# View Component

Mục Lục

[View Component 1](#_Toc172903843)

[I. Tạo View Component danh sách Category 2](#_Toc172903844)

[II. Tạo View Component Featured Products 9](#_Toc172903845)

[III. Tạo View Component Navbar 14](#_Toc172903846)

## Tạo View Component danh sách Category

Trang Cửa hàng (**Product/index.cshtml**)

Tạo Model mới: **CategoryMenu.cs**

**Models/CategoryMenu.cs**

using System.ComponentModel.DataAnnotations;

namespace ShopManager.Models

{

public class CategoryMenu

{

public int Id { get; set; }

public string Title { get; set; }

public string Content { get; set; }

public int Count { get; set; }

}

}

Tại folder DAL, tạo class mới: **CategoryDAL.cs**

**DAL/CategoryDAL.cs**

using Microsoft.Data.SqlClient;

using ShopManager.Database;

using ShopManager.Models;

namespace ShopManager.DAL

{

public class CategoryDAL

{

DBConnect connect = new DBConnect();

public List<CategoryMenu> getAllWithCount()

{

connect.openConnection();

List<CategoryMenu> list = new List<CategoryMenu>();

using (SqlCommand command = new SqlCommand())

{

command.Connection = connect.getConnecttion();

command.CommandType = System.Data.CommandType.Text;

string query = @"select c.id, c.title, c.content, count(p.id) as soluong

from category c left join product p

on c.id = p.categoryId

group by c.id,c.title, c.content

order by soluong desc

";

command.CommandText = query;

SqlDataReader reader = command.ExecuteReader();

while (reader.Read())

{

CategoryMenu category = new CategoryMenu()

{

Id = Convert.ToInt32(reader["Id"]),

Title = reader["Title"].ToString() ?? "",

Content = reader["Content"].ToString() ?? "",

Count = Convert.ToInt32(reader["soluong"].ToString())

};

list.Add(category);

}

}

connect.closeConnection();

return list;

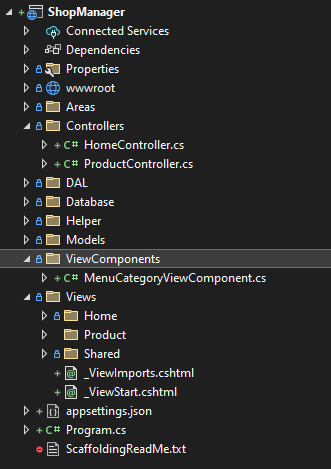
}

}

}

Tạo Folder mới: **ViewComponents**

Tạo Class mới: **MenuCategoryViewComponent.cs**



**ViewComponents/MenuCategoryViewComponent.cs**

using Microsoft.AspNetCore.Mvc;

using ShopManager.DAL;

using ShopManager.Models;

namespace ShopManager.ViewComponents

{

public class MenuCategoryViewComponent: ViewComponent

{

// query SQL

CategoryDAL categoryDAL = new CategoryDAL();

public IViewComponentResult Invoke()

{

List<CategoryMenu> categoryMenus = new List<CategoryMenu>();

categoryMenus = categoryDAL.getAllWithCount();

return View("Default",categoryMenus);

}

}

}

Tại Folder Views/Shared, tạo folder mới: **Components**

Tạo thêm Folder mới trong Folder vừa tạo: **MenuCategory** (tên Class trong folder ViewComponents)

Right Click tại folder **MenuCategory** -> Add -> View -> Razor View – Empty -> **Default.cshtml**

@model IEnumerable<ShopManager.Models.CategoryMenu>

<div class="col-lg-12">

<div class="mb-3">

<h4>Categories</h4>

<ul class="list-unstyled fruite-categorie">

@foreach(var item in Model)

{

<li>

<div class="d-flex justify-content-between fruite-name">

<a href="#"><i class="fas fa-apple-alt me-2"></i>@item.Title</a>

<span>(@item.Count)</span>

</div>

</li>

}

</ul>

</div>

</div>

Hiển thị Component **MenuCategory** tại trang **Cửa hàng (Product/Index.cshtml)**

**Views/Product/Index.cshtml**

@{

ViewData["Title"] = "Index";

}

<!-- Single Page Header start -->

<div class="container-fluid page-header py-5">

<h1 class="text-center text-white display-6">Shop</h1>

<ol class="breadcrumb justify-content-center mb-0">

<li class="breadcrumb-item"><a href="#">Home</a></li>

<li class="breadcrumb-item"><a href="#">Pages</a></li>

<li class="breadcrumb-item active text-white">Shop</li>

</ol>

</div>

<!-- Single Page Header End -->

<!-- Fruits Shop Start-->

<div class="container-fluid fruite py-5">

<div class="container py-5">

<h1 class="mb-4">Fresh fruits shop</h1>

<div class="row g-4">

<div class="col-lg-12">

<div class="row g-4">

<div class="col-xl-3">

<div class="input-group w-100 mx-auto d-flex">

<input type="search"

class="form-control p-3"

placeholder="keywords"

aria-describedby="search-icon-1" />

<span id="search-icon-1" class="input-group-text p-3">

<i class="fa fa-search"></i>

</span>

</div>

</div>

<div class="col-6"></div>

<div class="col-xl-3">

<div class="bg-light ps-3 py-3 rounded d-flex justify-content-between mb-4">

<label for="fruits">Default Sorting:</label>

<select id="fruits"

name="fruitlist"

class="border-0 form-select-sm bg-light me-3"

form="fruitform">

<**option** **value**="volvo">Nothing</**option**>

<**option** **value**="saab">Popularity</**option**>

<**option** **value**="opel">Organic</**option**>

<**option** **value**="audi">Fantastic</**option**>

</select>

</div>

</div>

</div>

<div class="row g-4">

<!-- Left Panel Start-->

<div class="col-lg-3">

<div class="row g-4">

<!-- Categories Start -->

<div class="col-lg-12">

~~@\* <div class="mb-3">~~

~~<h4>Categories</h4>~~

~~<ul class="list-unstyled fruite-categorie">~~

~~<li>~~

~~<div class="d-flex justify-content-between fruite-name">~~

~~<a href="#"><i class="fas fa-apple-alt me-2"></i>Apples</a>~~

~~<span>(3)</span>~~

~~</div>~~

~~</li>~~

~~<li>~~

~~<div class="d-flex justify-content-between fruite-name">~~

~~<a href="#"><i class="fas fa-apple-alt me-2"></i>Oranges</a>~~

~~<span>(5)</span>~~

~~</div>~~

~~</li>~~

~~<li>~~

~~<div class="d-flex justify-content-between fruite-name">~~

~~<a href="#"><i class="fas fa-apple-alt me-2"></i>Strawbery</a>~~

~~<span>(2)</span>~~

~~</div>~~

~~</li>~~

~~<li>~~

~~<div class="d-flex justify-content-between fruite-name">~~

~~<a href="#"><i class="fas fa-apple-alt me-2"></i>Banana</a>~~

~~<span>(8)</span>~~

~~</div>~~

~~</li>~~

~~<li>~~

~~<div class="d-flex justify-content-between fruite-name">~~

~~<a href="#"><i class="fas fa-apple-alt me-2"></i>Pumpkin</a>~~

~~<span>(5)</span>~~

~~</div>~~

~~</li>~~

~~</ul>~~

~~</div> \*@~~

@await Component.InvokeAsync("MenuCategory")

</div>

<!-- Categories End -->

<!-- Featured products Start -->

<div class="col-lg-12">

…

</div>

<!-- Featured products End -->

</div>

</div>

<!-- Left Panel End -->

<!-- Right Panel Start-->

<div class="col-lg-9">

...

</div>

<!-- Right Panel End -->

</div>

</div>

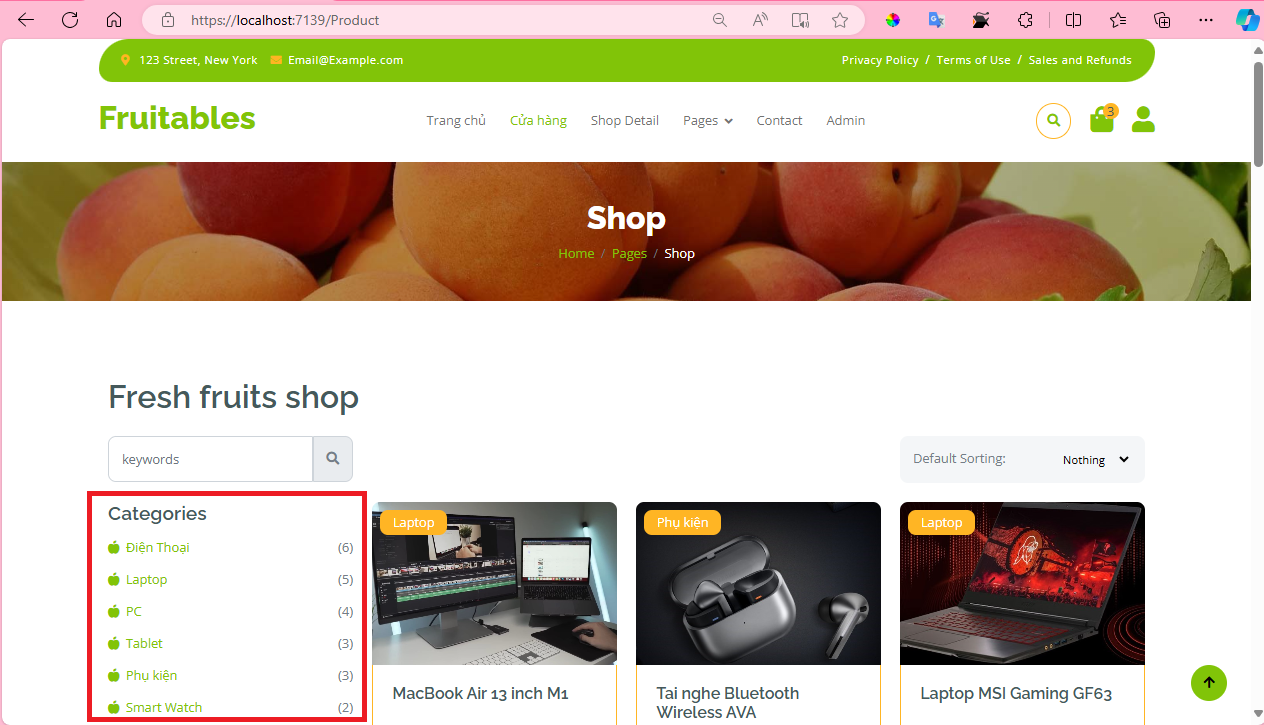
</div>

</div>

</div>

<!-- Fruits Shop End -->

Build Project -> Mở tab Cửa hàng



Thêm View Component cho trang **Product/Detail**

**Views/Product/Detail.cshtml**

@model ShopManager.Models.Product

@{

ViewData["Title"] = "Detail";

}

<!-- Single Page Header start -->

<div class="container-fluid page-header py-5">

<h1 class="text-center text-white display-6">Shop Detail</h1>

<ol class="breadcrumb justify-content-center mb-0">

<li class="breadcrumb-item"><a href="#">Home</a></li>

<li class="breadcrumb-item"><a href="#">Pages</a></li>

<li class="breadcrumb-item active text-white">Shop Detail</li>

</ol>

</div>

<!-- Single Page Header End -->

<!-- Single Product Start -->

<div class="container-fluid py-5 mt-5">

<div class="container py-5">

<div class="row g-4 mb-5">

<!-- Product Detail Start -->

<div class="col-lg-8 col-xl-9">

<div class="row g-4">....

</div>

<!-- Product Detail End -->

<!-- Right Panel Start -->

<div class="col-lg-4 col-xl-3">

<div class="row g-4 fruite">

<div class="col-lg-12">

<!-- Search Start -->

<div class="input-group w-100 mx-auto d-flex mb-4">

<input type="search"

class="form-control p-3"

placeholder="keywords"

aria-describedby="search-icon-1" />

<span id="search-icon-1" class="input-group-text p-3">

<i class="fa fa-search"></i>

</span>

</div>

<!-- Search End -->

<!-- Categories Start -->

~~@\* <div class="mb-4">~~

~~<h4>Categories</h4>~~

~~<ul class="list-unstyled fruite-categorie">~~

~~<li>~~

~~<div class="d-flex justify-content-between fruite-name">~~

~~<a href="#"><i class="fas fa-apple-alt me-2"></i>Apples</a>~~

~~<span>(3)</span>~~

~~</div>~~

~~</li>~~

~~<li>~~

~~<div class="d-flex justify-content-between fruite-name">~~

~~<a href="#"><i class="fas fa-apple-alt me-2"></i>Oranges</a>~~

~~<span>(5)</span>~~

~~</div>~~

~~</li>~~

~~<li>~~

~~<div class="d-flex justify-content-between fruite-name">~~

~~<a href="#"><i class="fas fa-apple-alt me-2"></i>Strawbery</a>~~

~~<span>(2)</span>~~

~~</div>~~

~~</li>~~

~~<li>~~

~~<div class="d-flex justify-content-between fruite-name">~~

~~<a href="#"><i class="fas fa-apple-alt me-2"></i>Banana</a>~~

~~<span>(8)</span>~~

~~</div>~~

~~</li>~~

~~<li>~~

~~<div class="d-flex justify-content-between fruite-name">~~

~~<a href="#"><i class="fas fa-apple-alt me-2"></i>Pumpkin</a>~~

~~<span>(5)</span>~~

~~</div>~~

~~</li>~~

~~</ul>~~

~~</div> \*@~~

@await Component.InvokeAsync("MenuCategory")

<!-- Categories End -->

</div>

<!-- Featured products Start -->

<div class="col-lg-12">...

<!-- Featured products End -->

</div>

</div>

<!-- Right Panel End -->

</div>

<!-- Related products Start -->

<h1 class="fw-bold mb-0">Related products</h1>

<div class="vesitable">...

<!-- Related products End -->

</div>

</div>

<!-- Single Product End -->

## Tạo View Component Featured Products

**DAL/ProductDAL.cs**

using Microsoft.Data.SqlClient;

using ShopManager.Areas.Admin.Models;

using ShopManager.Database;

using ShopManager.Models;

namespace ShopManager.DAL

{

public class ProductDAL

{

DBConnect connect = new DBConnect();

public List<Product> GetListProduct(int? idCategory)

{

…

}

public Product GetProductById(int Id)

{

...

}

public List<Product> GetFeaturedProducts (int limitProduct)

{

// get from product table

// top limitProduct

// sort by product.rate desc

connect.openConnection();

List<Product> list = new List<Product>();

using (SqlCommand command = new SqlCommand())

{

command.Connection = connect.getConnecttion();

command.CommandType = System.Data.CommandType.Text;

string query = @"select top " + limitProduct + @" a.id as ProductId, a.title as ProductTitle,

a.content as ProductContent, a.img as ProductImg, a.price as ProductPrice,

a.rate as ProductRate, a.createAt as ProductCreateAt, a.updateAt as ProductUpdateAt,

b.id as CategoryId, b.title as CategoryTitle

from product a join category b on a.categoryId = b.Id

order by a.rate desc;";

command.CommandText = query;

SqlDataReader reader = command.ExecuteReader();

while (reader.Read())

{

// ================== Cách 1 ===============

Product product = new Product()

{

Id = Convert.ToInt32(reader["ProductId"]),

Title = reader["ProductTitle"].ToString() ?? "",

Content = reader["ProductContent"].ToString() ?? "",

Img = reader["ProductImg"].ToString() ?? "",

Price = Convert.ToInt32(reader["ProductPrice"]),

Rate = Convert.ToDouble(reader["ProductRate"].ToString()),

CreateAt = DateTime.Parse(reader["ProductCreateAt"]?.ToString()),

UpdateAt = DateTime.Parse(reader["ProductUpdateAt"]?.ToString()),

CategoryId = Convert.ToInt32(reader["CategoryId"]),

CategoryTitle = reader["CategoryTitle"].ToString(),

};

list.Add(product);

}

}

connect.closeConnection();

return list;

}

}

}

Tại Folder ViewComponents, Tại Class mới: **FeaturedProductsViewComponent.cs**

**ViewComponents/FeaturedProductsViewComponent.cs**

using Microsoft.AspNetCore.Mvc;

using ShopManager.DAL;

using ShopManager.Models;

namespace ShopManager.ViewComponents

{

public class FeaturedProductsViewComponent : ViewComponent

{

ProductDAL productDAL = new ProductDAL();

public IViewComponentResult Invoke(int? limit)

{

int limitProduct = limit ?? 4;

List<Product> featuredProducts = new List<Product>();

featuredProducts = productDAL.GetFeaturedProducts(limitProduct);

return View("FeatureProduct", featuredProducts);

}

}

}

Tại Folder **Views/Shared/Components**, tạo Folder mới: **FeaturedProducts**

-> tạo View mới: **FeatureProduct.cshtml**

Mở file wwwroot/shop-detail.html

Copy thẻ <!-- Featured products Start --> .. <!-- Featured products End -->

Paste vào file **FeatureProduct.cshtml**

Hiển thị Model Product

**Views/Shared/Components/ FeaturedProducts/FeatureProduct.cshtml**

@model IEnumerable<ShopManager.Models.Product>

<h4 class="mb-4">Featured products</h4>

@foreach (var item in Model)

{

<div class="d-flex align-items-center justify-content-start">

<div class="rounded" style="width: 100px; height: 100px">

<**img** src="~/images/SanPham/@item.Img" class="img-fluid rounded" alt="@item.Img"

style="width: 90px; height: 50px; object-fit:contain" />

</div>

<div>

<h6 class="mb-2">@item.Title</h6>

<div class="d-flex mb-2">

<i class="fa fa-star text-secondary"></i>

<i class="fa fa-star text-secondary"></i>

<i class="fa fa-star text-secondary"></i>

<i class="fa fa-star text-secondary"></i>

<i class="fa fa-star"></i>

</div>

<div class="d-flex mb-2">

<h6 class="fw-bold me-2">@item.Price.ToString("#,##0 VND")</h6>

</div>

</div>

</div>

}

<div class="d-flex justify-content-center my-4">

<**a** **asp-area**="" **asp-controller**="Product" **asp-action**="Index"

class="btn border border-secondary px-4 py-3 rounded-pill text-primary w-100">View More</**a**>

</div>

Thêm Component FeaturedProducts tại trang **Home/Index.cshtml**

**Views/Home/Index.cshtml**

@model IEnumerable<ShopManager.Models.Product>

@{

ViewData["Title"] = "Index";

}

<!-- Single Page Header start -->

<div class="container-fluid page-header py-5">

...

</div>

<!-- Single Page Header End -->

<!-- Fruits Shop Start-->

<div class="container-fluid fruite py-5">

<div class="container py-5">

<h1 class="mb-4">Fresh fruits shop</h1>

<div class="row g-4">

<div class="col-lg-12">

<div class="row g-4">

...

</div>

<div class="row g-4">

<!-- Left Panel Start-->

<div class="col-lg-3">

<div class="row g-4">

<!-- Categories Start -->

<div class="col-lg-12">

@await Component.InvokeAsync("MenuCategory")

</div>

<!-- Categories End -->

<!-- Featured products Start -->

<div class="col-lg-12">

@await Component.InvokeAsync("FeaturedProducts",new { limit= 5})

</div>

<!-- Featured products End -->

</div>

</div>

<!-- Left Panel End -->

<!-- Right Panel Start-->

<div class="col-lg-9">

...

</div>

<!-- Right Panel End -->

</div>

</div>

</div>

</div>

</div>

Thêm Component FeaturedProducts tại trang **Product/Details.cshtml**

**Views/Product/Detail.cshtml**

@model ShopManager.Models.Product

@{

ViewData["Title"] = "Detail";

}

<!-- Single Page Header start -->

<div class="container-fluid page-header py-5"> ...</div>

<!-- Single Page Header End -->

<!-- Single Product Start -->

<div class="container-fluid py-5 mt-5">

<div class="container py-5">

<div class="row g-4 mb-5">

<!-- Product Detail Start -->

<div class="col-lg-8 col-xl-9">

...

</div>

<!-- Product Detail End -->

<!-- Right Panel Start -->

<div class="col-lg-4 col-xl-3">

<div class="row g-4 fruite">

<div class="col-lg-12">

<!-- Search Start -->

<div class="input-group w-100 mx-auto d-flex mb-4">

...

</div>

<!-- Search End -->

<!-- Categories Start -->

@await Component.InvokeAsync("MenuCategory")

<!-- Categories End -->

</div>

<!-- Featured products Start -->

<div class="col-lg-12">

@await Component.InvokeAsync("FeaturedProducts",new { limit= 5})

</div>

<!-- Featured products End -->

</div>

</div>

<!-- Right Panel End -->

</div>

<!-- Related products Start -->

<div class="col-lg-8 col-xl-9">

...

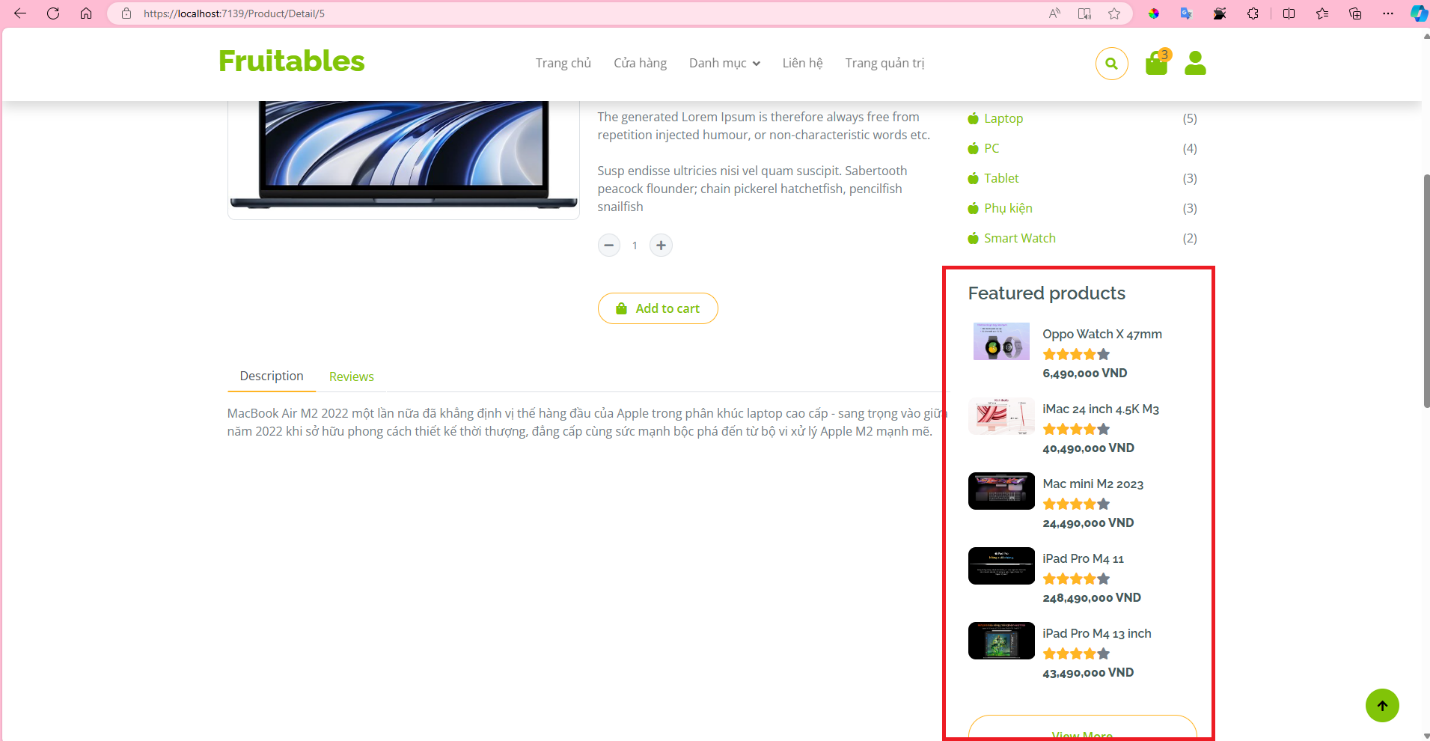
</div>

<!-- Related products End -->

</div>

</div>

<!-- Single Product End -->

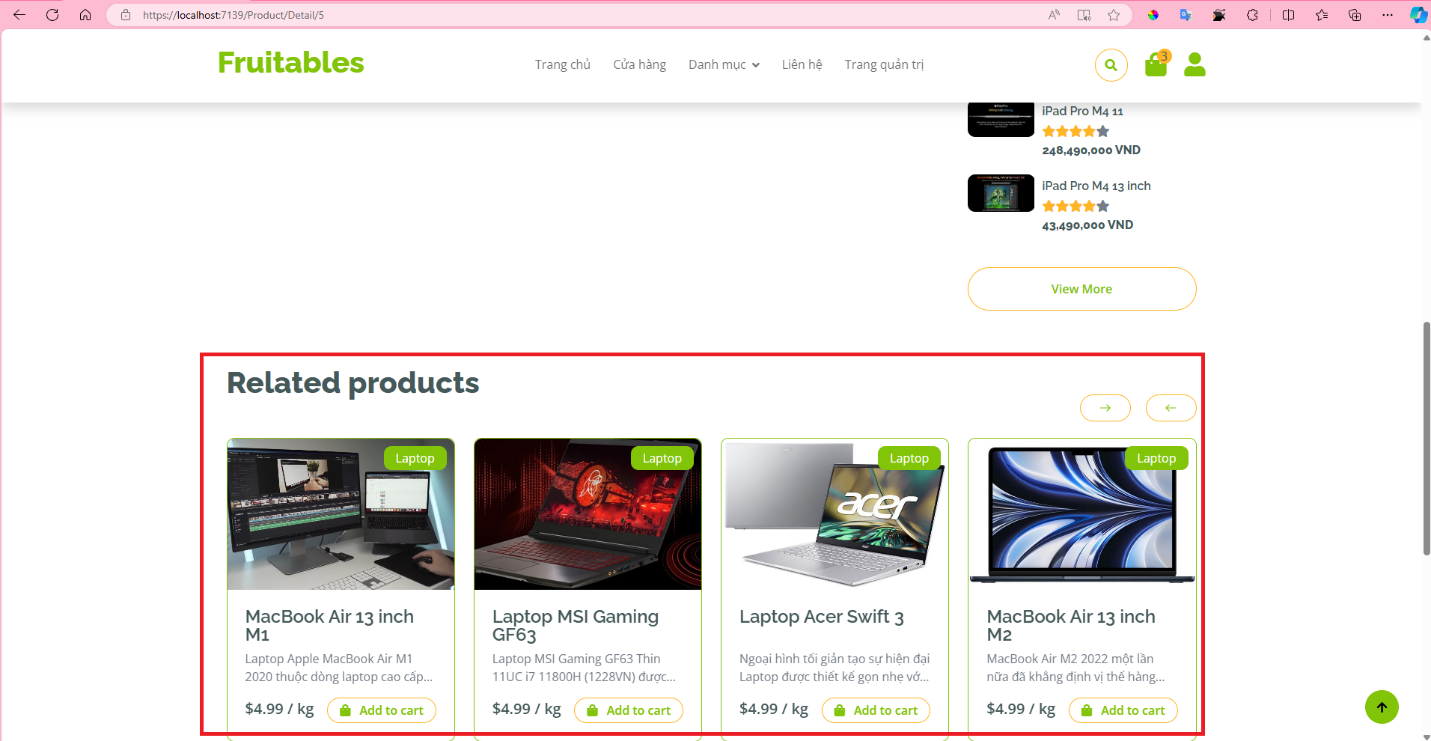


**Bài tập**: Tạo View Component Related Product (shop-detail.html), lấy danh sách Product có cùng Id Category, sắp xếp theo product.rating giảm dần.

Gợi ý:

* Tạo DAL lấy danh sách product có cùng Id Category, sắp xếp theo product.rating giảm dần. (không lấy product đang hiển thị)
* Tạo View Component: RelatedProductViewComponent.cs
* Thêm View Component tại Views/Product/Detail.cshtml

<!-- Related products Start --> .. <!-- Related products End -->



## Tạo View Component Navbar

Lấy dữ liệu từ bảng menu để tạo **Navbar View Component**

Tạo Model mới: **MenuItem.cs**

**Models/MenuItem.cs**

namespace ShopManager.Models

{

public class MenuItem

{

public int Id { get; set; }

public string Title { get; set; }

public int? ParentId { get; set; }

public string? MenuUrl { get; set; }

public int MenuIndex { get; set; }

public bool isVisible { get; set; }

}

}

Tạo Model mới: **NavbarItem.cs**

**Models/NavbarItem.cs**

namespace ShopManager.Models

{

public class NavbarItem

{

public int Id { get; set; }

public string Title { get; set; }

public int? ParentId { get; set; }

public string? MenuUrl { get; set; }

public int MenuIndex { get; set; }

public bool isVisible { get; set; }

public List<NavbarItem>? subItems { get; set; }

}

}

Tạo DAL mới: **MenuDAL.cs**

**DAL/MenuDAL.cs**

using Microsoft.Data.SqlClient;

using ShopManager.Database;

using ShopManager.Models;

namespace ShopManager.DAL

{

public class MenuDAL

{

DBConnect connect = new DBConnect();

public List<MenuItem> GetAllMenu()

{

connect.openConnection();

List<MenuItem> list = new List<MenuItem>();

using (SqlCommand command = new SqlCommand())

{

command.Connection = connect.getConnecttion();

command.CommandType = System.Data.CommandType.Text;

string query = @"select \* from menu order by menuIndex ";

command.CommandText = query;

SqlDataReader reader = command.ExecuteReader();

while (reader.Read())

{

//var parentId = DbNu;

MenuItem MenuItem = new MenuItem()

{

Id = Convert.ToInt32(reader["Id"]),

Title = reader["Title"].ToString() ?? "",

ParentId = reader["ParentId"] != DBNull.Value ? Convert.ToInt32(reader["ParentId"]) : null,

MenuUrl = reader["MenuUrl"]?.ToString() ?? null,

MenuIndex = Convert.ToInt32(reader["MenuIndex"]),

isVisible = Convert.ToInt32(reader["isVisible"]) == 1

};

list.Add(MenuItem);

}

}

connect.closeConnection();

return list;

}

}

}

Tạo ViewComponents mới: MenuDynamicViewComponent.cs

**ViewComponents/ MenuDynamicViewComponent.cs**

using Microsoft.AspNetCore.Mvc;

using ShopManager.DAL;

using ShopManager.Models;

using System.Security.Claims;

namespace ShopManager.ViewComponents

{

public class MenuDynamicViewComponent : ViewComponent

{

MenuDAL menuDAL = new MenuDAL();

public IViewComponentResult Invoke()

{

List<MenuItem> listMenu = new List<MenuItem>();

List<NavbarItem> navBar = new List<NavbarItem>();

listMenu = menuDAL.GetAllMenu();

// Sử dụng danh sách tạm thời để lưu các mục cần xóa

var itemsToRemove = new List<MenuItem>();

// Nếu không được phân quyền để truy cập trang Admin

// Và đường dẫn Url của Menu chứa Area Admin

foreach (var item in listMenu)

{

if (!item.isVisible)

{

itemsToRemove.Add(item);

}

}

// Xóa các phần tử không hợp lệ

foreach (var item in itemsToRemove)

{

listMenu.Remove(item);

}

//Lấy Tất cả Menu

foreach (var item in listMenu)

{

// Is Nav Bar Item

if (item.ParentId == null)

{

navBar.Add(

new NavbarItem()

{

Id = item.Id,

ParentId = item.ParentId,

Title = item.Title,

MenuUrl = item.MenuUrl,

MenuIndex = item.MenuIndex,

isVisible = item.isVisible,

subItems = new List<NavbarItem>(),

}

);

}

}

//Laays menu Con

foreach (var item in listMenu)

{

// Is Nav Bar Item

if (item.ParentId != null)

{

//Find Item Parent

var navbarParent = navBar.Find(p => p.Id == item.ParentId);

// if HasValue

if (navbarParent != null)

{

// Add to List Dropdown Item

navbarParent.subItems!.Add(

new NavbarItem()

{

Id = item.Id,

ParentId = item.ParentId,

Title = item.Title,

MenuUrl = item.MenuUrl,

MenuIndex = item.MenuIndex,

isVisible = item.isVisible,

subItems = null

}

);

}

}

}

Console.WriteLine(navBar);

return View("MenuDynamic", navBar);

}

}

}

Tại Folder **Views/Shared/Components**, tại Folder mới: **MenuDynamic**

Tại View mới: **MenuDynamic.cshtml**

Mở file **Views/Shares/\_LayoutCustomer.cshtml**

Copy thẻ <!-- Navbar Start --> ... <!-- Navbar End -->

Paste vào file **MenuDynamic.cshtml**

Hiển thị Model **NavbarItem**

**Views/Shared/Components/MenuDynamic/MenuDynamic.cshtml**

@model IEnumerable<ShopManager.Models.NavbarItem>

<!-- Navbar start -->

<div class="container-fluid fixed-top">

<div class="container topbar bg-primary d-none d-lg-block">

<div class="d-flex justify-content-between">

<div class="top-info ps-2">

<small class="me-3">

<i class="fas fa-map-marker-alt me-2 text-secondary"></i>

<a href="#" class="text-white">123 Street, New York</a>

</small>

<small class="me-3">

<i class="fas fa-envelope me-2 text-secondary"></i><a href="#" class="text-white">Email@Example.com</a>

</small>

</div>

<div class="top-link pe-2">

<a href="#" class="text-white"><small class="text-white mx-2">Privacy Policy</small>/</a>

<a href="#" class="text-white"><small class="text-white mx-2">Terms of Use</small>/</a>

<a href="#" class="text-white"><small class="text-white ms-2">Sales and Refunds</small></a>

</div>

</div>

</div>

<div class="container px-0">

<nav class="navbar navbar-light bg-white navbar-expand-xl">

<a href="index.html" class="navbar-brand"><h1 class="text-primary display-6">Fruitables</h1></a>

<button class="navbar-toggler py-2 px-3"

type="button"

data-bs-toggle="collapse"

data-bs-target="#navbarCollapse">

<span class="fa fa-bars text-primary"></span>

</button>

<div class="collapse navbar-collapse bg-white" id="navbarCollapse">

<div class="navbar-nav mx-auto">

@foreach (var item in Model)

{

if (item.subItems?.Count() == 0)

{

<a href="@item.MenuUrl"

class="nav-item nav-link

@(item!.MenuUrl == Context.Request.Path ? "active" : "")">

@item.Title

</a>

}

else

{

<div class="nav-item dropdown">

<a href="#" class="nav-link dropdown-toggle" data-bs-toggle="dropdown">@item.Title</a>

<div class="dropdown-menu m-0 bg-secondary rounded-0">

@foreach (var subItem in item!.subItems!)

{

<a href="@subItem.MenuUrl" class="dropdown-item">@subItem.Title</a>

}

</div>

</div>

}

}

~~@\* <a asp-controller="Home" asp-action="index"~~

~~class="nav-item nav-link @(ViewContext.RouteData.Values["Controller"]?.ToString() == "Home" ? "active" : "")">Trang chủ</a>~~

~~<a asp-controller="Product" asp-action="index"~~

~~class="nav-item nav-link @(ViewContext.RouteData.Values["Controller"]?.ToString() == "Product" ? "active" : "")">Cửa hàng</a>~~

~~<a href="shop-detail.html" class="nav-item nav-link">Shop Detail</a>~~

~~<div class="nav-item dropdown">~~

~~<a href="#" class="nav-link dropdown-toggle" data-bs-toggle="dropdown">Pages</a>~~

~~<div class="dropdown-menu m-0 bg-secondary rounded-0">~~

~~<a href="cart.html" class="dropdown-item">Cart</a>~~

~~<a href="chackout.html" class="dropdown-item">Chackout</a>~~

~~<a href="testimonial.html" class="dropdown-item">Testimonial</a>~~

~~<a href="404.html" class="dropdown-item">404 Page</a>~~

~~</div>~~

~~</div>~~

~~<a href="contact.html" class="nav-item nav-link">Contact</a>~~

~~<a asp-area="Admin" asp-controller="ProductAdmin" asp-action="Index" class="nav-item nav-link">Admin</a> \*@~~

</div>

<div class="d-flex m-3 me-0">

<button class="btn-search btn border border-secondary btn-md-square rounded-circle bg-white me-4"

data-bs-toggle="modal"

data-bs-target="#searchModal">

<i class="fas fa-search text-primary"></i>

</button>

<a href="#" class="position-relative me-4 my-auto">

<i class="fa fa-shopping-bag fa-2x"></i>

<span class="position-absolute bg-secondary rounded-circle d-flex align-items-center justify-content-center text-dark px-1"

style="top: -5px; left: 15px; height: 20px; min-width: 20px">3</span>

</a>

<**a** **asp-area**="" **asp-controller**="Customer" **asp-action**="Profile" class="my-auto">

<i class="fas fa-user fa-2x"></i>

</**a**>

</div>

</div>

</nav>

</div>

</div>

<!-- Navbar End -->

Thêm View Component **MenuDynamic** tại \_LayoutCustomer.cshtml

**Views/Shares/\_LayoutCustomer.cshtml**

<!DOCTYPE html>

<html>

<head>

..

</head>

<body>

<!-- Spinner Start -->

<div id="spinner"

..

</div>

<!-- Spinner End -->

<!-- Navbar Start -->

@await Component.InvokeAsync("MenuDynamic")

<!-- Navbar End -->

<!-- Modal Search Start -->

<div class="modal fade" id="searchModal" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">

..

</div>

<!-- Modal Search End -->

<div style="min-height:600px">

@RenderBody()

</div>

<!-- Footer Start -->

<div class="container-fluid bg-dark text-white-50 footer pt-5 mt-5">

..

</div>

<!-- Footer End -->

<!-- Back to Top -->

<a href="#" class="btn btn-primary border-3 border-primary rounded-circle back-to-top">

<i class="fa fa-arrow-up"></i>

</a>

<!-- JavaScript Libraries -->

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.4/jquery.min.js"></script>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0/dist/js/bootstrap.bundle.min.js"></script>

<**script** src="~/lib/easing/easing.min.js"></**script**>

<**script** src="~/lib/waypoints/waypoints.min.js"></**script**>

<**script** src="~/lib/lightbox/js/lightbox.min.js"></**script**>

<**script** src="~/lib/owlcarousel/owl.carousel.min.js"></**script**>

<!-- Template Javascript -->

<**script** src="~/js/main.js"></**script**>

</body>

</html>

**--- THE END ---**