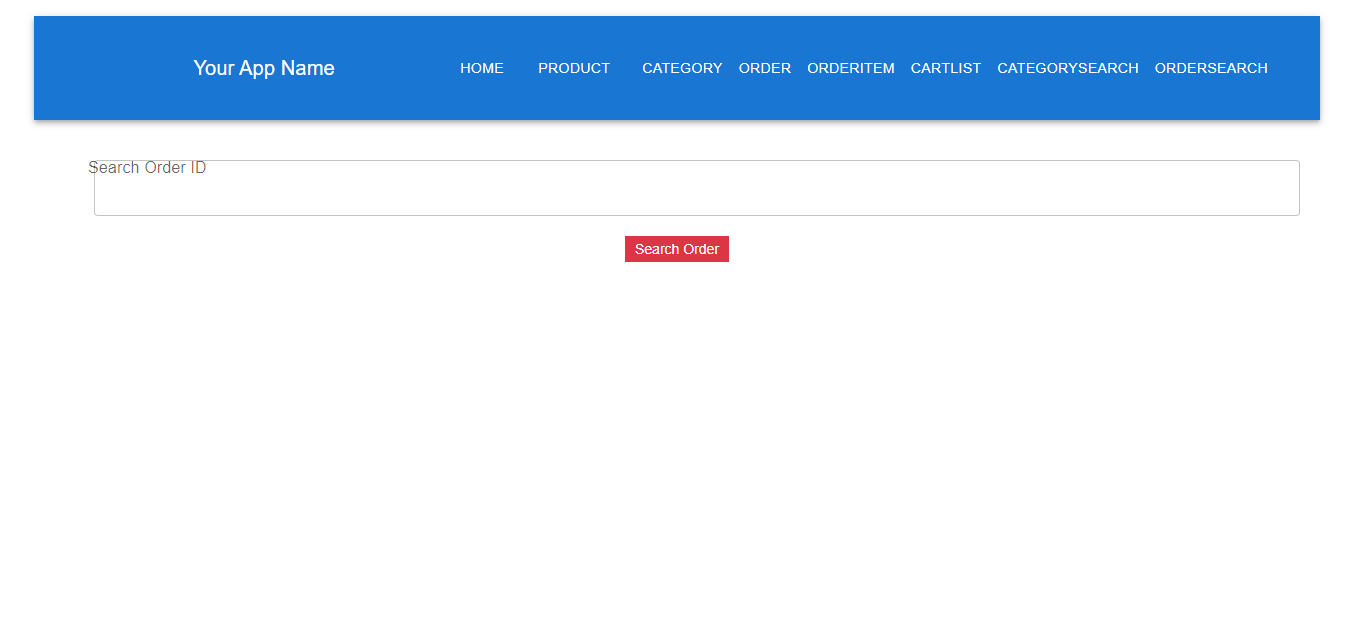
1. **Category Page**

1. **Order Page**
2. **OrderItem**

****

1. **CartList**
2. **CategorySearchByCartId Page**

****

1. **ProductSearchByOrderId Page**