# Giỏ hàng Session và thanh toán

**Mục Lục**

[Giỏ hàng Session và thanh toán 1](#_Toc173834898)

[I. Tạo giỏ hàng với Session 2](#_Toc173834899)

[1. Khai báo Session 2](#_Toc173834900)

[2. Hiển thị giỏ hàng 3](#_Toc173834901)

[3. Thêm sản phẩm vào giỏ hàng 5](#_Toc173834902)

[4. Thay đổi số lượng sản phẩm trong giỏ hàng 11](#_Toc173834903)

[5. Xóa sản phẩm trong giỏ hàng 14](#_Toc173834904)

[6. Hiển thị số lượng sản phẩm trong giỏ hàng 16](#_Toc173834905)

[II. Thanh toán 20](#_Toc173834906)

## Tạo giỏ hàng với Session

### Khai báo Session

Tài liệu: [Session in ASP.NET Core | Microsoft Learn](https://learn.microsoft.com/en-us/aspnet/core/fundamentals/app-state?view=aspnetcore-8.0)

**Program.cs**

using Microsoft.AspNetCore.Authentication.Cookies;

var builder = WebApplication.CreateBuilder(args);

// Add services to the container.

builder.Services.AddControllersWithViews();

// Use Cookie

builder.Services.AddAuthentication(CookieAuthenticationDefaults.AuthenticationScheme)

.AddCookie(option =>

{

option.LoginPath = "/Customer/SignIn";

//truy cập trang có yêu cầu đăng nhập

//mà người dùng chưa đăng nhập

//thì sẽ tự động chuyển sang đường dẫn được khai báo trong LoginPath

option.AccessDeniedPath = "/AccessDenied";

//Khi người dùng truy cập trang

// mà người dùng đó không được quyền truy cập

// thì sẽ tự động chuyển sang đường dẫn được khai báo trong AccessDeniedPath

});

//Cho phép Field Null trong Form

builder.Services.AddControllersWithViews(options => options.SuppressImplicitRequiredAttributeForNonNullableReferenceTypes = true);

// Use Session

builder.Services.AddDistributedMemoryCache();

builder.Services.AddSession(options =>

{

options.IdleTimeout = TimeSpan.FromSeconds(10);

options.Cookie.HttpOnly = true;

options.Cookie.IsEssential = true;

});

var app = builder.Build();

// Configure the HTTP request pipeline.

if (!app.Environment.IsDevelopment())

{

app.UseExceptionHandler("/Home/Error");

// The default HSTS value is 30 days. You may want to change this for production scenarios, see https://aka.ms/aspnetcore-hsts.

app.UseHsts();

}

app.UseHttpsRedirection();

app.UseStaticFiles();

app.UseRouting();

app.UseAuthentication();

app.UseAuthorization();

app.UseSession();

app.MapControllerRoute(

name: "admin",

pattern: "{area:exists}/{controller=ProductAdmin}/{action=Index}/{id?}");

app.MapControllerRoute(

name: "default",

pattern: "{controller=Home}/{action=Index}/{id?}");

app.Run();

### Hiển thị giỏ hàng

Tại Folder Helper, tạo Class mới: MyConst.cs

Helper/MyConst.cs

namespace ShopManager.Helper

{

public class MyConst

{

public const string CART\_KEY = "MYCART";

}

}

Tạo Controller mới: CartController.cs

Controllers/CartController.cs

using Microsoft.AspNetCore.Mvc;

using ShopManager.DAL;

using ShopManager.Helper;

using ShopManager.Models;

using System.Runtime.CompilerServices;

namespace ShopManager.Controllers

{

public class CartController : Controller

{

ProductDAL productDAL = new ProductDAL();

public List<CartItem> Cart => HttpContext.Session.Get<List<CartItem>>(MyConst.CART\_KEY) ??

new List<CartItem>();

public IActionResult Index()

{

return View(Cart);

}

}

}

Thêm Razor View – Empty cho function Index

Mở file wwwroot/cart.html

Copy

<!-- Single Page Header Start --> ... <!-- Single Page Header End -->

<!-- Cart Page Start --> ... <!-- Cart Page End -->

Paste vào file Index.cshtml

Hiển thị Model CartItem

**Views/Cart/Index.cshtml**

@model IEnumerable<ShopManager.Models.CartItem>

<!-- Single Page Header start -->

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

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

</ol>

</div>

<!-- Single Page Header End -->

<!-- Cart Page Start -->

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

<div class="container py-5">

<div class="table-responsive">

<table class="table">

<thead>

<tr>

<th scope="col">Products</th>

<th scope="col">Name</th>

<th scope="col">Price</th>

<th scope="col">Quantity</th>

<th scope="col">Total</th>

<th scope="col">Handle</th>

</tr>

</thead>

<tbody>

@foreach (var item in Model)

{

<tr id="idProduct-@item.IdProduct">

<th scope="row">

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

<**img** src="~/images/SanPham/@item.Img" class="img-fluid me-5 rounded-circle" style="width: 80px; height: 80px;" alt="@item.Img">

</div>

</th>

<td>

<p class="mb-0 mt-4">@item.Name</p>

</td>

<td>

<p class="mb-0 mt-4">@item.Price.ToString("#,##0 VND")</p>

</td>

<td>

<div class="input-group quantity mt-4" style="width: 100px;">

<div class="input-group-btn">

<**a** class="btn btn-sm btn-minus rounded-circle bg-light border">

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

</**a**>

</div>

<input type="number"

class="form-control form-control-sm text-center border-0"

value="@item.Quantity">

<div class="input-group-btn">

<**a** class="btn btn-sm btn-plus rounded-circle bg-light border">

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

</**a**>

</div>

</div>

</td>

<td>

<p class="mb-0 mt-4">@item.Total.ToString("#,##0 VND")</p>

</td>

<td>

<**a** class="btn btn-md rounded-circle bg-light border mt-4">

<i class="fa fa-times text-danger"></i>

</**a**>

</td>

</tr>

}

</tbody>

</table>

</div>

<div class="row g-4 justify-content-end">

<div class="col-8"></div>

<div class="col-sm-8 col-md-7 col-lg-6 col-xl-4">

<div class="bg-light rounded">

<div class="py-4 mb-4 border-top border-bottom d-flex justify-content-between">

<h5 class="mb-0 ps-4 me-4">Tổng tiền</h5>

<p class="mb-0 pe-4">@Model.Sum(p => p.Total).ToString("#,##0 VND")</p>

</div>

<button class="btn border-secondary rounded-pill px-4 py-3 text-primary text-uppercase mb-4 ms-4" type="button">Thanh toán</button>

</div>

</div>

</div>

</div>

</div>

<!-- Cart Page End -->

### Thêm sản phẩm vào giỏ hàng

Controllers/CartController.cs

using Microsoft.AspNetCore.Mvc;

using ShopManager.DAL;

using ShopManager.Helper;

using ShopManager.Models;

using System.Runtime.CompilerServices;

namespace ShopManager.Controllers

{

public class CartController : Controller

{

ProductDAL productDAL = new ProductDAL();

public List<CartItem> Cart => HttpContext.Session.Get<List<CartItem>>(MyConst.CART\_KEY) ??

new List<CartItem>();

public IActionResult Index()

{

return View(Cart);

}

public IActionResult AddToCart(int id, int quantity = 1)

{

var gioHang = Cart;

var item = gioHang.SingleOrDefault(p => p.IdProduct == id);

if (item == null)

{

Product? productById = productDAL.GetProductById(id);

if (productById == null)

{

TempData["Message"] = "Khong tim thay san pham";

return Redirect("/404");

}

item = new CartItem

{

IdProduct = productById.Id,

Img = productById.Img,

Name = productById.Title,

Price = productById.Price,

Rate = productById.Rate,

Quantity = quantity

};

gioHang.Add(item);

}

else

{

item.Quantity += quantity;

}

HttpContext.Session.Set(MyConst.CART\_KEY, gioHang);

return RedirectToAction("Index");

}

}

}

**Views/Product/Index.cshtml**

@model IEnumerable<ShopManager.Models.Product>

@{

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">

@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 class="row g-4 justify-content-center">

@foreach (var item in Model)

{

<!-- Item 1 Start -->

<div class="col-md-6 col-lg-6 col-xl-4">

<div class="rounded position-relative fruite-item">

<div class="fruite-img">

<**a** **asp-area**="" **asp-controller**="Product" **asp-action**="Detail" **asp-route-id**="@item.Id">

<**img** src="~/Images/SanPham/@item.Img"

class="img-fluid w-100 rounded-top"

style="aspect-ratio:1.5; object-fit:cover"

alt="@item.Img" />

</**a**>

</div>

<div class="text-white bg-secondary px-3 py-1 rounded position-absolute"

style="top: 10px; left: 10px">

@item.CategoryTitle

</div>

<div class="p-4 border border-secondary border-top-0 rounded-bottom">

<h5 style="line-height:1.5rem; height:3rem; display: -webkit-box; -webkit-line-clamp: 2; -webkit-box-orient: vertical; overflow: hidden;">

@item.Title

</h5>

<p style="line-height:1.5rem; height:4.5rem; display: -webkit-box; -webkit-line-clamp: 3; -webkit-box-orient: vertical; overflow: hidden;">

@item.Content

</p>

<div class="d-flex justify-content-start flex-lg-wrap">

<p class="text-dark fs-6 fw-bold mb-0">@item.Price.ToString("#,##0 VND")</p>

</div>

<div class="d-flex justify-content-end flex-lg-wrap">

<**a** **asp-controller**="Cart" **asp-action**="AddToCart" **asp-route-id**="@item.Id"

class="btn border border-secondary rounded-pill px-3 text-primary">

<i class="fa fa-shopping-bag me-2 text-primary"></i> Add to cart

</**a**>

</div>

</div>

</div>

</div>

<!-- Item 1 End -->

}

<!-- Pagination Start-->

<div class="col-12">

<div class="pagination d-flex justify-content-center mt-5">

<a href="#" class="rounded">&laquo;</a>

<a href="#" class="active rounded">1</a>

<a href="#" class="rounded">2</a>

<a href="#" class="rounded">3</a>

<a href="#" class="rounded">4</a>

<a href="#" class="rounded">5</a>

<a href="#" class="rounded">6</a>

<a href="#" class="rounded">&raquo;</a>

</div>

</div>

<!-- Pagination End-->

</div>

</div>

<!-- Right Panel End -->

</div>

</div>

</div>

</div>

</div>

**Views/Product/Detail.cshtml**

@model ShopManager.Models.Product

@{

ViewData["Title"] = "Detail";

}

<!-- Single Page Header start -->

...

<!-- 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 class="col-lg-6">

<div class="border rounded">

<a href="#">

<**img** src="~/Images/SanPham/@Model.Img" class="img-fluid rounded" alt="Image" />

</a>

</div>

</div>

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

<!-- Text Start -->

<h4 class="fw-bold mb-3">@Model.Title</h4>

<p class="mb-3">Category: @Model.CategoryTitle</p>

<h5 class="fw-bold mb-3">

@Model.Price.ToString("#,##0 VND")

</h5>

<!-- Text End -->

<!-- Rating Start -->

<div class="d-flex mb-4">

<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>

<!-- Rating End -->

<p class="mb-4">

The generated Lorem Ipsum is therefore always free from repetition injected humour,

or non-characteristic words etc.

</p>

<p class="mb-4">

Susp endisse ultricies nisi vel quam suscipit. Sabertooth peacock flounder; chain

pickerel hatchetfish, pencilfish snailfish

</p>

<**form** **asp-controller**="Cart" **asp-action**="AddToCart" **asp-route-id**="@Model.Id">

<div class="input-group quantity mb-5" style="width: 100px">

<div class="input-group-btn">

<button class="btn btn-sm btn-minus rounded-circle bg-light border">

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

</button>

</div>

<input type="number"

name="quantity"

class="form-control form-control-sm text-center border-0"

value="1" />

<div class="input-group-btn">

<button class="btn btn-sm btn-plus rounded-circle bg-light border">

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

</button>

</div>

</div>

<button type="submit" class="btn border border-secondary rounded-pill px-4 py-2 mb-4 text-primary">

<i class="fa fa-shopping-bag me-2 text-primary"></i>

Add to cart

</button>

</**form**>

</div>

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

<nav>

...

</nav>

<div class="tab-content mb-5">

<!-- Description Start-->

<div class="tab-pane active"

id="nav-about"

role="tabpanel"

aria-labelledby="nav-about-tab">

<p>

@Model.Content

</p>

</div>

<!-- Description End-->

...

### Thay đổi số lượng sản phẩm trong giỏ hàng

Controllers/CartController.cs

namespace ShopManager.Controllers

{

public class CartController : Controller

{

ProductDAL productDAL = new ProductDAL();

public List<CartItem> Cart => HttpContext.Session.Get<List<CartItem>>(MyConst.CART\_KEY) ??

new List<CartItem>();

public IActionResult Index()

{

return View(Cart);

}

public IActionResult AddToCart(int id, int quantity = 1)

{

var gioHang = Cart;

var item = gioHang.SingleOrDefault(p => p.IdProduct == id);

if (item == null)

{

Product? productById = productDAL.GetProductById(id);

if (productById == null)

{

TempData["Message"] = "Khong tim thay san pham";

return Redirect("/404");

}

item = new CartItem

{

IdProduct = productById.Id,

Img = productById.Img,

Name = productById.Title,

Price = productById.Price,

Rate = productById.Rate,

Quantity = quantity

};

gioHang.Add(item);

}

else

{

item.Quantity += quantity;

}

HttpContext.Session.Set(MyConst.CART\_KEY, gioHang);

return RedirectToAction("Index");

}

public IActionResult ChangeQuantityCart(int id, bool isIncrement = true, int quantity = 1)

{

// Lấy toàn bộ giỏ hàng

var gioHang = Cart;

var item = gioHang.SingleOrDefault(p => p.IdProduct == id);

//kiểm tra tồn tại Product

if (item == null)

{

TempData["Message"] = "Khong tim thay san pham";

return Redirect("/404");

}

else

{

// Nếu là button tăng số lượng

if (isIncrement)

{

item.Quantity += quantity;

}

// Nếu là button giảm số lượng

else

{

item.Quantity -= quantity;

// Nếu khách hàng nhập số lượng <= 0 thì xóa sản phẩm đó ra khỏi giỏ hàng

if (item.Quantity <= 0)

{

gioHang.Remove(item);

}

}

}

// Lưu thay đổi

HttpContext.Session.Set(MyConst.CART\_KEY, gioHang);

return RedirectToAction("Index");

}

}

}

Views/Cart/Index.cshtml

@model IEnumerable<ShopManager.Models.CartItem>

<!-- Single Page Header start -->

...

<!-- Single Page Header End -->

<!-- Cart Page Start -->

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

<div class="container py-5">

<div class="table-responsive">

<table class="table">

<thead>

...

</thead>

<tbody>

@foreach (var item in Model)

{

<tr id="idProduct-@item.IdProduct">

<th scope="row">

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

<**img** src="~/images/SanPham/@item.Img" class="img-fluid me-5 rounded-circle" style="width: 80px; height: 80px;" alt="@item.Img">

</div>

</th>

<td>

<p class="mb-0 mt-4">@item.Name</p>

</td>

<td>

<p class="mb-0 mt-4">@item.Price.ToString("#,##0 VND")</p>

</td>

<td>

<div class="input-group quantity mt-4" style="width: 100px;">

<div class="input-group-btn">

<**a** **asp-controller**="cart" **asp-action**="ChangeQuantityCart"

**asp-route-id**="@item.IdProduct" **asp-route-isIncrement**="false"

class="btn btn-sm btn-minus rounded-circle bg-light border">

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

</**a**>

</div>

<input type="number"

class="form-control form-control-sm text-center border-0"

value="@item.Quantity">

<div class="input-group-btn">

<**a** **asp-controller**="cart" **asp-action**="ChangeQuantityCart"

**asp-route-id**="@item.IdProduct" **asp-route-isIncrement**="true"

class="btn btn-sm btn-plus rounded-circle bg-light border">

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

</**a**>

</div>

</div>

</td>

<td>

<p class="mb-0 mt-4">@item.Total.ToString("#,##0 VND")</p>

</td>

<td>

<**a** class="btn btn-md rounded-circle bg-light border mt-4">

<i class="fa fa-times text-danger"></i>

</**a**>

</td>

</tr>

}

</tbody>

</table>

</div>

<div class="row g-4 justify-content-end">

<div class="col-8"></div>

<div class="col-sm-8 col-md-7 col-lg-6 col-xl-4">

<div class="bg-light rounded">

<div class="py-4 mb-4 border-top border-bottom d-flex justify-content-between">

<h5 class="mb-0 ps-4 me-4">Tổng tiền</h5>

<p class="mb-0 pe-4">@Model.Sum(p => p.Total).ToString("#,##0 VND")</p>

</div>

<button class="btn border-secondary rounded-pill px-4 py-3 text-primary text-uppercase mb-4 ms-4" type="button">Thanh toán</button>

</div>

</div>

</div>

</div>

</div>

<!-- Cart Page End -->

### Xóa sản phẩm trong giỏ hàng

Controllers/CartController.cs

namespace ShopManager.Controllers

{

public class CartController : Controller

{

ProductDAL productDAL = new ProductDAL();

public List<CartItem> Cart => HttpContext.Session.Get<List<CartItem>>(MyConst.CART\_KEY) ??

new List<CartItem>();

public IActionResult Index(){ ... }

public IActionResult AddToCart(int id, int quantity = 1){...}

public IActionResult ChangeQuantityCart(int id, bool isIncrement = true, int quantity = 1)

{ ... }

public IActionResult RemoveCart(int id)

{

var gioHang = Cart;

var item = gioHang.SingleOrDefault(p => p.IdProduct == id);

if (item != null)

{

gioHang.Remove(item);

HttpContext.Session.Set(MyConst.CART\_KEY, gioHang);

}

return RedirectToAction("Index");

}

}

}

Views/Cart/Index.cshtml

@model IEnumerable<ShopManager.Models.CartItem>

<!-- Single Page Header start -->

...

<!-- Single Page Header End -->

<!-- Cart Page Start -->

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

<div class="container py-5">

<div class="table-responsive">

<table class="table">

...

<tbody>

@foreach (var item in Model)

{

<tr id="idProduct-@item.IdProduct">

<th scope="row">

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

<**img** src="~/images/SanPham/@item.Img" class="img-fluid me-5 rounded-circle" style="width: 80px; height: 80px;" alt="@item.Img">

</div>

</th>

<td>

<p class="mb-0 mt-4">@item.Name</p>

</td>

<td>

<p class="mb-0 mt-4">@item.Price.ToString("#,##0 VND")</p>

</td>

<td>

<div class="input-group quantity mt-4" style="width: 100px;">

<div class="input-group-btn">

<**a** **asp-controller**="cart" **asp-action**="ChangeQuantityCart"

**asp-route-id**="@item.IdProduct" **asp-route-isIncrement**="false"

class="btn btn-sm btn-minus rounded-circle bg-light border">

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

</**a**>

</div>

<input type="number"

class="form-control form-control-sm text-center border-0"

value="@item.Quantity">

<div class="input-group-btn">

<**a** **asp-controller**="cart" **asp-action**="ChangeQuantityCart"

**asp-route-id**="@item.IdProduct" **asp-route-isIncrement**="true"

class="btn btn-sm btn-plus rounded-circle bg-light border">

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

</**a**>

</div>

</div>

</td>

<td>

<p class="mb-0 mt-4">@item.Total.ToString("#,##0 VND")</p>

</td>

<td>

<**a** **asp-controller**="cart" **asp-action**="RemoveCart"

**asp-route-id**="@item.IdProduct" class="btn btn-md rounded-circle bg-light border mt-4">

<i class="fa fa-times text-danger"></i>

</**a**>

</td>

</tr>

}

</tbody>

</table>

</div>

<div class="row g-4 justify-content-end">

<div class="col-8"></div>

<div class="col-sm-8 col-md-7 col-lg-6 col-xl-4">

<div class="bg-light rounded">

<div class="py-4 mb-4 border-top border-bottom d-flex justify-content-between">

<h5 class="mb-0 ps-4 me-4">Tổng tiền</h5>

<p class="mb-0 pe-4">@Model.Sum(p => p.Total).ToString("#,##0 VND")</p>

</div>

<button class="btn border-secondary rounded-pill px-4 py-3 text-primary text-uppercase mb-4 ms-4" type="button">Thanh toán</button>

</div>

</div>

</div>

</div>

</div>

<!-- Cart Page End -->

### Hiển thị số lượng sản phẩm trong giỏ hàng

Tạo ViewComponent để hiển thị số lượng trong giỏ hàng

ViewComponents/CartViewComponent.cs

using Microsoft.AspNetCore.Mvc;

using ShopManager.Helper;

using ShopManager.Models;

namespace ShopManager.ViewComponents

{

public class CartViewComponent: ViewComponent

{

public IViewComponentResult Invoke()

{

// Lấy danh sách giỏ h

var cart = HttpContext.Session.Get<List<CartItem>>(MyConst.CART\_KEY)

?? new List<CartItem>();

return View(new CartModel()

{

Quantity = cart.Sum(p => p.Quantity),

Total = cart.Sum(p => p.Total)

});

}

}

}

Tại Folder Views/Shared/Components, tạo Folder mới: Cart

Tạo View mới: **Default.cshtml**

**Views/Shared/Components/Default.cshtml**

@model ShopManager.Models.CartModel

<**a** **asp-controller**="Cart" **asp-action**="Index" 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;">

@Model.Quantity

</span>

</**a**>

Hiển thị CartViewComponent

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

@model IEnumerable<ShopManager.Models.NavbarItem>

<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">

@if (@Context.User.Identity.IsAuthenticated)

{

...

}

else

{

...

}

</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)

{

...

}

</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> \*@~~

@await Component.InvokeAsync("Cart")

<**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>

## Thanh toán

Tại Folder DAL, tại class mới: CartDAL.cs

**DAL/CartDAL.cs**

using Microsoft.Data.SqlClient;

using ShopManager.Database;

using ShopManager.Models;

namespace ShopManager.DAL

{

public class CartDAL

{

DBConnect connect = new DBConnect();

public bool **CheckOut**(Customer customer, List<CartItem> cart)

{

connect.openConnection();

bool insertPaymentSuccess = false;

bool CheckOutSuccess = true;

int? SCOPE\_IDENTITY = null;

int TOTAL\_CART = cart.Sum(p => p.Total);

// Insert vào bảng Payment và lấy Id được tạo tự động sau khi insert

using (SqlCommand command = new SqlCommand())

{

command.Connection = connect.getConnecttion();

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

string query = @"

insert into payment (customerId, firstName, lastName, phone, email, createAt, total)

values (@customerId, @firstName, @lastName, @phone, @email, GETDATE(), @total);

SELECT SCOPE\_IDENTITY() AS SCOPE\_IDENTITY;

";

command.CommandText = query;

command.Parameters.AddWithValue("@customerId", customer.Id);

command.Parameters.AddWithValue("@firstName", customer.FirstName);

command.Parameters.AddWithValue("@lastName", customer.LastName);

command.Parameters.AddWithValue("@phone", customer.Phone);

command.Parameters.AddWithValue("@email", customer.Email);

command.Parameters.AddWithValue("@total", TOTAL\_CART);

SqlDataReader reader = command.ExecuteReader();

while (reader.Read())

{

insertPaymentSuccess = true;

SCOPE\_IDENTITY = reader["SCOPE\_IDENTITY"] == DBNull.Value ? null : Convert.ToInt32(reader["SCOPE\_IDENTITY"]);

}

}

connect.closeConnection();

//Nếu InSert vào bảng Payment không thành công thì trả về false và dừng lại

if (!insertPaymentSuccess || SCOPE\_IDENTITY == null)

{

return false;

}

// Insert vào bảng Payment Detail với Id tự động vừa lấy được (SCOPE\_IDENTITY)

foreach (var itemCart in cart)

{

bool success = **InsertToPaymentDetail**(SCOPE\_IDENTITY ?? 0, itemCart);

if (!success)

{

CheckOutSuccess = false;

}

}

return CheckOutSuccess;

}

// Insert vào bảng Payment Detail

public bool **InsertToPaymentDetail**(int SCOPE\_IDENTITY, CartItem itemCart)

{

connect.openConnection();

int numberOfRows = 0;

using (SqlCommand command = new SqlCommand())

{

command.Connection = connect.getConnecttion();

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

string query = @"

insert into paymentDetail (paymentId, productId, price, quantity, total, createAt)

values (@paymentId, @productId, @price, @quantity, @total, GETDATE()) ";

command.CommandText = query;

command.Parameters.AddWithValue("@paymentId", SCOPE\_IDENTITY);

command.Parameters.AddWithValue("@productId", itemCart.IdProduct);

command.Parameters.AddWithValue("@price", itemCart.Price);

command.Parameters.AddWithValue("@quantity", itemCart.Quantity);

command.Parameters.AddWithValue("@total", itemCart.Total);

numberOfRows = command.ExecuteNonQuery();

}

connect.closeConnection();

return numberOfRows > 0;

}

}

}

**Controllers/CartController.cs**

using Microsoft.AspNetCore.Authorization;

using Microsoft.AspNetCore.Mvc;

using ShopManager.DAL;

using ShopManager.Helper;

using ShopManager.Models;

using System.Runtime.CompilerServices;

using System.Security.Claims;

namespace ShopManager.Controllers

{

public class CartController : Controller

{

//Khai báo DAL

ProductDAL productDAL = new ProductDAL();

CustomerDAL customerDAL = new CustomerDAL();

CartDAL cartDAL = new CartDAL();

//Lấy thông tin giỏ hàng

public List<CartItem> Cart => HttpContext.Session.Get<List<CartItem>>(MyConst.CART\_KEY) ??

new List<CartItem>();

//Hiển thị Giỏ hàng

public IActionResult Index()

{

return View(Cart);

}

//Thêm vào giỏ hàng

public IActionResult AddToCart(int id, int quantity = 1)

{

...

}

public IActionResult RemoveCart(int id)

{

...

}

public IActionResult InputQuantityCart(int id, int quantity = 1)

{

...

}

[Authorize] //Kiểm tra user đăng nhập ?

public IActionResult CheckOut()

{

try

{

//Không thanh toán khi không có sản phẩm trong giỏ hàng

if (Cart.Count() == 0)

{

TempData["CheckOutErrorMessage"] = "Thanh toán thất bại";

return RedirectToAction("Index");

}

// Lấy Customer id nếu người dùng đã đăng nhập

string? idTam = null;

if (HttpContext.User.FindFirstValue("CustomerId") != null)

{

idTam = HttpContext.User.FindFirstValue("CustomerId");

}

if (idTam == null)

{

return Redirect("/Customer/SignIn");

}

//Lấy thông tin User

int idCustomer = Convert.ToInt32(idTam!);

Customer? customer = new Customer();

customer = customerDAL.GetCustomerById(idCustomer);

// Nếu không tìm thấy User thì trả về trang 404 - Not Found

if (customer == null)

{

return Redirect("/404");

}

//Insert Vào Database

bool isSuccess = cartDAL.CheckOut(customer, Cart);

if (isSuccess)

{

//Insert Database thành công

TempData["CheckOutSuccessMessage"] = "Thanh toán thành công";

var gioHang = Cart;

gioHang = new List<CartItem>();

HttpContext.Session.Set(MyConst.CART\_KEY, gioHang);

}

else

{

//Insert Database thất bại

TempData["CheckOutErrorMessage"] = "Thanh toán thất bại";

}

return RedirectToAction("Index");

}

catch (Exception ex)

{

Console.WriteLine(ex.Message.ToString());

TempData["CheckOutErrorMessage"] = "Lỗi hệ thống: "+ ex.Message.ToString();

return RedirectToAction("Index");

}

}

}

}

**Views/Cart/Index.cshtml**

@model IEnumerable<ShopManager.Models.CartItem>

<!-- Single Page Header start -->

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

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

</ol>

</div>

<!-- Single Page Header End -->

<!-- Cart Page Start -->

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

<div class="container py-5">

<!-- Alert Success Start -->

@if (!string.IsNullOrWhiteSpace(TempData["CheckOutSuccessMessage"]?.ToString()))

{

<div class="w-100 d-flex justify-content-end">

<div class="alert alert-success alert-dismissible fade show r-0 mr-5" role="alert" id="alert\_message" style="width:40%">

<strong>Success!</strong> @TempData["CheckOutSuccessMessage"]?.ToString()

<button type="button" class="btn-close" data-dismiss="alert" aria-label="Close">

</button>

</div>

</div>

}

<!-- Alert Success End -->

<!-- Alert Error Start -->

@if (!string.IsNullOrWhiteSpace(TempData["CheckOutErrorMessage"]?.ToString()))

{

<div class="w-100 d-flex justify-content-end">

<div class="alert alert-danger alert-dismissible fade show r-0 mr-5" role="alert" id="alert\_message" style="width:40%">

<strong>Success!</strong> @TempData["CheckOutErrorMessage"]?.ToString()

<button type="button" class="btn-close" data-dismiss="alert" aria-label="Close">

</button>

</div>

</div>

}

<!-- Alert Error End -->

<div class="table-responsive">

<table class="table">

<thead>

<tr>

<th scope="col">Products</th>

<th scope="col">Name</th>

<th scope="col">Price</th>

<th scope="col">Quantity</th>

<th scope="col">Total</th>

<th scope="col">Handle</th>

</tr>

</thead>

<tbody>

@foreach (var item in Model)

{

<tr id="idProduct-@item.IdProduct">

<th scope="row">

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

<**img** src="~/images/SanPham/@item.Img" class="img-fluid me-5 rounded-circle" style="width: 80px; height: 80px;" alt="@item.Img">

</div>

</th>

<td>

<p class="mb-0 mt-4">@item.Name</p>

</td>

<td>

<p class="mb-0 mt-4">@item.Price.ToString("#,##0 VND")</p>

</td>

<td>

<div class="input-group quantity mt-4" style="width: 100px;">

<div class="input-group-btn">

<**a** **asp-controller**="cart" **asp-action**="ChangeQuantityCart"

**asp-route-id**="@item.IdProduct" **asp-route-isIncrement**="false"

class="btn btn-sm btn-minus rounded-circle bg-light border">

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

</**a**>

</div>

<input type="number"

class="form-control form-control-sm text-center border-0"

value="@item.Quantity">

<div class="input-group-btn">

<**a** **asp-controller**="cart" **asp-action**="ChangeQuantityCart"

**asp-route-id**="@item.IdProduct" **asp-route-isIncrement**="true"

class="btn btn-sm btn-plus rounded-circle bg-light border">

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

</**a**>

</div>

</div>

</td>

<td>

<p class="mb-0 mt-4">@item.Total.ToString("#,##0 VND")</p>

</td>

<td>

<**a** **asp-controller**="cart" **asp-action**="RemoveCart"

**asp-route-id**="@item.IdProduct" class="btn btn-md rounded-circle bg-light border mt-4">

<i class="fa fa-times text-danger"></i>

</**a**>

</td>

</tr>

}

</tbody>

</table>

</div>

<div class="row g-4 justify-content-end">

<**form** **asp-controller**="Cart" **asp-action**="CheckOut" class="row g-4 justify-content-end">

<div class="col-8"></div>

<div class="col-sm-8 col-md-7 col-lg-6 col-xl-4">

<div class="bg-light rounded">

<div class="py-4 mb-4 border-top border-bottom d-flex justify-content-between">

<h5 class="mb-0 ps-4 me-4">Tổng tiền</h5>

<p class="mb-0 pe-4">@Model.Sum(p => p.Total).ToString("#,##0 VND")</p>

</div>

<button class="btn border-secondary rounded-pill px-4 py-3 text-primary text-uppercase mb-4 ms-4" type="submit" >Thanh toán</button>

</div>

</div>

</**form**>

</div>

</div>

</div>

<!-- Cart Page End -->

<script>

// ------------- Auto Close Alert ----------------

const alert\_error = document.getElementById("alert\_message")

window.setTimeout(function () {

$(".alert").fadeTo(500, 0).slideUp(500, function () {

$(this).remove();

});

}, 2000);

</script>

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