Lớp ShoppingCart.cs

Cài đặt các chức năng của giỏ hàng

public class ShoppingCart

{

// internal member variables

private String user;

private DataTable items;

private Double total;

// public properties

public String UserID

{

get { return user; }

set { user = value; }

}

public DataTable CartItems

{

get { return items; }

set { items = value; }

}

public Double TotalValue

{

get { return total; }

set { total = value; }

}

// default constructor

public ShoppingCart()

{

// create an empty shopping cart

user = String.Empty;

// create an empty DataTable to hold the cart items

items = new DataTable("Items");

items.Columns.Add(new DataColumn("ProductID", Type.GetType("System.String")));

items.Columns.Add(new DataColumn("Quantity", Type.GetType("System.Int32")));

items.Columns.Add(new DataColumn("Price", Type.GetType("System.Double")));

// set total value of the new cart to zero

total = 0;

}

public void UpdateCart()

{

//your Update cart

}

public void RemoveItem(string productID)

{

DataRow[] rows = items.Select("ProductID='" + productID + "'");

if (rows.Length > 0)

{

items.Rows.Remove(rows[0]);

}

}

// add an item to the cart

public void AddItem(String proId, Int32 Qty, Double price)

{

if (!IsExistItem(proId))

{

// create new DataTable row and populate with values - Hàng mới

DataRow row = items.NewRow();

row["ProductID"] = proId;

row["Quantity"] = Qty;

row["Price"] = price;

// add row to DataTable update total value

items.Rows.Add(row);

total += (Qty \* price);

}

else

{

//tăng số lượng hàng trong giỏ hàng

for (int i = 0; i < items.Rows.Count; i++)

{

if (items.Rows[i]["ProductID"].Equals(proId))

{

items.Rows[i]["Quantity"] = int.Parse(items.Rows[i]["Quantity"].ToString()) + Qty;

total += (Qty \* price);

break;//Thoat vong lap

}

}

}

}

public bool IsExistItem(string proID)

{

bool b = false;

if (items.Rows.Count > 0)

{

for (int i = 0; i < items.Rows.Count; i++)

{

if (items.Rows[i]["ProductID"].Equals(proID))

{

b = true;

break;//Thoat vong lap

}

}

}

return b;

}

// empty the cart by clearing the DataTable

public void Clear()

{

items.Rows.Clear();

total = 0;

}

}

HomeController.cs

public class HomeController : Controller

{

public ActionResult Index(int? id)

{

Models.NorthwindEntities db = new Models.NorthwindEntities();

CatProViewModel cp = new CatProViewModel();

if(id.HasValue){

var products = from p in db.Products

where p.Category.CategoryID == id

select p;

cp.Products = products.ToList<Product>();

cp.Categories = db.Categories.ToList<Category>();

}else{

cp.Products = db.Products.ToList<Product>();

cp.Categories = db.Categories.ToList<Category>();

}

return View(cp);

}

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

{

ShoppingCart cart = (ShoppingCart)Session["cart"];

if (cart == null)

{

cart = new ShoppingCart();

}

cart.AddItem(id.ToString(), quantity, 300);

//Save cart

Session["cart"] = cart;

return RedirectToAction("YourCart");

}

public ActionResult YourCart()

{

//Models.NorthwindEntities db = new Models.NorthwindEntities();

//return View(db.Categories.Include("Products").ToList());

return View();

}

public ActionResult Category(int? id)

{

if(!id.HasValue){

return RedirectToAction("Index");

}

Models.NorthwindEntities db = new Models.NorthwindEntities();

var categories = from c in db.Categories.Include("Products")

where c.CategoryID==id

select c;

return View(categories.ToList());

}

public ActionResult RemoveItem(int id)

{

ShoppingCart cart = (ShoppingCart)Session["cart"];

if (cart == null)

{

cart = new ShoppingCart();

}

cart.RemoveItem(id.ToString());

return RedirectToAction("YourCart");

}

public ActionResult About()

{

ViewBag.Message = "Your application description page.";

return View();

}

public ActionResult Contact()

{

ViewBag.Message = "Your contact page.";

return View();

}

}

Views:

Index.cshtml – Hiển thị các sản phẩm

@model ShoppingCart.Models.CatProViewModel

@{

ViewBag.Title = "Index";

}

<h2>Index</h2>

<div class="row">

@{

var products = Model.Products;

foreach (var p in products)

{

<div class="col-md-4">

<div class="row">

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

<img src="~/Content/images/ms.png" style="width:140%;" />

</div>

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

Product ID: @p.ProductID <br />

Product ID: @p.ProductName <br />

Product ID: @p.UnitPrice <br />

Product ID: @p.Category.CategoryName <br />

<p>

@Html.ActionLink("Add to cart", "AddToCart", "Home",

new { @Id = @p.ProductID }, new { @class = "btn btn-primary" })

</p>

</div>

</div>

</div>

}

}

</div>

Yourcart.cshtml – HIển thị các sản phẩm trong giỏ hàng

@{

ViewBag.Title = "YourCart";

}

<h2>YourCart</h2>

@{ var cart = Session["cart"] as ShoppingCart.Controllers.ShoppingCart;}

<table>

<tr>

<td>Product ID</td>

<td>Quantity</td>

<td>Price</td>

<td>Action</td>

</tr>

@if (cart != null)

{

for (int i = 0; i < cart.CartItems.Rows.Count; i++)

{

<tr>

<td>@cart.CartItems.Rows[i][0].ToString()</td>

<td>@cart.CartItems.Rows[i][1].ToString()</td>

<td>@cart.CartItems.Rows[i][2].ToString()</td>

<td>@Html.ActionLink("Delete", "RemoveItem", "Home",

new { @Id = cart.CartItems.Rows[i][0].ToString() },

new { @class = "btn btn-primary" })</td>

</tr>

}

}

</table>