# giỏ hàng - P1

Lưu ý: Khi nhấn nút chọn vào giỏ hàng thì không lưu vào cơ sở dữ liệu

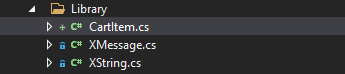
Lưu tạm trên trình duyệt (session)

Mua hàng, ta sẽ lưu:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Mã SP | Tên SP | Hình SP | Giá | Số lượng | Thành tiền |
| 12 | … |  |  |  | =Giá\*SL |
| 13 |  |  |  |  |  |
| 08 |  |  |  |  |  |

**Cách làm:**

## Tạo ra 1 đối tượng (class = Cart) trong Library



using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

namespace UDW.Library

{

public class CartItem

{

// cac truong thong tin duoc khai bao o day

public int ProductId { get; set; }

public string Name { get; set; }

public string Img { get; set; }

public decimal Price { get; set; }

public int Ammount { get; set; }

public decimal Total { get; set; }

}

}

## Tạo ra session

Tập tin Global.asax.cs thiết lập session cho giỏ hàng

//Tao moi Session

protected void Session\_Start()

{

Session["UserID"] = "1";

Session["MyCart"] = "";//khoi tao gia tri = NULL

}

## HOMEPRODUCT.cshtml

Khi ta click vào biểu tượng giỏ hàng ngay trên trang chủ (tất cả sản phẩm: PRODUCT.cshtml và Sản phẩm theo loại: HOMEPRODDUCT.cshtml) sẽ chuyển trạng thái về GiohangController.

Nơi cần gán link cho cả 2 trang giao diện ở trên:

<li><button><i class="ti-shopping-cart"></i></button></li>

Ta sẽ gán cho nó link (link này sẽ được liên kết trong GiohangController)

<li>

<a href="@Url.Action("AddCart","Giohang", new { productid [[1]](#footnote-1) = product.Id[[2]](#footnote-2)})">

<span class="custom-button"><i class="ti-shopping-cart"></i></span>

</a>

</li>

Nghĩa là sẽ chuyển đến Action = AddCart của Controller=Giohang

## Giohangcontroller

Thực hiện tạo mới Action = AddCart.

//////////////////////////////////////////////////////////////////

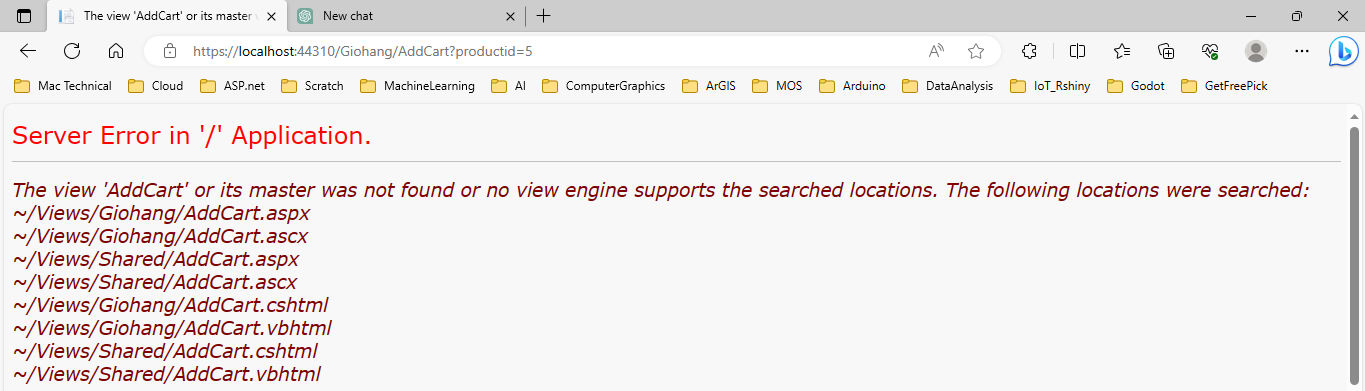
///Them vao gio hang

public ActionResult AddCart(int productid)

{

return View();

}



Tiếp theo ta sẽ truy vấn thông tin sản phẩm thông qua productid từ hàm getRow trong ProductsDAO

Products products = productsDAO.getRow(productid);

//////////////////////////////////////////////////////////////////

///Them vao gio hang

public ActionResult AddCart(int productid)

{

Products products = productsDAO.getRow(productid);

CartItem cartitem = new CartItem ();//tao moi danh sach gio hang

return View();

}

Để tạo mới danh sách giỏ hàng chúng ta cần truyền các thông số của Cart.cs

## Cart.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

namespace UDW.Library

{

public class CartItem

{

public int ProductId { get; set; }

public string Name { get; set; }

public string Img { get; set; }

public decimal Price { get; set; }

public int Ammount { get; set; }

public decimal Total { get; set; }

public CartItem ()

{

}

public CartItem (int proid, string name, string img, decimal price, int qty)

{

this.ProductId = proid;

this.Name = name;

this.Img = img;

this.Price = price;

this.Ammount = qty;

this.Total = price \* qty;

}

}

}

Sau đó ta truyền các thông số cho GiohangController như sau:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

using MyClass.DAO;

using MyClass.Model;

using UDW.Library;

namespace UDW.Controllers

{

public class GiohangController : Controller

{

ProductsDAO productsDAO = new ProductsDAO();

// GET: Cart

public ActionResult Index()

{

//da co thong tin trong gio hang, lay thong tin cua session -> ep kieu ve list

List<CartItem> list = (List<CartItem>)Session["MyCart"];

return View("Index",list);

}

//////////////////////////////////////////////////////////////////

///Them vao gio hang

public ActionResult AddCart(int productid)

{

Products products = productsDAO.getRow(productid);

CartItem cartitem = new CartItem (products.Id, products.Name, products.Image, products.Price,1);

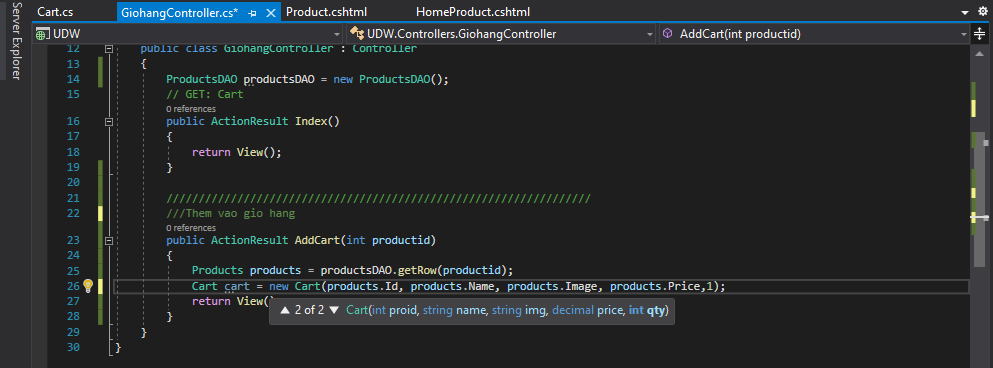
return View();

}

}

}

Chú ý thứ tự của các trường thông tin.



Xét các trường hợp của giỏ hàng như sau:

//////////////////////////////////////////////////////////////////

///Them vao gio hang

public ActionResult AddCart(int productid)

{

Products products = productsDAO.getRow(productid);

CartItem cartitem = new CartItem(products.Id, products.Name, products.Image, products.Price,1);

//Them vao gio hang voi danh sách list phan tu = Session = MyCart

if (Session["MyCart"].Equals(""))//session chua co gio hang

{

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

list.Add(cartitem);

Session["MyCart"] = list;

}

else

{

//da co thong tin trong gio hang, lay thong tin cua session -> ep kieu ve list

List<CartItem> list = (List<CartItem>) Session["MyCart"];

//them vao gio hang moi

list.Add(cartitem);

Session["MyCart"] = list;

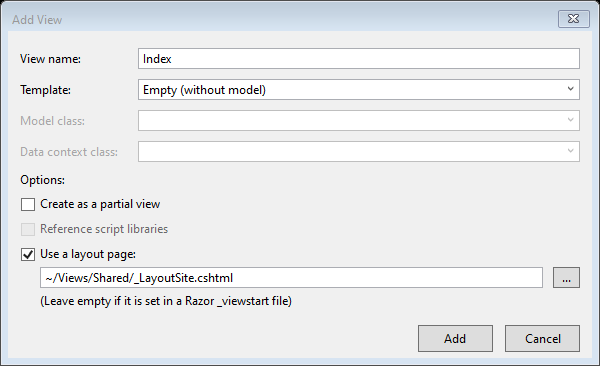
}

//chuyen huong trang

return RedirectToAction("Index","Giohang");

}

## Xây dựng trang INDEX cho Gio hang



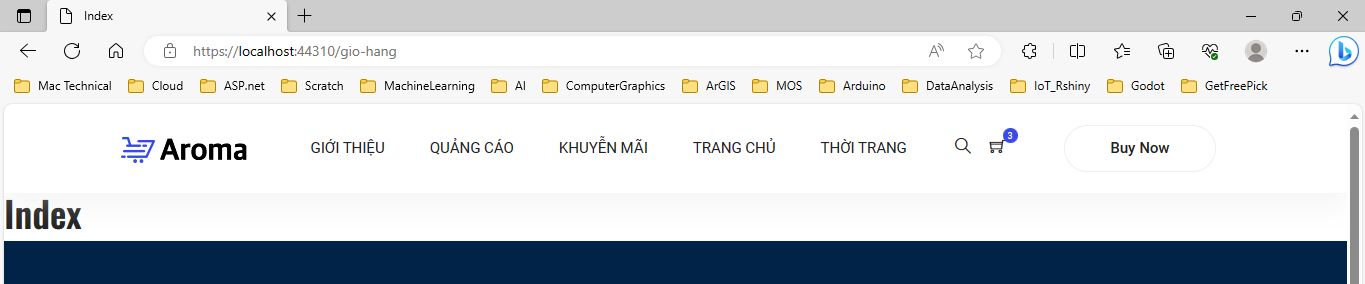
@{

ViewBag.Title = "Index";

Layout = "~/Views/Shared/\_LayoutSite.cshtml";

}

<h2>Index</h2>



Trước tiên, ta nhận thấy dữ liệu truyền từ GiohangController sang có dạng danh sách (List)

public ActionResult Index()

{

//da co thong tin trong gio hang, lay thong tin cua session -> ep kieu ve list

List<CartItem> list = (List<CartItem>)Session["MyCart"];

return View("Index",list);

}

Nên ta sẽ truyền dưới dạng model cho trang Index.cshtml

@model IEnumerable<UDW.Library.CartItem>

@{

ViewBag.Title = "Giỏ hàng";

Layout = "~/Views/Shared/\_LayoutSite.cshtml";

}

<!-- Content Wrapper. Contains page content -->

<div class="content-wrapper">

<!-- Main content -->

<section class="content my-2">

<!-- Default box -->

<div class="card">

<div class="card-header">

<div class="row">

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

<strong class="text-danger">DANH SÁCH GIỎ HÀNG</strong>

</div>

</div>

</div>

<div class="card-body">

<table class="table table-bordered table-striped" id="myTable">

<thead>

<tr>

<th>Sản phẩm</th>

<th style="width:50px;">Hình ảnh</th>

<th>Số lượng</th>

<th>Đơn giá</th>

<th>Thành tiền</th>

<th></th>

</tr>

</thead>

<tbody>

@foreach (var item in Model)

{

<tr>

<td>@item.Name </td>

<td><imgsrc="~/Public/img/product/@item.Img"alt="Img" class="img-fluid"/></td>

<td class="card-product\_\_price">@item.Ammount</td>

<td class="card-product\_\_price">@string.Format("{0:0,0}", @item.Price)<sup>đ</sup></td>

<td class="card-product\_\_price">@string.Format("{0:0,0}", @item.Total)<sup>đ</sup></td>

</tr>

}

</tbody>

</table>

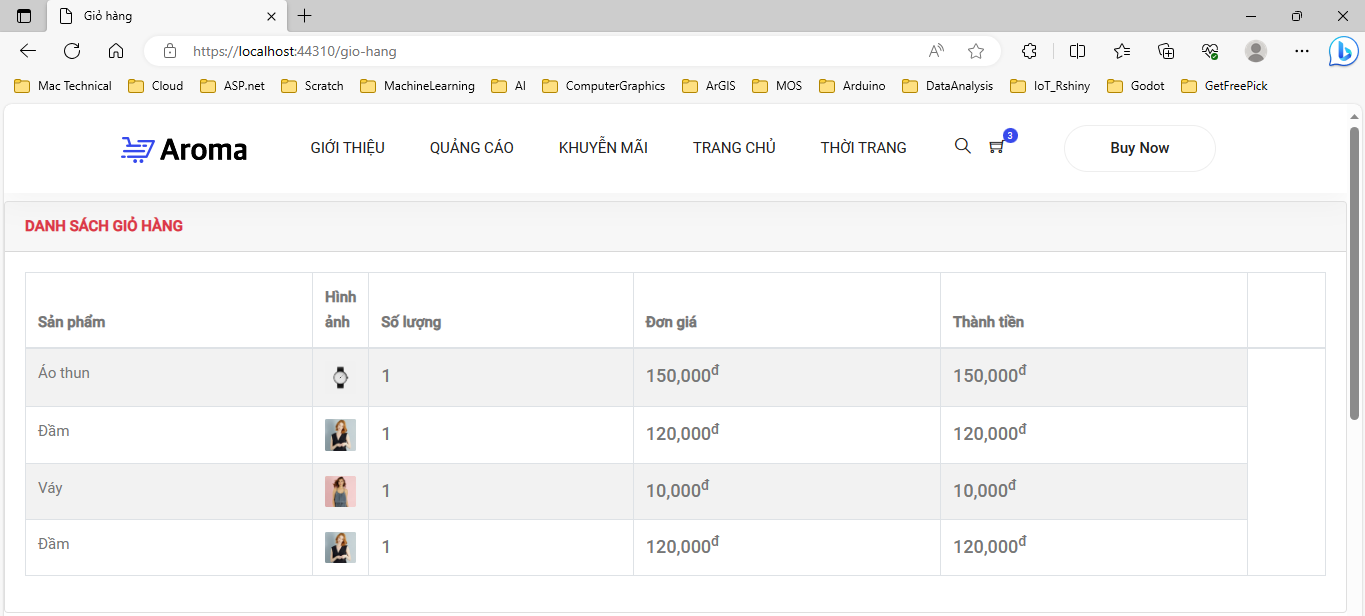
</div><!-- /.card-body -->

</div><!-- /.card -->

</section><!-- /.content -->

</div><!-- /.content-wrapper -->

Khi click vào biểu tượng giỏ hàng, chúng ta có được



Vấn đề số lượng, nếu ta chọn nhiều lần vào cùng một sản phẩm thì sản phẩm tự động tăng số lượng. Chúng ta thực hiện trong GiohangController.

Ngay bên trên đoạn mã:

//them vao gio hang moi

list.Add(cartitem);

Session["MyCart"] = list;

Chúng ta cần kiểm tra điều kiện xem thử mã của sản phẩm này đã tồn tại hay chưa?

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

using MyClass.DAO;

using MyClass.Model;

using UDW.Library;

namespace UDW.Controllers

{

public class GiohangController : Controller

{

ProductsDAO productsDAO = new ProductsDAO();

// GET: Cart

public ActionResult Index()

{

//da co thong tin trong gio hang, lay thong tin cua session -> ep kieu ve list

List<CartItem> list = (List<CartItem>)Session["MyCart"];

return View("Index", list);

}

//////////////////////////////////////////////////////////////////

///Them vao gio hang

public ActionResult AddCart(int productid)

{

Products products = productsDAO.getRow(productid);

CartItem cartitem = new CartItem(products.Id, products.Name, products.Image, products.Price, 1);

//Them vao gio hang voi danh sách list phan tu = Session = MyCart

if (Session["MyCart"].Equals(""))//session chua co gio hang

{

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

list.Add(cartitem);

Session["MyCart"] = list;

}

else

{

//da co thong tin trong gio hang, lay thong tin cua session -> ep kieu ve list

List<CartItem> list = (List<CartItem>)Session["MyCart"];

//kiewm tra productid da co trong danh sach hay chua

int count = list.Where(m => m.ProductId == productid).Count();

if (count > 0)//da co trong danh sach gio hang truoc do

{

cartitem.Ammount += 1;

//cap nhat lai danh sach

int vt = 0;

foreach (var item in list)

{

if (item.ProductId == productid)

{

list[vt].Ammount += 1;

}

vt++;

}

Session["MyCart"] = list;

}

else

{

//them vao gio hang moi

list.Add(cartitem);

Session["MyCart"] = list;

}

}

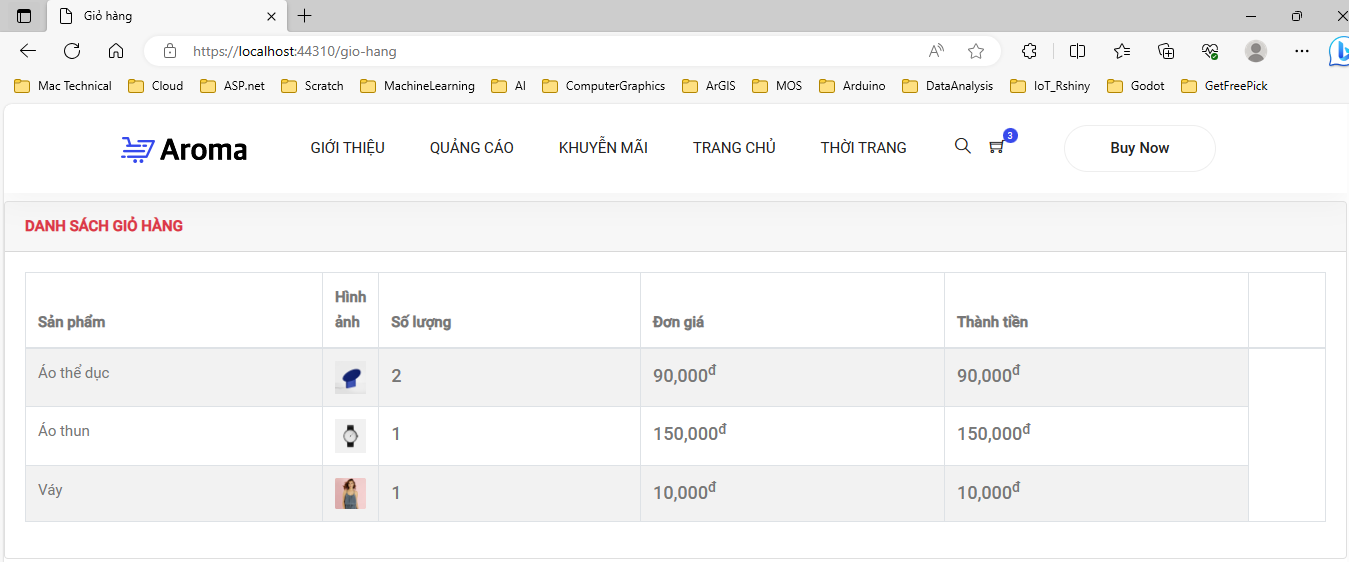
//chuyen huong trang

return RedirectToAction("Index", "Giohang");

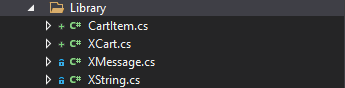
}

}

}



Để chuẩn bị cho các bài sau có sử dụng lại đoạn mã thêm vào giỏ hàng, ta sẽ tạo ra một thư viện Xcart.cs cho tiện sử dụng sau này.



using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

namespace UDW.Library

{

public class XCart

{

List<CartItem> list;

public List<CartItem> AddCart(CartItem cartitem, int productid)

{

if(System.Web.HttpContext.Current.Session["MyCart"].Equals(""))

//session chua co gio hang

{

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

list.Add(cartitem);

System.Web.HttpContext.Current.Session["MyCart"] = list;

}

else

{

//da co thong tin trong gio hang, lay thong tin cua session -> ep kieu ve list

List<CartItem> list = (List<CartItem>)System.Web.HttpContext.Current.Session["MyCart"];

//kiewm tra productid da co trong danh sach hay chua

int count = list.Where(m => m.ProductId == productid).Count();

if (count > 0)//da co trong danh sach gio hang truoc do

{

cartitem.Ammount += 1;

//cap nhat lai danh sach

int vt = 0;

foreach (var item in list)

{

if (item.ProductId == productid)

{

list[vt].Ammount += 1;

}

vt++;

}

System.Web.HttpContext.Current.Session["MyCart"] = list;

}

else

{

//them vao gio hang moi

list.Add(cartitem);

System.Web.HttpContext.Current.Session["MyCart"] = list;

}

}

return list;

}

public void UpdateCart()

{

}

public void DelCart()

{

}

public List<CartItem> GetCart()

{

if (System.Web.HttpContext.Current.Session["MyCart"].Equals(""))

{

return null;

}

return (List<CartItem>)System.Web.HttpContext.Current.Session["MyCart"];

}

}

}

Do đó, bên phía GiohangController chúng ta có sự thay đổi

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

using MyClass.DAO;

using MyClass.Model;

using UDW.Library;

namespace UDW.Controllers

{

public class GiohangController : Controller

{

ProductsDAO productsDAO = new ProductsDAO();

XCart xcart = new XCart();

// GET: Cart

public ActionResult Index()

{

//da co thong tin trong gio hang, lay thong tin cua session -> ep kieu ve list

List<CartItem> list = xcart.GetCart();

return View("Index", list);

}

//////////////////////////////////////////////////////////////////

///Them vao gio hang

public ActionResult AddCart(int productid)

{

Products products = productsDAO.getRow(productid);

CartItem cartitem = new CartItem(products.Id, products.Name, products.Image, products.Price, 1);

//Them vao gio hang voi danh sách list phan tu = Session = MyCart

XCart xcart = new XCart();

List<CartItem> list = xcart.AddCart(cartitem, productid);

//chuyen huong trang

return RedirectToAction("Index", "Giohang");

}

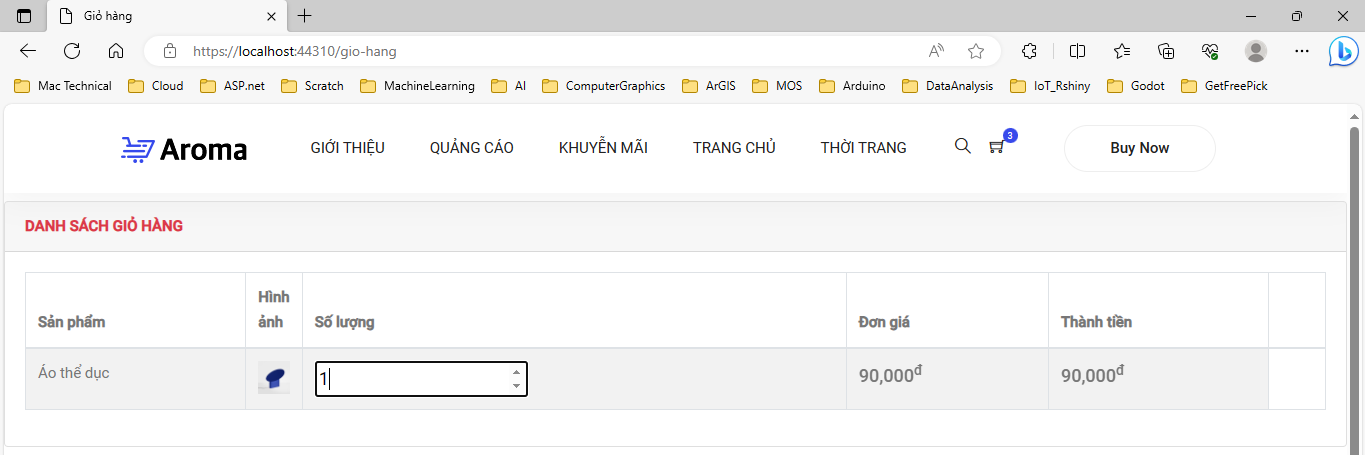
}

}



Tiếp theo ta thiết kế nút cho ô số lượng, thay vì người dùng mỗi lần muốn tăng số lượng sẽ chọn lại sản phẩm, chúng ta thiết kế nút tăng, giảm.

Thêm vào đó, Thành tiền cũng tự động thay đổi theo



### Index.cshtml

<tbody>

@foreach (var item in Model)

{

<tr>

<td>@item.Name </td>

<td><img src="~/Public/img/product/@item.Img" alt="Img" class="img-fluid" /></td>

<td class="card-product\_\_price"><input name="amount" value="@item.Ammount" type="number" min="0"/></td>

<td class="card-product\_\_price">@string.Format("{0:0,0}", @item.Price)<sup>đ</sup></td>

<td class="card-product\_\_price">@string.Format("{0:0,0}", @item.Total)<sup>đ</sup></td>

</tr>

}

</tbody>

### XCart.cs

else

{

//da co thong tin trong gio hang, lay thong tin cua session -> ep kieu ve list

List<CartItem> list = (List<CartItem>)System.Web.HttpContext.Current.Session["MyCart"];

//kiewm tra productid da co trong danh sach hay chua

int count = list.Where(m => m.ProductId == productid).Count();

if (count > 0)//da co trong danh sach gio hang truoc do

{

cartitem.Ammount += 1;

//cap nhat lai danh sach

int vt = 0;

foreach (var item in list)

{

if (item.ProductId == productid)

{

list[vt].Ammount += 1;

list[vt].Total = list[vt].Ammount \* list[vt].Price;

}

vt++;

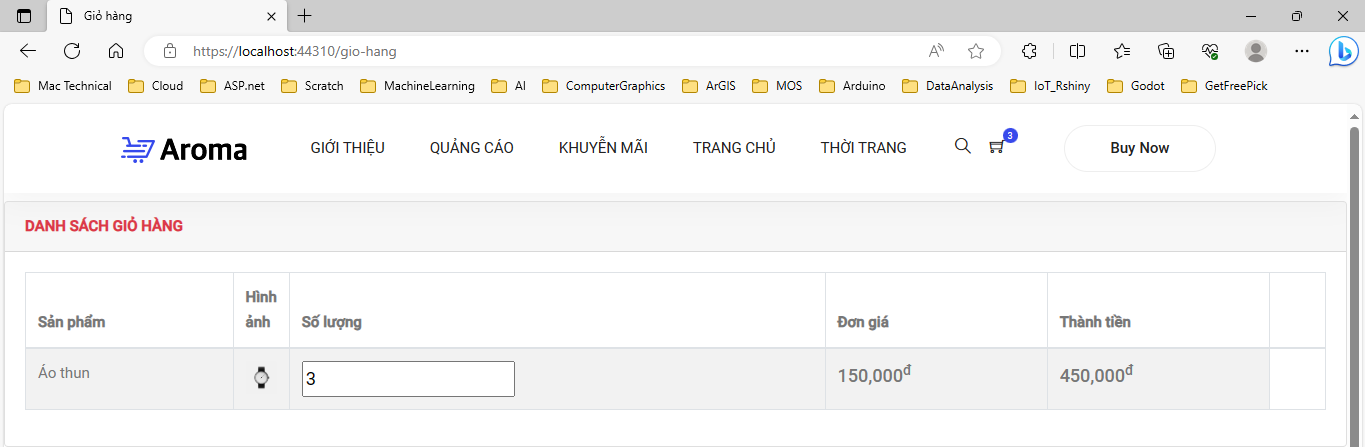
}

System.Web.HttpContext.Current.Session["MyCart"] = list;

}

…

}



Tiếp theo, ta sẽ thiết kế chức năng hiển thị tổng số tiền cần thanh toán

### Index.cshtml

<tbody>

@{ decimal tongtien = 0;}

@foreach (var item in Model)

{

<tr>

<td>@item.Name </td>

<td><img src="~/Public/img/product/@item.Img" alt="Img" class="img-fluid" /></td>

<td class="card-product\_\_price"><input name="amount" value="@item.Ammount" type="number" min="0"/></td>

<td class="card-product\_\_price">@string.Format("{0:0,0}", @item.Price)<sup>đ</sup></td>

<td class="card-product\_\_price">@string.Format("{0:0,0}", @item.Total)<sup>đ</sup></td>

</tr>

tongtien += item.Total;

}

<tr>

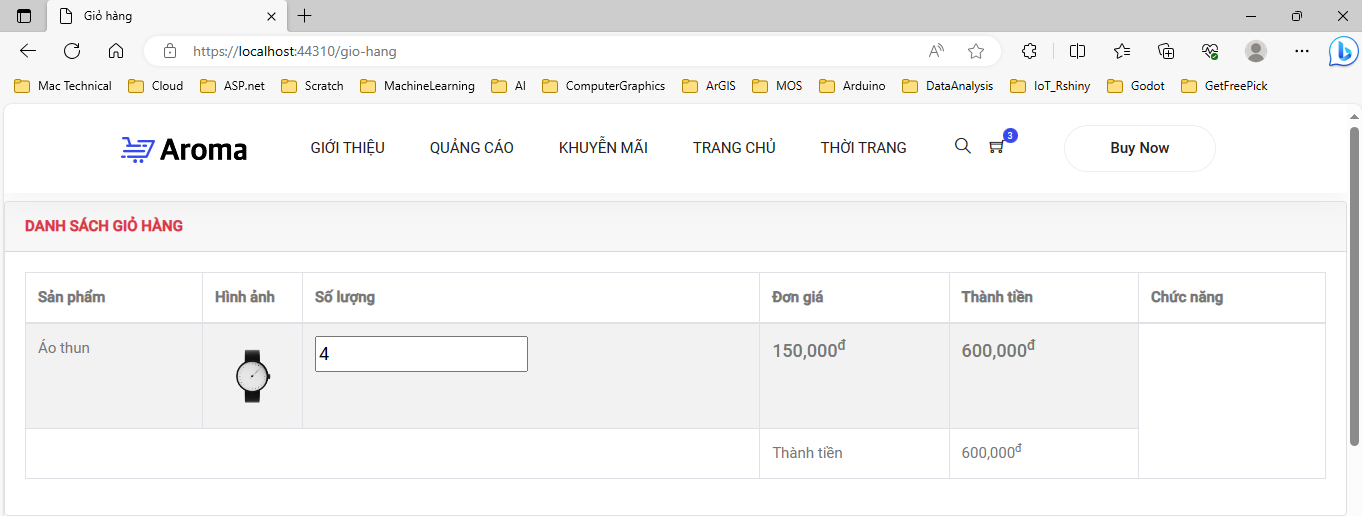
<td colspan="3"></td>

<td>Thành tiền</td>

<td colspan="1">@string.Format("{0:0,0}", tongtien)<sup>đ</sup></td>

</tr>

</tbody>



1. Tham số đầu vào của hàm AddCart trong GiohangController [↑](#footnote-ref-1)
2. Biến của trang cshtml được lấy ra từ bảng product [↑](#footnote-ref-2)