Nộp bài kiểm tra

Công nghệ Web và ứng dụng ASP.Net

Trương Khánh Duy DPM205420

Email: tkduy\_21pm@student.agu.edu.vn

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Phần mềm đa phương tiện

Mô tả được tạo tự độngẢnh có chứa văn bản, ảnh chụp màn hình, phần mềm

Mô tả được tạo tự động

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, màn hình

Mô tả được tạo tự động

KhachHang.cs

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Phần mềm đa phương tiện

Mô tả được tạo tự động

**KhachhangRepository.cs**

Ảnh có chứa văn bản, ảnh chụp màn hình, màn hình, phần mềm

Mô tả được tạo tự động

KhachHangService.cs

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, màn hình

Mô tả được tạo tự động

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm

Mô tả được tạo tự động

KhachHangController.cs

Using AutoMapper;

using Infrastructure.Entities;

using Infrastructure.Service;

using learnX.Models;

using Microsoft.AspNetCore.Mvc;

using Microsoft.EntityFrameworkCore;

using System;

namespace learnX.Controllers

{

public class KhachangController : Controller

{

private readonly IKhachHangService KhachhangService;

private readonly IMapper mapper;

public KhachangController(IKhachHangService KhachhangService, IMapper mapper)

{

this.KhachhangService = KhachhangService;

this.mapper = mapper;

}

public IActionResult Index()

{

return View(KhachhangService.GetKhachHang());

}

public IActionResult AddOrEdit(int id = 0)

{

KhachHangViewModel data = new KhachHangViewModel();

ViewBag.RenderedHtmlTitle = id == 0 ? "THÊM MỚI TÀI KHOẢN" : "CẬP NHẬT TÀI KHOẢN";

if (id != 0)

{

KhachHang res = KhachhangService.GetKhachHang(id);

data = mapper.Map<KhachHangViewModel>(res);

if (data == null)

{

return NotFound();

}

}

return View(data);

}

[HttpPost]

[ValidateAntiForgeryToken]

public IActionResult AddOrEdit(int id, KhachHangViewModel data)

{

ViewBag.RenderedHtmlTitle = id == 0 ? "THÊM MỚI TÀI KHOẢN" : "CẬP NHẬT TÀI KHOẢN";

if (ModelState.IsValid)

{

try

{

KhachHang res = mapper.Map<KhachHang>(data);

if (id != 0)

{

KhachhangService.UpdateKhachHang(res);

}

else

{

KhachhangService.InsertKhachHang(res);

}

}

catch (DbUpdateConcurrencyException)

{

return NotFound();

}

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

}

return View(data);

}

[HttpPost]

[ValidateAntiForgeryToken]

public IActionResult Delete(int id)

{

KhachHang res = KhachhangService.GetKhachHang(id);

KhachhangService.DeleteKhachHang(res);

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

}

}

}

KhachHangViewModel.cs

using System.ComponentModel.DataAnnotations;

using System.ComponentModel;

using System;

namespace learnX.Models

{

public class KhachHangViewModel

{

public KhachHangViewModel()

{

}

[DisplayName("Ma khach hang")]

public int Makh { get; set; }

[DisplayName("Họ và tên")]

[MaxLength(125, ErrorMessage = "Họ và tên không quá 125 ký tự.")]

public string ten { get; set; }

[DisplayName("So dien thoai")]

public bool SDT { get; set; }

}

AddOrEdit.cshtml

@model KhachHangViewModel

@{

}

<div class="container">

<div class="card">

<div class="card-body">

<p class="mt-0 mb-2 fw-semibold text-dark font-18 text-center">@ViewBag.RenderedHtmlTitle</p>

<**form** **asp-action**="AddOrEdit" **asp-controller**="KhachHang" **asp-route-id**="@Model.Makh">

<**div** **asp-validation-summary**="ModelOnly" class="text-danger"></**div**>

<**input** **type**="hidden" **asp-for**="Makh" **value**="@Model.Makh" />

<div>

<div class="form-group mb-2">

<**label** **asp-for**="ten" class="control-label"></**label**>

<div class="input-group mb-2">

<**input** **asp-for**="ten" class="form-control" placeholder="Nhập họ và tên" **value**="@Model.ten" />

</div>

<**span** **asp-validation-for**="ten" class="text-danger"></**span**>

</div><!--end form-group-->

<div class="form-group mb-2">

<**label** **asp-for**="SDT" class="control-label"></**label**>

<div class="input-group mb-2">

<**input** **asp-for**="SDT" **type**="number" class="form-control" placeholder="Nhập số điện thoại" **value**="@Model.SDT" />

</div>

<**span** **asp-validation-for**="SDT" class="text-danger"></**span**>

</div><!--end form-group-->

<div class="form-group">

<br />

<div class="col-md-12 offset-md-5">

<div class="d-flex">

<div style="margin-right:10px;">

<input type="submit" value="Submit" class="btn btn-primary btn-block" style="width:80px" />

</div>

<div>

<**a** **asp-controller**="KhachHang" **asp-action**="Index" class="btn btn-success text-white"><i class="fe-log-out"></i> Close</**a**>

</div>

</div>

</div>

</div>

</div>

</**form**>

</div>

</div> <!-- end card-->

</div>

@section scripts{

}

Index

<div class="row d-flex justify-content-center">

<div class="col-12 align-self-center">

<div class="card">

<div class="card-body">

<p class="mt-0 mb-2 fw-semibold text-dark font-18 text-center">DANH SÁCH KHÁCH HÀNG</p>

<div class="table-responsive">

<div class="mb-2">

<**a** **asp-controller**="KhachHang" **asp-action**="AddOrEdit" class="btn btn-success text-white"><i class="fas fa-random"></i> Thêm mới</**a**>

</div>

<table id="basic-datatable" class="table table-hover table-centered dt-responsive nowrap w-100 mb-0">

<thead>

<tr>

<th>Họ và tên</th>

<th>Số Điện Thọai</th>

<th></th>

</tr>

</thead>

<tbody>

@foreach (var item in Model)

{

<tr>

<td>@item.ten</td>

<td>@item.SDT</td>

<td>

<**a** **asp-controller**="KhachHang" **asp-action**="AddOrEdit" **asp-route-id**="@item.Makh" class="btn btn-info text-white">

<i class="fas fa-edit text-white"></i> Edit

</**a**>

<**form** **asp-action**="Delete" **asp-route-id**="@item.Makh" onsubmit="return jQueryAjaxDelete(this)" class="d-inline">

<button class="btn btn-danger" type="submit">

<i class="fas fa-trash-alt text-white"></i> Delete

</button>

</**form**>

</td>

</tr>

}

</tbody>

</table><!--end /table-->

</div>

</div>

</div>

</div>

</div>

@section scripts{

<script>

jQueryAjaxDelete = form => {

try {

if (confirm('Are you sure to delete this record ?')) {

$.ajax({

type: 'POST',

url: form.action,

data: new FormData(form),

contentType: false,

processData: false,

success: function (res) {

location.reload(false);

},

error: function (err) {

console.log(err)

}

})

}

} catch (ex) {

console.log(ex)

}

//prevent default form submit event

return false;

}

</script>

}

Mapping

using AutoMapper;

using Infrastructure.Entities;

using learnX.Models;

namespace learnX

{

public class Mapping : Profile

{

public Mapping()

{

// Add as many of these lines as you need to map your objects

CreateMap<User, UserViewModel>();

CreateMap<UserViewModel, User>();

CreateMap<KhachHang, KhachHangViewModel>();

CreateMap<KhachHangViewModel, KhachHang>();

}

}

}

Appsetting.json

{

"Logging": {

"LogLevel": {

"Default": "Information",

"Microsoft": "Warning",

"Microsoft.Hosting.Lifetime": "Information"

}

},

"ConnectionStrings": {

"EXDbContextConnection":"Data Source=.\\SQLEXPRESS;Initial Catalog=Example;MultipleActiveResultSets=true;Integrated Security=True;",

},

"AllowedHosts": "\*"

}