

BÀI TẬP THỰC HÀNH 08

(SV sử dụng kết quả bài 7 để thực hiện tiếp các yêu cầu trong bài tập này.)

Chuẩn bị:

- Thêm các gói sau vào project:

Newtonsoft.Json (hoặc System.Text.Json.JsonSerializer)

Microsoft.AspNetCore.Session

Microsoft.Extensions.Caching.Memory

- Mở **Program.cs**, bổ sung các khai báo sau (dòng 8-13, dòng 25)

```
5 // Add services to the container.
6 builder.Services.AddControllersWithViews();
7 builder.Services.AddDbContext<ApplicationDbContext>();
8 builder.Services.AddDistributedMemoryCache();
9 builder.Services.AddSession(options =>
10 {
11     options.Cookie.Name = "ABC";
12     options.IdleTimeout = TimeSpan.FromMinutes(30);
13 });
25 app.UseHttpsRedirection();
26 app.UseStaticFiles();
27 app.UseSession();
28 app.UseRouting();
```

1. Quản lý giỏ hàng sử dụng Session

- Trong thư mục **Models**, tạo lớp **CartItem.cs** (đại diện cho mặt hàng trong giỏ):

```
public class CartItem
{
    10 references
    public Mathang Mathang { get; set; }
    5 references
    public int SoLuong { get; set; }
}
```

- Trong **Controllers/CustomerController.cs**, viết các phương thức phục vụ xử lý giỏ hàng (tên session lưu thông tin giỏ hàng là “shopcart”)

Đọc danh sách CartItem trong giỏ hàng từ session:

```
// Đọc danh sách CartItem từ session
List<CartItem> GetCartItems()
{
    var session = HttpContext.Session;
    string? jsoncart = session.GetString("shopcart");
    if (jsoncart != null)
    {
        var cartItems = JsonConvert.DeserializeObject<List<CartItem>>(jsoncart);
        return cartItems ?? new List<CartItem>();
    }
    return new List<CartItem>();
}
```

```

    }

    // Lưu danh sách CartItem trong giỏ hàng vào session
    void SaveCartSession(List<CartItem> list)
    {
        var session = HttpContext.Session;
        string jsoncart = JsonConvert.SerializeObject(list);
        session.SetString("shopcart", jsoncart);
    }

    // Xóa session giỏ hàng
    void ClearCart()
    {
        var session = HttpContext.Session;
        session.Remove("shopcart");
    }
}

```

➤ Chức năng cho một mặt hàng vào giỏ

- Trang **View/Customer/Index.cshtml** (trang danh sách sản phẩm), bổ sung nút cho khách đưa hàng vào giỏ

```

<a class="btn btn-outline-info mt-auto"
    asp-controller="Home" asp-action="AddToCart" asp-route-id="@item.MaMh">
    Chọn mua
</a>

```

- Bổ sung vào **CustomerController.cs** phương thức thêm hàng vào giỏ

```

// Cho hàng vào giỏ
public async Task<IActionResult> AddToCart(int id)
{
    var mathang = await _context.Mathangs
        .FirstOrDefaultAsync(m => m.MaMh == id);
    if (mathang == null)
    {
        return NotFound("Sản phẩm không tồn tại");
    }
    var cart = GetCartItems();
    var item = cart.Find(p => p.Mathang.MaMh == id);
    if (item != null)
    {
        item.SoLuong++;
    }
    else
    {
        cart.Add(new CartItem() { Mathang = mathang, SoLuong = 1
    });
    }
    SaveCartSession(cart);
    return RedirectToAction(nameof(ViewCart));
}



// Chuyển đến view xem giỏ hàng
public IActionResult ViewCart()
{
    return View(GetCartItems());
}

```

- Trong **View/Customer**, tạo trang **ViewCart.cshtml** liệt kê các sản phẩm trong giỏ hàng. Đồng thời đặt liên kết mở giỏ hàng lên thanh navigation.

ABC Shop
Trang chính
Giới thiệu
Danh mục sản phẩm
Đăng nhập
Giỏ hàng 2

Giỏ hàng của bạn

STT	Sản phẩm	Giá	Số lượng	Thành tiền
1	 Máy Tính Casio FX580VN X-PK (Màu Hồng)	681,000 đ	<input type="text" value="1"/> <input type="button" value="Cập nhật"/>	681,000 đ <input type="button" value="Xóa"/>
2	 Máy Tính CASIO FX-880BTG (Màu Xanh Biển)	738,000 đ	<input type="text" value="1"/> <input type="button" value="Cập nhật"/>	738,000 đ <input type="button" value="Xóa"/>
Thành tiền				1,419,000 đ
<input type="button" value="Thanh toán"/>				

```

@model List<shop.Models.CartItem>
<br />
<h3>Giỏ hàng của bạn</h3>

@if (Model.Count > 0)
{
    int stt = 1;          // số thứ tự
    int thanhtien = 0;     // thành tiền
    int tongtien = 0;     // tổng tiền

    <table class="table">
        <tr>
            <th>STT</th>
            <th>Sản phẩm</th>
            <th>Giá</th>
            <th>Số lượng</th>
            <th>Thành tiền</th>
            <th></th>
        </tr>
        @foreach (var i in Model)
        {
            thanhtien = i.MatHang.GiaBan * i.SoLuong ?? 1;
            tongtien += thanhtien;
            <tr class="align-middle">
                <td>@(stt++)</td>
                <td>
                    
                    @(i.MatHang.Ten)
                </td>
                <td>@(i.MatHang.GiaBan?.ToString("N0")) đ</td>
                <td>
                    <form asp-action="UpdateItem" method="post">
                        <div class="row">
                            <div class="col">
                                <input name="id" type="hidden" asp-
for="@(i.MatHang.MaMh)" />
                                <input name="quantity" type="number" asp-
for="@(i.SoLuong)" class="form-control" />
                            </div>
                            <div class="col">
                                <input class="btn btn-warning" type="submit"
value="Cập nhật" />
                            </div>
                        </div>
                    </form>
                </td>
                <td>@(thanhtien.ToString("N0")) đ</td>
            </tr>
        }
    </table>
}

```

```

        <td><a asp-action="RemoveItem" asp-route-id="@i.MatHang.MaMh"
class="text-danger">Xóa</a></td>
        </tr>
    }
    <tr>
        <td colspan="4" class="text-end">Thành tiền</td>
        <td colspan="2"><strong class="text-
danger">@(tongtien.ToString("N0")) đ</strong></td>
    </tr>
</table>
<div class="text-end">
    <a class="btn btn-info" asp-action="CheckOut">Thanh toán</a>
</div>
}
else
{
    <p class="alert alert-danger">Giỏ hàng rỗng!</p>
}

```

➤ Chức năng cập nhật giỏ hàng, gồm:

- Xóa một mặt hàng khỏi giỏ
- Thay đổi số lượng một mặt hàng trong giỏ

```

// Xóa một mặt hàng khỏi giỏ
public IActionResult RemoveItem(int id)
{
    var cart = GetCartItems();
    var item = cart.Find(p => p.MatHang.MaMh == id);
    if (item != null)
    {
        cart.Remove(item);
    }
    SaveCartSession(cart);
    return RedirectToAction(nameof(ViewCart));
}

// Cập nhật số lượng một mặt hàng trong giỏ
public IActionResult UpdateItem(int id, int quantity)
{
    var cart = GetCartItems();
    var item = cart.Find(p => p.MatHang.MaMh == id);
    if (item != null)
    {
        item.SoLuong = quantity;
    }
    SaveCartSession(cart);
    return RedirectToAction(nameof(ViewCart));
}

```

2. Xử lý đơn hàng

(Xây dựng giao diện lập đơn hàng, *chưa yêu cầu khách phải có tài khoản khi đặt hàng*. Lưu thông tin các đối tượng sau vào database: khách hàng, đơn hàng, chi tiết đơn hàng.)

- Từ nút Thanh toán trong trang xem giỏ hàng ViewCart, dẫn tới trang thanh toán **CheckOut** (mẫu gợi ý cho View, SV có thể thiết kế khác):

Vui lòng kiểm tra đầy đủ thông tin trước khi hoàn tất đơn hàng

Thông tin khách hàng

Email
kh01@gmail.com

Họ tên
Nguyễn Thị Thu An

Điện thoại
0123456789

Địa chỉ
25 Võ Thị Sáu, Đồng Xuyên, An Giang

Xác nhận đơn hàng

Thông tin đơn hàng

STT	Sản phẩm	Giá	Số lượng	Thành tiền
1	 Máy Tính Casio FX580VN X-PK (Màu Hồng)	681,000 đ	1	681,000 đ
2	 Khay Cầm Bút Flexoffice FO-PS01	54,000 đ	5	270,000 đ
Thành tiền				951,000 đ

```

@model List<shop.Models.CartItem>
@{
    ViewData["Title"] = "Thanh toán";
}
<p>Vui lòng kiểm tra đầy đủ thông tin trước khi hoàn tất đơn hàng</p>
<div class="row">
    <div class="col">
        <h4 class="text-info">Thông tin khách hàng</h4>
        <form asp-action="CreateBill">
            <div class="mb-3">
                <label>Email</label>
                <input name="email" class="form-control" type="text" />
            </div>
            <div class="mb-3">
                <label>Họ tên</label>
                <input name="hoten" class="form-control" type="text" />
            </div>
            <div class="mb-3">
                <label>Điện thoại</label>
                <input name="dienthoai" class="form-control" type="text" />
            </div>
            <div class="mb-3">
                <label>Địa chỉ</label>
                <textarea name="diachi" class="form-control"></textarea>
            </div>
            <div class="mb-3">
                <input type="submit" class="btn btn-primary" value="Xác nhận đơn
hàng" />
            </div>
        </form>
    </div>
    <div class="col">
        <h4 class="text-info">Thông tin đơn hàng</h4>
        @if (Model.Count > 0)
        {
            int stt = 1; // số thứ tự
            int thanhtien = 0; // thành tiền
            int tongtien = 0; // tổng tiền

            <table class="table">
                <tr class="table-info">
                    <th>STT</th>
                    <th>Sản phẩm</th>
                    <th>Giá</th>
                    <th>Số lượng</th>
                </tr>
            </table>
        }
    </div>
</div>

```

```

        <th>Thành tiền</th>

    </tr>
    @foreach (var i in Model)
    {
        thanh tien = i.MatHang.GiaBan * i.SoLuong??1;
        tongtien += thanh tien;
        <tr class="align-middle">
            <td>@(stt++)</td>
            <td>
                
                @(i.MatHang.Ten)
            </td>
            <td>@(i.MatHang.GiaBan?.ToString("n0")) đ</td>
            <td>@(i.SoLuong)</td>
            <td>@(thanh tien.ToString("n0")) đ</td>

        </tr>
    }
    <tr>
        <td colspan="4" class="text-end fw-bold">Thành tiền</td>
        <td><strong class="text-danger">@(tongtien.ToString("n0"))
đ</strong></td>
    </tr>
</table>

}
</div>
</div>

```

- Trong **Controllers/CustomerController.cs**, viết phương thức **CreateBill** lưu thông tin đơn hàng, chi tiết đơn hàng,...:

```

// Lập hóa đơn: lưu hóa đơn, lưu chi tiết hóa đơn
[HttpPost, ActionName("CreateBill")]
public async Task<IActionResult> CreateBill(string email, string hoten,
string dienthoai, string diachi)
{
    // Xử lý thông tin khách hàng (trường hợp khách mới)
    var kh = new Khachhang();
    kh.Email = email;
    kh.Ten = hoten;
    kh.DienThoai = dienthoai;
    _context.Add(kh);
    await _context.SaveChangesAsync();

    var hd = new Hoadon();
    hd.Ngay = DateTime.Now;
    hd.MaKh = kh.MaKh;

    _context.Add(hd);
    await _context.SaveChangesAsync();

    // thêm chi tiết hóa đơn
    var cart = GetCartItems();

    int thanh tien = 0;
    int tongtien = 0;
    foreach (var i in cart)
    {
        var ct = new Cthoadon();
        ct.MaHd = hd.MaHd;
        ct.MaMh = i.MatHang.MaMh;
    }
}

```

```

        thanhTien = i.MatHang.GiaBan * i.SoLuong??1;
        tongtien += thanhTien;
        ct.DonGia = i.MatHang.GiaBan;
        ct.SoLuong = (short)i.SoLuong;
        ct.ThanhTien = thanhTien;
        _context.Add(ct);
    }
    await _context.SaveChangesAsync();

    // cập nhật tổng tiền hóa đơn
    hd.TongTien = tongtien;
    _context.Update(hd);
    await _context.SaveChangesAsync();

    // xóa giỏ hàng
    ClearCart();

    return View(hd);
}

```

- Tạo view thông báo đơn hàng đã lưu vào hệ thống:

```

@model shop.Models.Hoadon
@{
    ViewData["Title"] = "Đặt hàng thành công";
}
<h3>Đặt hàng thành công</h3>
<p>Đơn hàng mã số @Model.MaHd trị giá @(Model.TongTien)đ đã được hệ thống ghi nhận. Chúng tôi sẽ sớm liên hệ bạn để xác nhận...</p>

```

- Xử lý thêm trường hợp khách hàng đã có tài khoản, yêu cầu đăng nhập để mua hàng:
SV tự viết.