

## **Project Overview:**

The ASP.NET MVC Machine Test project consists of two main controllers: Category and Product controller.

1. **Category Master:** This allows users to create, read, update, and delete categories. Each category can have multiple products associated with it.
2. **Product Master:** This includes CRUD operations for products, with server-side pagination implemented to efficiently manage and display large datasets. Each product is linked to a category, ensuring a structured organization of items.

## **1. Project Setup**

- **Framework:** The project is built using ASP.NET MVC, which follows the Model-ViewController architectural pattern.
- **Database:** A SQL Server database is used to store data for categories and products.

## **2. Database Design**

- **Categories Table:**
  - **CategoryId (Primary Key)**
  - **CategoryName (String)**
- **Products Table:**
  - **ProductId (Primary Key)**
  - **ProductName (String)**
  - **CategoryId (Foreign Key referencing Categories)**

## **3. Category Controller • CRUD Operations:**

- **Create:** A form to add new categories.
- **Read:** A list view displaying all categories.
- **Update:** An edit form to modify existing categories.
- **Delete:** A confirmation dialog to remove categories.
- **Implementation:** Each operation is handled by corresponding controller actions and views.

#### 4. Product Controller

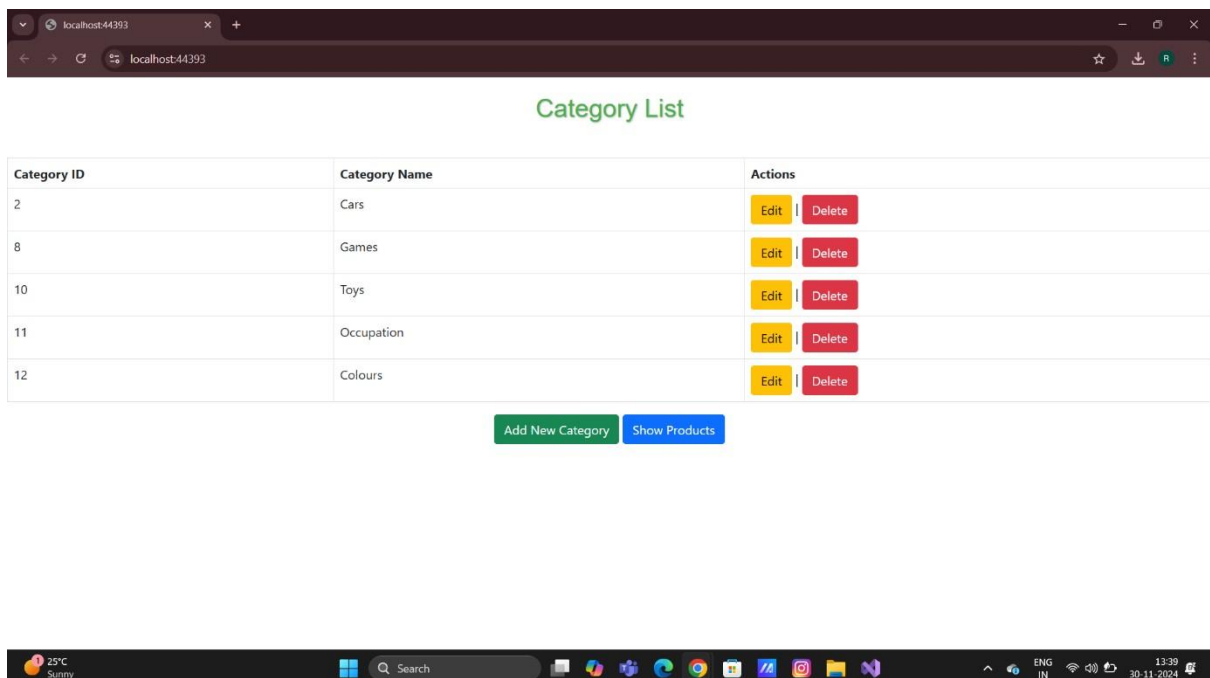
- **CRUD Operations:**
- **Create:** A form to add new products, including a dropdown to select the category.
- **Read:** A list view displaying all products with their associated categories.
- **Update:** An edit form to modify existing products.
- **Delete:** A confirmation dialog to remove products.
- **Implementation:** Similar to the Category Master, each operation is managed by controller actions and views.

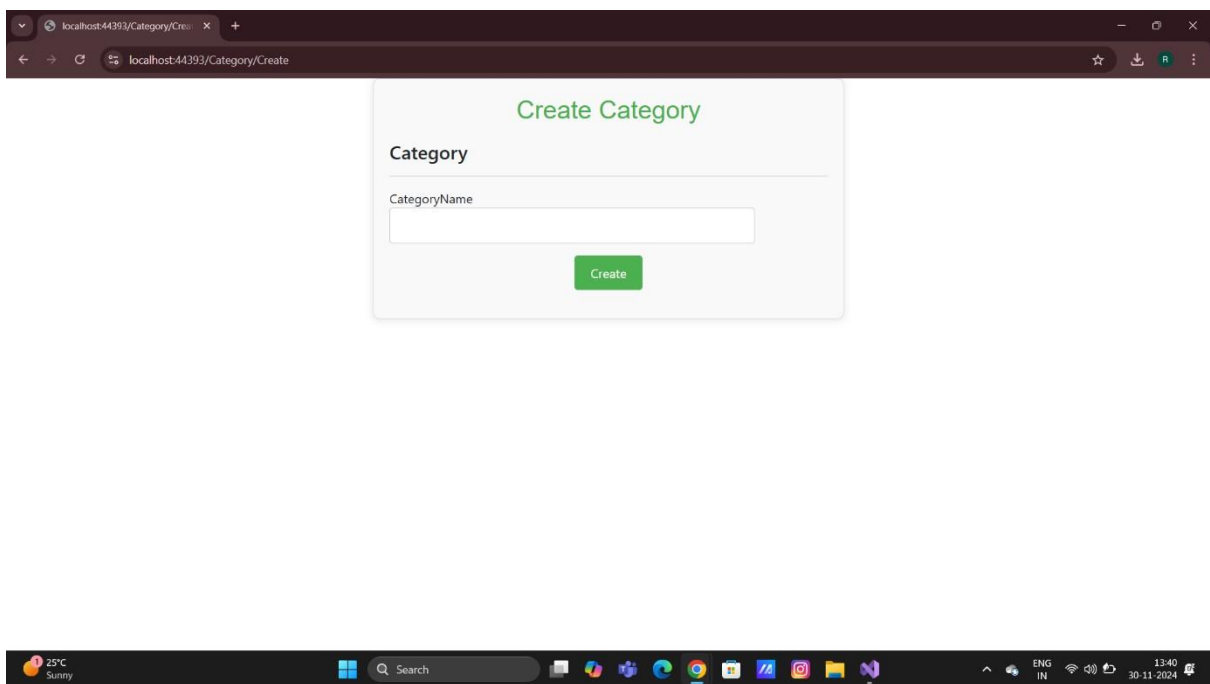
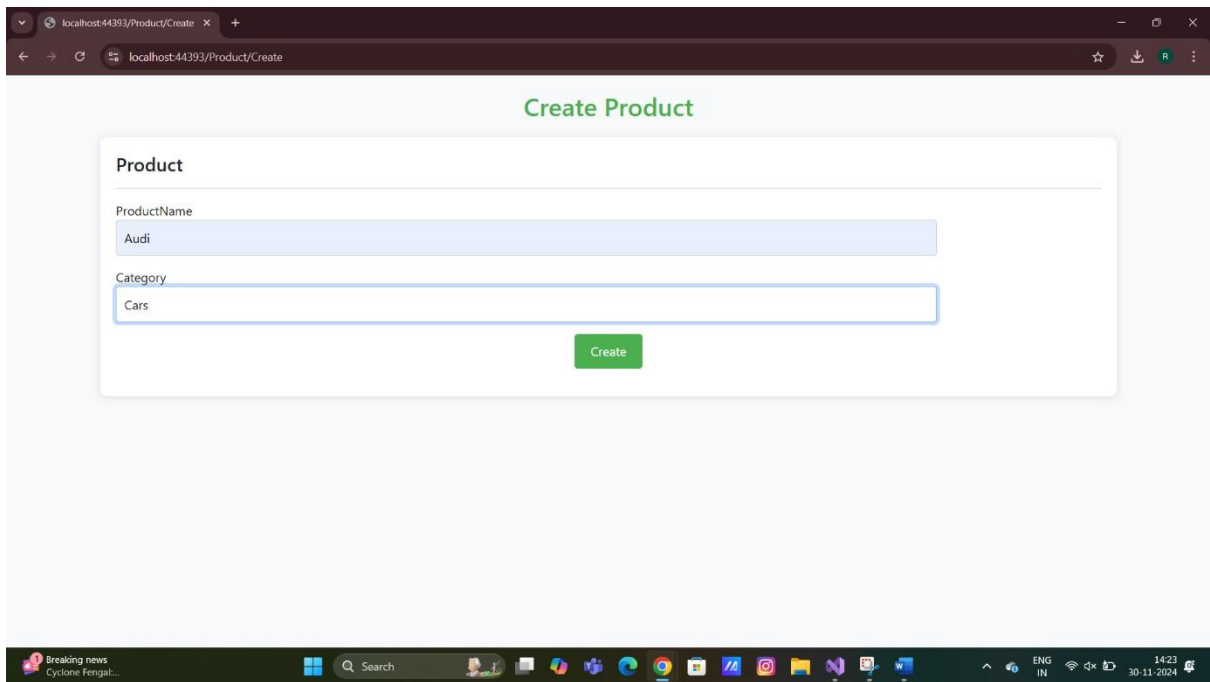
#### How I Ran the Machine Test:

I followed the requirements provided, implementing the Category and Product masters with CRUD operations and server-side pagination for the product list.

Category List – Here you can add category and also display Products .CRUD operations are performed.

Output Images -





localhost:44393/Product

localhost:44393/Product

### Product List

Productid	ProductName	Categoryid	CategoryName	Action
1	Audi	2	Cars	<a href="#">Edit</a>   <a href="#">Delete</a>
5	Pubg	8	Games	<a href="#">Edit</a>   <a href="#">Delete</a>
6	Alto	2	Cars	<a href="#">Edit</a>   <a href="#">Delete</a>
7	Porsche	2	Cars	<a href="#">Edit</a>   <a href="#">Delete</a>
8	BMW	2	Cars	<a href="#">Edit</a>   <a href="#">Delete</a>
9	Free Fire	8	Games	<a href="#">Edit</a>   <a href="#">Delete</a>
10	Agriculture	11	Occupation	<a href="#">Edit</a>   <a href="#">Delete</a>
11	Bussiness	11	Occupation	<a href="#">Edit</a>   <a href="#">Delete</a>
12	Temple Run	8	Games	<a href="#">Edit</a>   <a href="#">Delete</a>
13	Mini Milita	8	Games	<a href="#">Edit</a>   <a href="#">Delete</a>

Breaking news  
Cyclone Fengat...

Search

ENG  
IN

13:41  
30-11-2024

localhost:44393/Product

localhost:44393/Product

1	Audi	2	Cars	<a href="#">Edit</a>   <a href="#">Delete</a>
5	Pubg	8	Games	<a href="#">Edit</a>   <a href="#">Delete</a>
8	BMW	2	Cars	<a href="#">Edit</a>   <a href="#">Delete</a>
10	Agriculture	11	Occupation	<a href="#">Edit</a>   <a href="#">Delete</a>
11	Bussiness	11	Occupation	<a href="#">Edit</a>   <a href="#">Delete</a>
12	Temple Run	8	Games	<a href="#">Edit</a>   <a href="#">Delete</a>
13	Mini Milita	8	Games	<a href="#">Edit</a>   <a href="#">Delete</a>
14	Cricket18	8	Games	<a href="#">Edit</a>   <a href="#">Delete</a>
16	Swift	2	Cars	<a href="#">Edit</a>   <a href="#">Delete</a>
17	Real-Estate	11	Occupation	<a href="#">Edit</a>   <a href="#">Delete</a>

[Add New Product](#) [Back To Categories](#)

Breaking news  
Cyclone Fengat...

Search

ENG  
IN

13:42  
30-11-2024

**Product List**

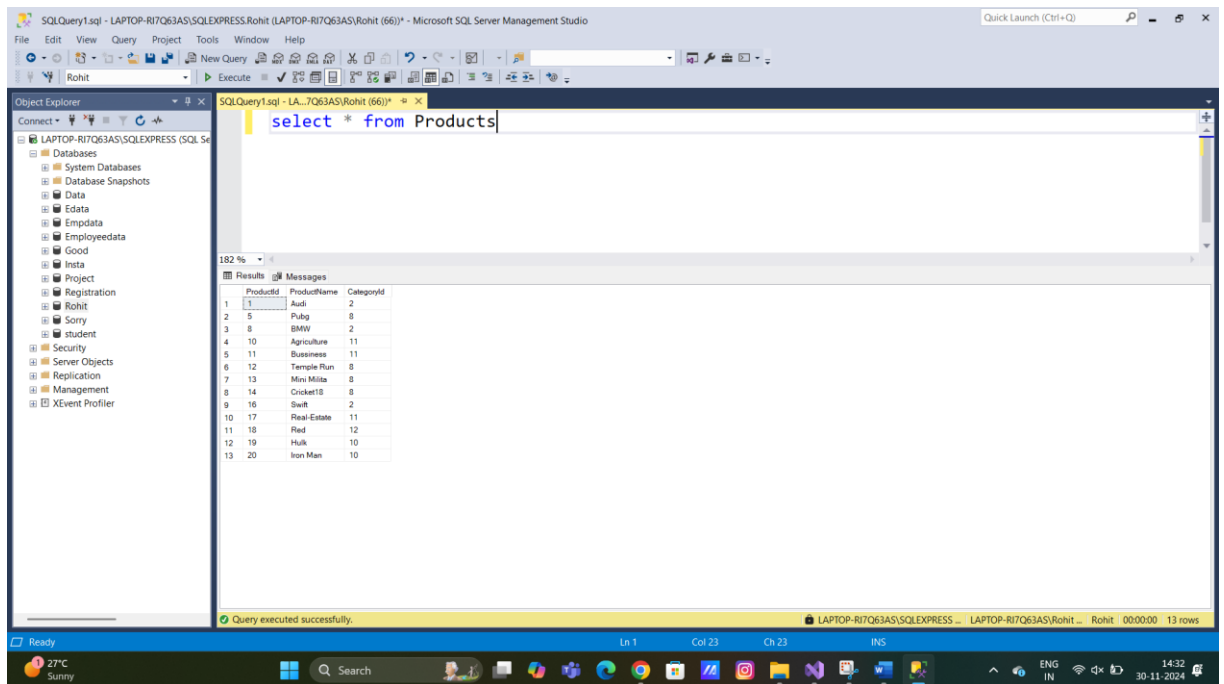
ProductId	ProductName	CategoryId	CategoryName	Action
1	Audi	2	Cars	<a href="#">Edit</a>   <a href="#">Delete</a>
5	Pubg	8	Games	<a href="#">Edit</a>   <a href="#">Delete</a>
8	BMW	2	Cars	<a href="#">Edit</a>   <a href="#">Delete</a>
10	Agriculture	11	Occupation	<a href="#">Edit</a>   <a href="#">Delete</a>
11	Bussiness	11	Occupation	<a href="#">Edit</a>   <a href="#">Delete</a>
12	Temple Run	8	Games	<a href="#">Edit</a>   <a href="#">Delete</a>
13	Mini Milita	8	Games	<a href="#">Edit</a>   <a href="#">Delete</a>
14	Cricket18	8	Games	<a href="#">Edit</a>   <a href="#">Delete</a>
16	Swift	2	Cars	<a href="#">Edit</a>   <a href="#">Delete</a>
17	Real-Estate	11	Occupation	<a href="#">Edit</a>   <a href="#">Delete</a>

## 5. Pagination Implementation

- **Server-Side Pagination:**
- The product list view implements server-side pagination to handle large datasets efficiently.
- **How It Works:**
- The controller fetches a specific number of products per page (e.g., 10 products).
- The total number of products is calculated to determine the total number of pages.
- The view displays the current page of products along with pagination controls (Next, Previous, Page Numbers).



## Product Table –



## Conclusion

This project demonstrates my ability to create a structured web application using ASP.NET MVC, implement CRUD operations, and efficiently manage data display through pagination. If you have any questions or need further clarification, please feel free to ask.

Thank you for the opportunity!

Best regards,

Rohit Shinde +91

9890899104

shinderohit0405@gmail.com