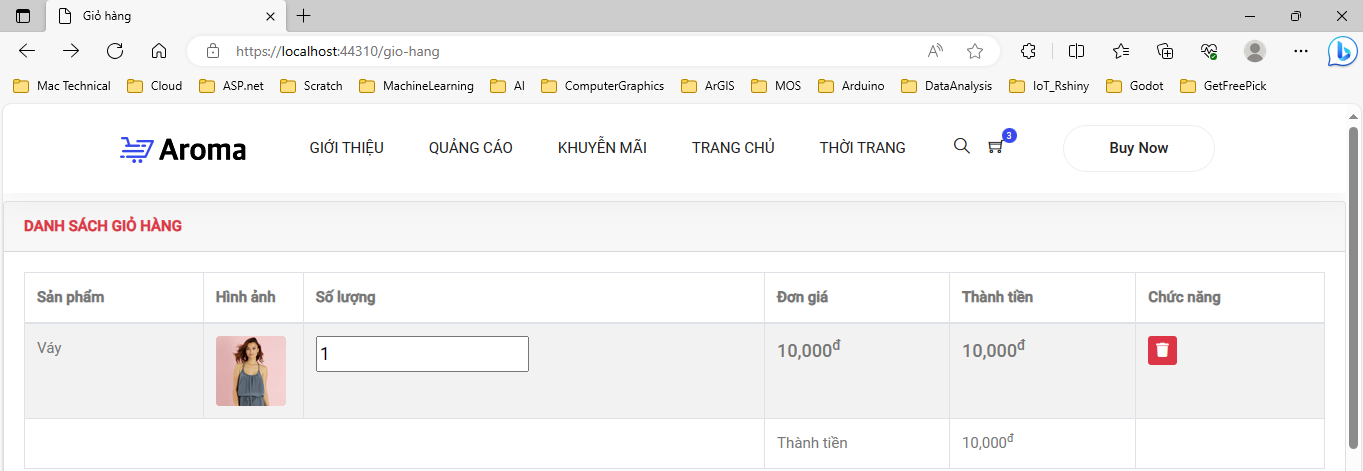
# giỏ hàng - THÊM VÀO GIỎ HÀNG

## Thiết kế các nút chức năng



Chức năng Xóa sản phẩm trong giỏ hàng

### Index.cshtml

@if (Model != null)

{

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

<thead>

<tr>

<th>Sản phẩm</th>

<th style="width:100px;">Hình ảnh</th>

<th>Số lượng</th>

<th>Đơn giá</th>

<th>Thành tiền</th>

<th>Chức năng</th>

</tr>

</thead>

<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>

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

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

<td><a href="@Url.Action("DelCart","Giohang", new {productid = @item.ProductId})" class="btn-sm btn-danger"> <i class="fa fa-trash"></i></a></td>[[1]](#footnote-1)

</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>

</table>

}

### GiohangController.cs

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

///CartDel

public ActionResult DelCart(int productid)

{

xcart.DelCart(productid);

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

}

Do vây, trong XCart.cs chúng ta cần truyền tham số productid cho hàm DelCart()

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

///DelCart

public void DelCart(int productid)

{

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

{

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

int vt = 0;

foreach (var item in list)

{

if (item.ProductId == productid)

{

list.RemoveAt(vt);

break;

}

vt++;

}

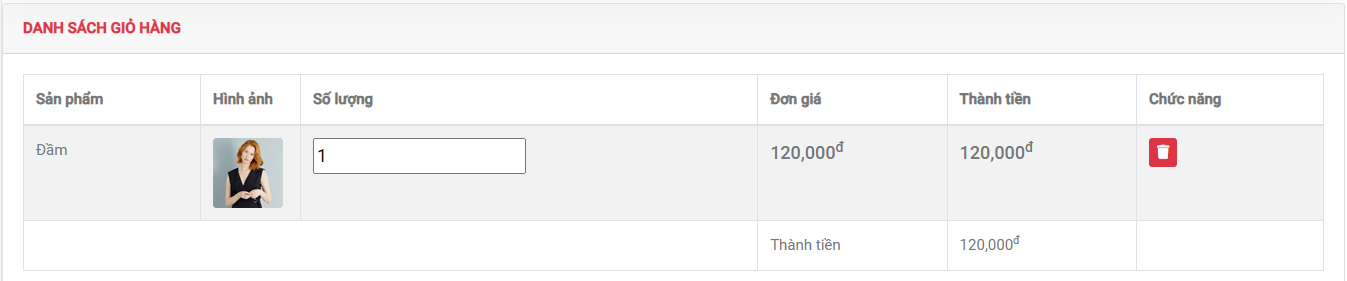
//cap nhat lai gio hang

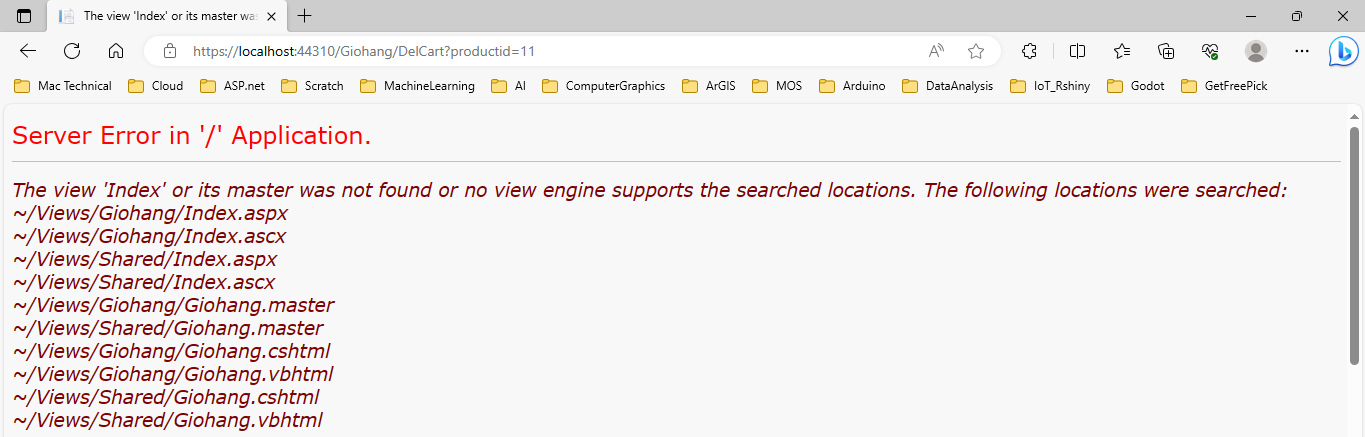
System.Web.HttpContext.Current.Session["MyCart"] = list;

}

}

# Thử nghiệm





Trang Index chưa trỏ về được

## Thêm chức năng mua thêm

### Index.cshtml

<tr>

<td colspan="3">

<a href="@Url.Action("Index","Site")" class="btn-sm btn-success"> <i class="fa fa-plus"></i>Mua thêm sản phẩm

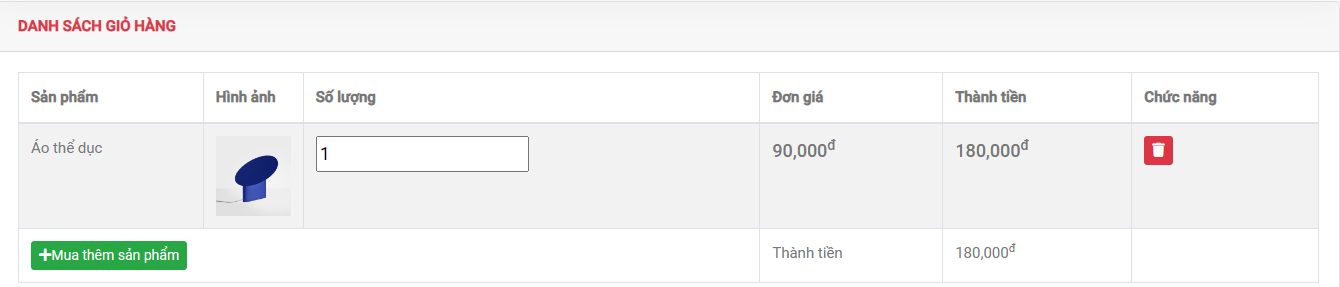
</a>

</td>

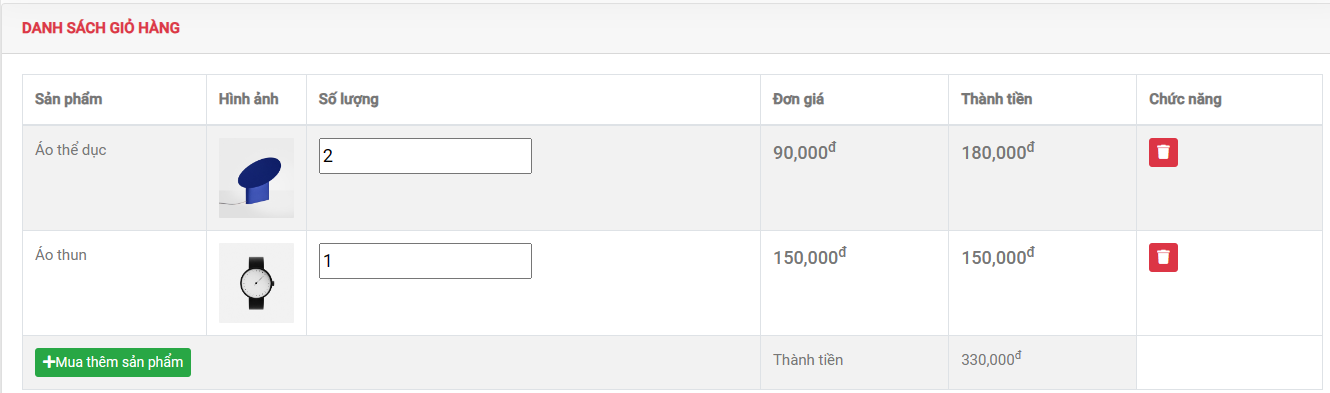
<td>Thành tiền</td>

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

</tr>



Khi nhấn nút mua thêm, quay về trang chủ và chọn thêm sản phẩm



## Bổ sung thêm chức năng

### Index.cshtml

<tr>

<td colspan="3">

<a href="@Url.Action("Index","Site")" class="btn-sm btn-success"> <i class="fa fa-plus"></i>Mua thêm sản phẩm

</a>

<button class="btn-sm btn-info" type="submit">Cập nhật đơn hàng</button>

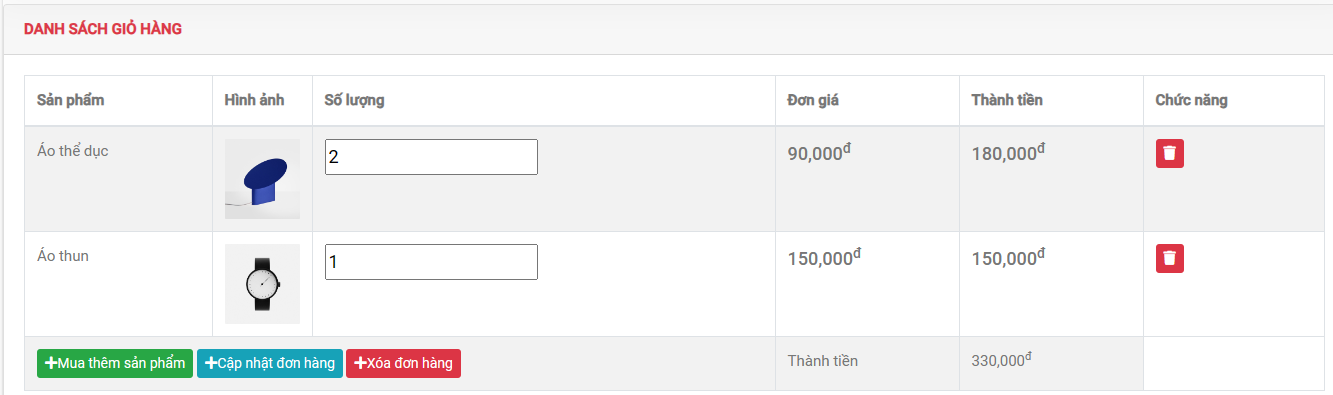
<a href="@Url.Action("CartDelAll","Giohang")" class="btn-sm btn-danger"> <i class="fa fa-plus"></i>Xóa đơn hàng</a>

</td>

<td>Thành tiền</td>

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

</tr>



Để có thể thực hiện các thao tác thêm, xóa, sửa dữ liệu trước đây trong các trang Admin chúng ta thực hiện thao tác trong form với cú pháp

@using (Html.BeginForm())

{…}

Do vậy, để cập nhật cho Giỏ hàng chúng ta xây dựng using (Html.BeginForm("CartUpdate", "Giohang", FormMethod.Post)). Nội dung này sẽ thay thế cho các nội dung trước đây. Cụ thể:

### 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">

@if (Model != null)

{

using (Html.BeginForm("CartUpdate", "Giohang", FormMethod.Post))[[2]](#footnote-2)

{

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

<thead>

<tr>

<th>Sản phẩm</th>

<th style="width:100px;">Hình ảnh</th>

<th>Số lượng</th>

<th>Đơn giá</th>

<th>Thành tiền</th>

<th>Chức năng</th>

</tr>

</thead>

<tbody>

@{ decimal tongtien = 0;}

@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"><input name="amount" value="@item.Ammount" type="number" min="1" /></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>

<td><a href="@Url.Action("DelCart", "Giohang", new { productid = @item.ProductId })" class="btn-sm btn-danger"> <i class="fa fa-trash"></i></a></td>

</tr>

tongtien += item.Total;

}

<tr>

<td colspan="3">

<a href="@Url.Action("Index", "Site")" class="btn-sm btn-success"> <i class="fa fa-plus"></i>Mua thêm sản phẩm</a>

<input class="btn-sm btn-info" name="capnhat" value="Cập nhật đơn hàng" type="submit"/> </button>

<a href="@Url.Action("DelCartAll", "Giohang")" class="btn-sm btn-danger"> <i class="fa fa-plus"></i>Xóa đơn hàng</a>

</td>

<td>Thành tiền</td>

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

</tr>

</tbody>

</table>

}

}

else

{

< h1 > Chưa có giỏ hàng </ h1 >

}

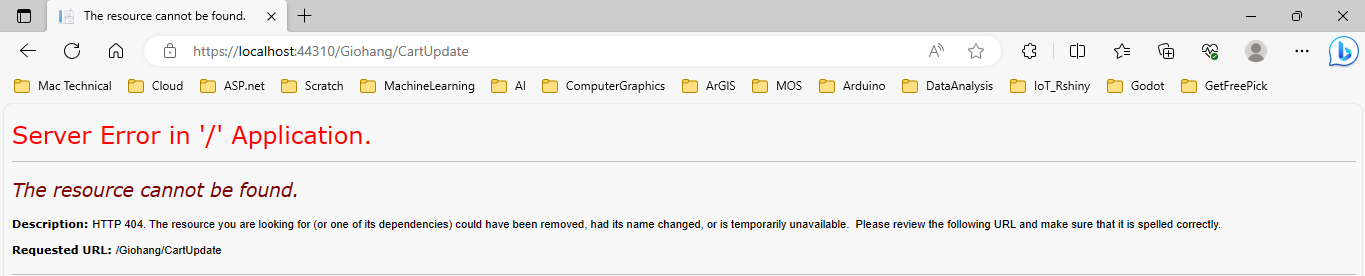
</div><!-- /.card-body -->

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

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

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

# Kết quả



## gio hangcontroller

### Xây dựng hàm CartUpdate

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

///CartUpdate

public ActionResult CartUpdate(FormCollection form)

{

if (!string.IsNullOrEmpty(form["capnhat"]))//nut ThemCategory duoc nhan

{

var listamount = form["amount"];

//chuyen danh sach thanh dang mang: vi du 1,2,3,...

var listarr = listamount.Split(',');//cat theo dau ,

xcart.UpdateCart(listarr);

}

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

}

Trong hàm này chúng ta có thực hiện lời gọi đến hàm UpdateCart trong Xcart. Do vậy, chúng ta tiếp tục xây dựng hàm UpdateCart cho Xcart

### Xcart.cs

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

///UpdateCart

public void UpdateCart(string[] arramout)

{

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

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

int vt = 0;

foreach (CartItem cartitem in list)

{

list[vt].Ammount = int.Parse(arramout[vt]);

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

vt++;

}

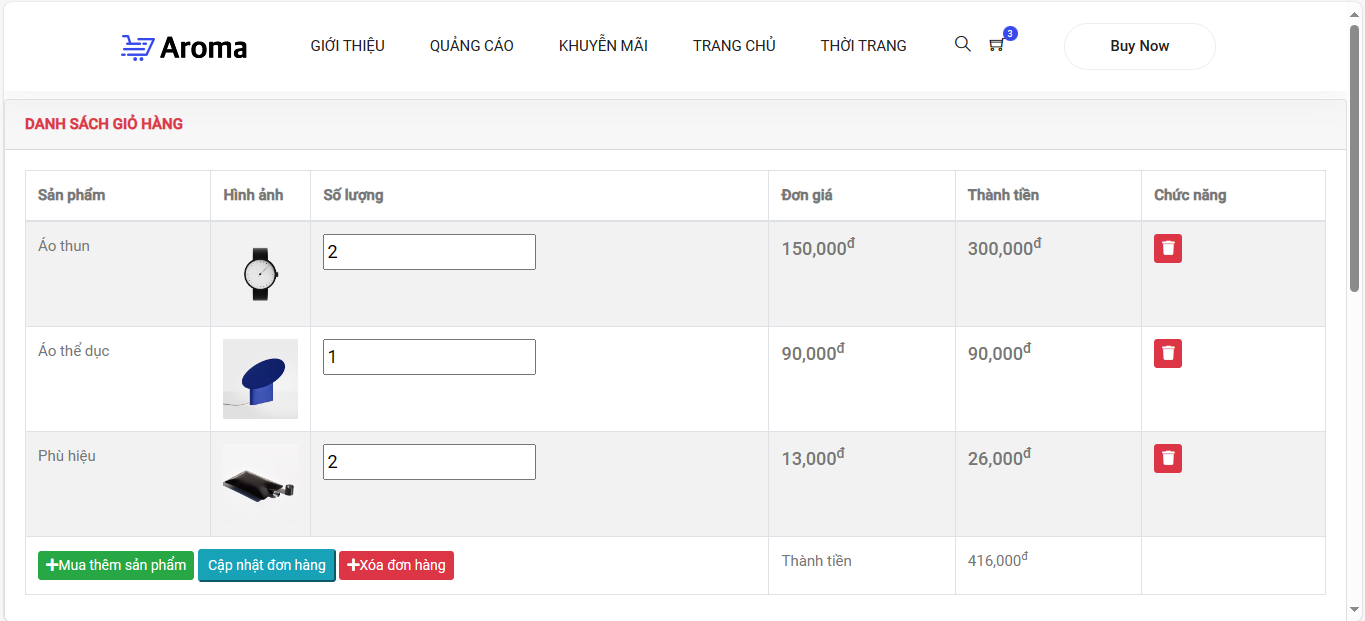
//cap nhat lai gio hang

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

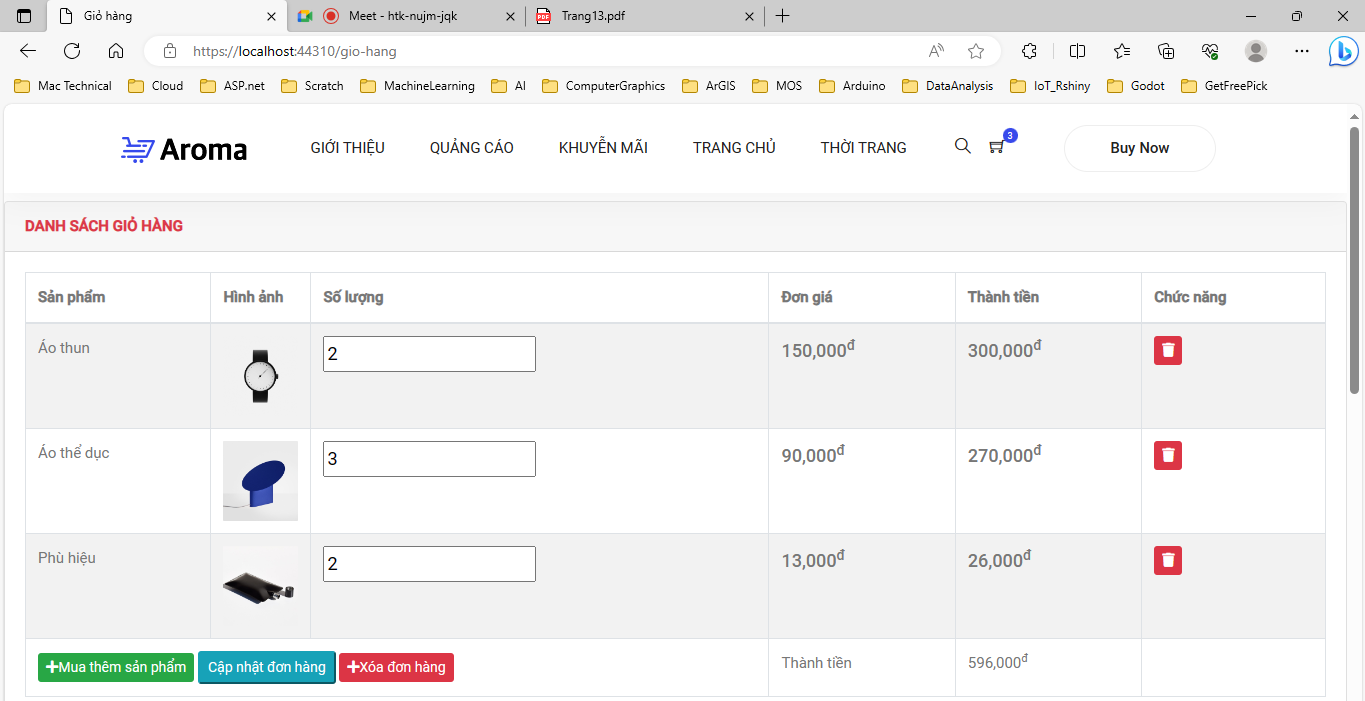
}

# Thử nghiệm

#### Trước khi nhấn nút tăng số lượng



#### Sau khi nhấn thay đổi số lượng và cập nhật



1. Chức năng xóa giỏ hàng [↑](#footnote-ref-1)
2. Bổ sung phương thức cập nhật đơn hàng [↑](#footnote-ref-2)