Министерство науки и высшего образования Российской Федерации

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

**(ВлГУ)**

**Кафедра информационных систем и программной инженерии**

Лабораторная работа №5

по дисциплине «Технологии программирования»

Тема: «Шаблон проектирования MVC в ASP.NET»

Выполнил:

студент гр. ПРИ-123

А.Ц. Нямаа

Принял:

Ст. пр. кафедры ИСПИ

В.В. Данилов

Владимир, 2025 г.

ЦЕЛЬ РАБОТЫ:

Познакомиться с технологией создания и использования шаблона проектирования MVC на платформе ASP.NET.

ЗАДАНИЕ (вариант 12).

Использовать шаблон MVC в приложении, созданном в предыдущих лабораторных работах. Также добавить следующие страницы: страница детального просмотра элемента (Details), страница редактирования элемент (Edit), страница удаления элемента (Delete).

ВЫПОЛНЕНИЕ РАБОТЫ:

В начале выполнения данной работы я определил физическую структуру веб-приложения, которая соответствует шаблону MVC.

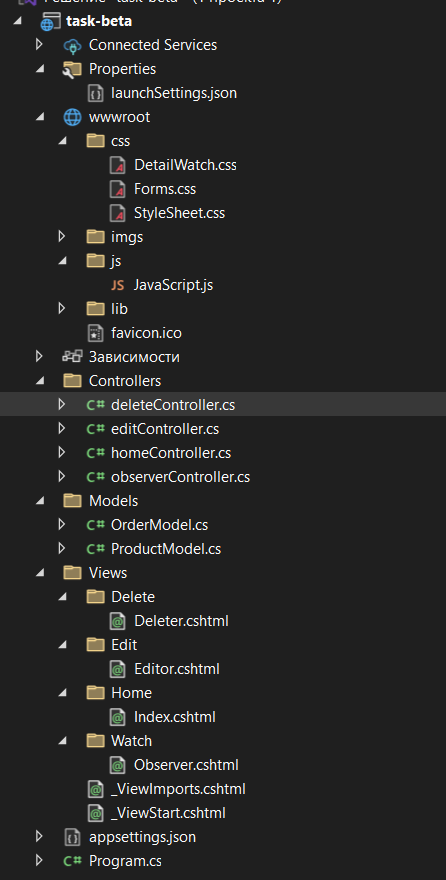


Рисунок 1. Физическая структура веб-приложения.

Следующим шагом написал необходимый код, соответствующий всем заданным в методическом указании требованиям.

Листинг Program.cs:

var builder = WebApplication.CreateBuilder(args);

builder.Services.AddRazorPages();

builder.Services.AddControllers();

builder.Services.AddEndpointsApiExplorer();

var app = builder.Build();

app.UseRouting();

app.UseStaticFiles();

app.MapRazorPages();

app.MapControllerRoute(

name: "default",

pattern: "{controller=Home}/{action=Index}/{id?}"

);

app.Run();

Листинг Home/Index.cshtml:

@using task\_beta.Controllers

@model HomeController

@{

}

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<link rel="preconnect" href="https://fonts.googleapis.com">

<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

<link href="https://fonts.googleapis.com/css2?family=Overpass:ital,wght@0,100..900;1,100..900&display=swap" rel="stylesheet">

<title>Clock Shop</title>

<link rel="stylesheet" href="~/css/StyleSheet.css">

</head>

<body>

<header>

<ul class="list-style">

<li><a href="#">Home</a></li>

<li><a href="~/Watch/Observer">Observer</a></li>

<li><a href="~/Edit/Editor">Editor</a></li>

<li><a href="~/Delete/Deleter">Deleter</a></li>

</ul>

<div class="leftText">

<h1> ДИЗАЙНЕРСКИЙ ЧАСЫ </h1>

<p> ПОЛНОСТЬЮ РУЧНОЕ ИЗГОТОВЛЕНИЕ </p>

<button class="buyHeader buttonToChange openModel" data-price="900000"> КУПИТЬ </button>

<dialog class="dialogMain">

<form asp-action="Order" class="orderForm" method="post">

<h2>Оформление заказа</h2>

@if (ViewBag.successValid != null && ViewBag.successValid != "")

{

<p class="validSuccess">@ViewBag.successValid</p>

}

else if (ViewBag.errorValid != null && ViewBag.errorValid != "")

{

<p class="validError">@ViewBag.errorValid</p>

}

<p>

<label asp-for="order.NameOrder">Имя:</label>

<input type="text" id="name" asp-for="order.NameOrder" placeholder="Имя">

<span asp-validation-for="order.NameOrder" class="text-danger"></span>

</p>

<p>

<label asp-for="order.SecondNameOrder">Фамилия:</label>

<input type="text" id="SecondNameOrder" asp-for="order.SecondNameOrder" placeholder="Фамилия">

<span asp-validation-for="order.SecondNameOrder" class="text-danger"></span>

</p>

<p>

<label asp-for="order.EmailOrder">Почтовый ящик:</label>

<input type="email" id="email" asp-for="order.EmailOrder" placeholder="Почтовый ящик">

<span asp-validation-for="order.EmailOrder" class="text-danger"></span>

</p>

<p>

<label asp-for="order.AddressOrder">Адрес доставки:</label>

<input type="text" id="address" asp-for="order.AddressOrder" placeholder="Ул. Пушкино, д. 2, кв. 1">

<span asp-validation-for="order.AddressOrder" class="text-danger"></span>

</p>

<p>

<label asp-for="order.QuantityOrder">Количество:</label>

<input type="number" id="quantity" asp-for="order.QuantityOrder" placeholder="1">

<span asp-validation-for="order.QuantityOrder" class="text-danger"></span>

</p>

<p>

<label>Промокод:</label>

<input type="text" id="promo" placeholder="\*Промокод\*">

</p>

<p class="priceHeader"></p>

<button> Заказать! </button>

</form>

<button id="closeModel" class="closeModel">&#x2715 </button>

</dialog>

</div>

<div class="rightText">

<p> JACOB & CO <br>EPIC X TOURBILON BAGUETTE <br>900.000$</p>

<img src="imgs/mainPhotoWatches.png">

</div>

</header>

<main>

<div class="textInMain">

<p>ХИТЫ</p>

<p>ПОСЛЕДНИЕ ПОСТУПЛЕНИЯ</p>

</div>

<div class="productsList">

@foreach(var product in ViewBag.products)

{

<div class="productUnit">

<img src="@product.imagePathProduct" />

<p>@product.NameProduct</p>

<p>@($"{product.CostProduct}$")</p>

<button class="buttonUnit buttonToChange openModel" data-price="@product.CostProduct"> Купить </button>

</div>

}

</div>

</main>

<script src="~/js/JavaScript.js"></script>

</body>

</html>

Листинг HomeController.cs

using Microsoft.AspNetCore.Mvc;

using MySql.Data.MySqlClient;

using OrderModel.Models;

using ProductModel.Models;

namespace task\_beta.Controllers

{

public class HomeController : Controller

{

string connection = "Server=localhost;database=task;user=root;password=1arthur\_nyamaa1";

private List<Product> products = new List<Product>();

[BindProperty]

public Order order { get; set; } = new Order();

[HttpGet]

public async Task<IActionResult> Index()

{

products.Clear();

string sqlCommandProduct = "select \* from product";

using (MySqlConnection connectionCurrent = new MySqlConnection(connection))

{

connectionCurrent.Open();

MySqlCommand commandCurrent = new MySqlCommand(sqlCommandProduct, connectionCurrent);

using (var sqlSelectProduct = commandCurrent.ExecuteReader())

{

while (sqlSelectProduct.Read())

{

products.Add(new Product

{

IdProduct = sqlSelectProduct.GetInt32("ProductId"),

NameProduct = sqlSelectProduct.GetString("ProductName"),

CostProduct = sqlSelectProduct.GetInt32("ProductCost"),

imagePathProduct = sqlSelectProduct.GetString("ProductIMG")

});

}

}

}

ViewBag.Products = products;

ViewBag.succesValid = successValid;

ViewBag.errorValid = errorValid;

return View();

}

private string successValid = "";

private string errorValid = "";

[HttpPost]

public IActionResult Order(Order order)

{

if (!ModelState.IsValid)

{

errorValid = "Некоторые данные были введены неправильно!";

return RedirectToAction("Index");

}

string sqlCommand = "insert into orderinfo (OrderName, OrderSurname, OrderEmail, OrderAddress, OrderQuantity)" +

"values (@NameOrder, @SecondNameOrder, @EmailOrder, @AddressOrder, @QuantityOrder); ";

using (MySqlConnection connectionCurrent = new MySqlConnection(connection))

{

connectionCurrent.Open();

MySqlCommand commandCurrent = new MySqlCommand(sqlCommand, connectionCurrent);

commandCurrent.Parameters.AddWithValue("@NameOrder", order.NameOrder);

commandCurrent.Parameters.AddWithValue("@SecondNameOrder", order.SecondNameOrder);

commandCurrent.Parameters.AddWithValue("@EmailOrder", order.EmailOrder);

commandCurrent.Parameters.AddWithValue("@AddressOrder", order.AddressOrder);

commandCurrent.Parameters.AddWithValue("@QuantityOrder", order.QuantityOrder);

commandCurrent.ExecuteNonQuery();

connectionCurrent.Close();

}

successValid = $"Спасибо {order.NameOrder} {order.SecondNameOrder}, заказ выполнен успешно!{Environment.NewLine}" +

$"Доставим товар по адресу {order.AddressOrder} в количестве {order.QuantityOrder} шт.{Environment.NewLine}" +

$"Электронный чек придет в ближайшее время на почту {order.EmailOrder}";

return RedirectToAction("Index");

}

}

}

Листинг Watch/Observer.cshtml:

@using task\_beta.Controllers

@model watchController

@{

}

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<link rel="preconnect" href="https://fonts.googleapis.com">

<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

<link href="https://fonts.googleapis.com/css2?family=Overpass:ital,wght@0,100..900;1,100..900&display=swap" rel="stylesheet">

<title>Details</title>

<link rel="stylesheet" href="~/css/DetailWatch.css">

</head>

<body>

<header>

<ul class="list-style">

<li><a href="~/Home">Home</a></li>

<li><a href="#">Observer</a></li>

<li><a href="~/Edit/Editor">Editor</a></li>

<li><a href="~/Delete/Deleter">Deleter</a></li>

</ul>

</header>

<main>

<div class="addModule orderContainer ">

<div class="textInMain addText textOrder">

<p>СТРАНИЦА ДЕТАЛЬНОГО ПРОСМОТРА ЭЛЕМЕНТА (DETAILS)</p>

</div>

<div class="orderItemsGrid">

<div class="orderHeader">Id заказа</div>

<div class="orderHeader">Имя покупателя</div>

<div class="orderHeader">Фамилия покупателя</div>

<div class="orderHeader">Почта покупателя</div>

<div class="orderHeader">Адрес доставки</div>

<div class="orderHeader">Количество товара</div>

@foreach (var order in ViewBag.orders)

{

<div class="singleItem">@order.IdOrder</div>

<div class="singleItem">@order.NameOrder</div>

<div class="singleItem">@order.SecondNameOrder</div>

<div class="singleItem">@order.EmailOrder</div>

<div class="singleItem">@order.AddressOrder</div>

<div class="singleItem">@order.QuantityOrder</div>

}

</div>

</div>

</main>

</body>

</html>

Листинг watchController.cs:

using Microsoft.AspNetCore.Mvc;

using MySql.Data.MySqlClient;

using OrderModel.Models;

namespace task\_beta.Controllers

{

public class watchController : Controller

{

string connection = "Server=localhost;database=task;user=root;password=1arthur\_nyamaa1";

private static List<Order> orders = new List<Order>();

public IActionResult Observer()

{

orders.Clear();

string sqlCommandOrder = "select \* from orderinfo";

using (MySqlConnection connectionCurrent = new MySqlConnection(connection))

{

connectionCurrent.Open();

MySqlCommand commandCurrent = new MySqlCommand(sqlCommandOrder, connectionCurrent);

using (var sqlSelectProduct = commandCurrent.ExecuteReader())

{

while (sqlSelectProduct.Read())

{

orders.Add(new Order

{

IdOrder = sqlSelectProduct.GetInt32("OrderId"),

NameOrder = sqlSelectProduct.GetString("OrderName"),

SecondNameOrder = sqlSelectProduct.GetString("OrderSurname"),

EmailOrder = sqlSelectProduct.GetString("OrderEmail"),

AddressOrder = sqlSelectProduct.GetString("OrderAddress"),

QuantityOrder = sqlSelectProduct.GetInt32("OrderQuantity"),

});

}

}

}

ViewBag.Orders = orders;

return View();

}

}

}

Листинг Edit/Editor.cshtml:

@using task\_beta.Controllers

@model editController

@{

}

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<link rel="preconnect" href="https://fonts.googleapis.com">

<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

<link href="https://fonts.googleapis.com/css2?family=Overpass:ital,wght@0,100..900;1,100..900&display=swap" rel="stylesheet">

<title>Details</title>

<link rel="stylesheet" href="~/css/Forms.css">

</head>

<body>

<header>

<ul class="list-style">

<li><a href="~/Home">Home</a></li>

<li><a href="~/Watch/Observer">Observer</a></li>

<li><a href="#">Editor</a></li>

<li><a href="~/Delete/Deleter">Deleter</a></li>

</ul>

</header>

<main>

<div class="addModule orderContainer ">

<div class="textInMain addText textOrder">

<p>СТРАНИЦА РЕДАКТИРОВАНИЯ ЭЛЕМЕНТА (EDIT)</p>

</div>

<form asp-action="EditorPost" class="Form" method="post">

<p>

<label asp-for="order.IdOrder">Id заказа:</label>

<input required type="text" id="idOrder" asp-for="order.IdOrder" placeholder="Id...">

<span asp-validation-for="order.IdOrder" class="text-danger"></span>

</p>

<p>

<label asp-for="order.NameOrder">Имя:</label>

<input required type="text" id="name" asp-for="order.NameOrder" placeholder="Имя">

<span asp-validation-for="order.NameOrder" class="text-danger"></span>

</p>

<p>

<label asp-for="order.SecondNameOrder">Фамилия:</label>

<input required type="text" id="SecondNameOrder" asp-for="order.SecondNameOrder" placeholder="Фамилия">

<span asp-validation-for="order.SecondNameOrder" class="text-danger"></span>

</p>

<p>

<label asp-for="order.EmailOrder">Почтовый ящик:</label>

<input required type="email" id="email" asp-for="order.EmailOrder" placeholder="Почтовый ящик">

<span asp-validation-for="order.EmailOrder" class="text-danger"></span>

</p>

<p>

<label asp-for="order.AddressOrder">Адрес доставки:</label>

<input required type="text" id="address" asp-for="order.AddressOrder" placeholder="Ул. Пушкино, д. 2, кв. 1">

<span asp-validation-for="order.AddressOrder" class="text-danger"></span>

</p>

<p>

<label asp-for="order.QuantityOrder">Количество:</label>

<input required type="number" id="quantity" asp-for="order.QuantityOrder" placeholder="1">

<span asp-validation-for="order.QuantityOrder" class="text-danger"></span>

</p>

<button> Изменить </button>

</form>

</div>

</main>

</body>

</html>

Листинг editController.cs:

using Microsoft.AspNetCore.Mvc;

using MySql.Data.MySqlClient;

using OrderModel.Models;

namespace task\_beta.Controllers

{

public class editController : Controller

{

string connection = "Server=localhost;database=task;user=root;password=1arthur\_nyamaa1";

public Order order { get; set; } = new Order();

[HttpGet]

public IActionResult Editor()

{

return View();

}

[HttpPost]

public IActionResult EditorPost(Order order)

{

string sqlCommand = "UPDATE orderinfo " +

"SET OrderName = @NameOrder, " +

"OrderSurname = @SecondNameOrder," +

"OrderEmail = @EmailOrder, " +

"OrderAddress = @AddressOrder," +

"OrderQuantity = @QuantityOrder " +

"WHERE OrderId = @idOrder";

using (MySqlConnection connectionCurrent = new MySqlConnection(connection))

{

connectionCurrent.Open();

MySqlCommand commandCurrent = new MySqlCommand(sqlCommand, connectionCurrent);

commandCurrent.Parameters.AddWithValue("@idOrder", order.IdOrder);

commandCurrent.Parameters.AddWithValue("@NameOrder", order.NameOrder);

commandCurrent.Parameters.AddWithValue("@SecondNameOrder", order.SecondNameOrder);

commandCurrent.Parameters.AddWithValue("@EmailOrder", order.EmailOrder);

commandCurrent.Parameters.AddWithValue("@AddressOrder", order.AddressOrder);

commandCurrent.Parameters.AddWithValue("@QuantityOrder", order.QuantityOrder);

commandCurrent.ExecuteNonQuery();

connectionCurrent.Close();

}

return RedirectToAction("Editor", "Edit");

}

}

}

Листинг Delete/Deleter.cshtml:

@using task\_beta.Controllers

@model deleteController

@{

}

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<link rel="preconnect" href="https://fonts.googleapis.com">

<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

<link href="https://fonts.googleapis.com/css2?family=Overpass:ital,wght@0,100..900;1,100..900&display=swap" rel="stylesheet">

<title>Details</title>

<link rel="stylesheet" href="~/css/Forms.css">

</head>

<body>

<header>

<ul class="list-style">

<li><a href="~/Home">Home</a></li>

<li><a href="~/Watch/Observer">Observer</a></li>

<li><a href="~/Edit/Editor">Editor</a></li>

<li><a href="#">Deleter</a></li>

</ul>

</header>

<main>

<div class="addModule orderContainer ">

<div class="textInMain addText textOrder">

<p>СТРАНИЦА УДАЛЕНИЯ ЭЛЕМЕНТА (DELETE)</p>

</div>

<form asp-action="DeleterPost" class="Form" method="post">

<p>

<label asp-for="order.IdOrder">Id заказа:</label>

<input type="text" id="idOrder" asp-for="order.IdOrder" placeholder="Id...">

<span asp-validation-for="order.IdOrder" class="text-danger"></span>

</p>

<button> Удалить </button>

</form>

</div>

</main>

</body>

</html>

Листинг deleteController.cs:

using Microsoft.AspNetCore.Mvc;

using MySql.Data.MySqlClient;

using OrderModel.Models;

namespace task\_beta.Controllers

{

public class deleteController : Controller

{

string connection = "Server=localhost;database=task;user=root;password=1arthur\_nyamaa1";

public Order order = new Order();

[HttpGet]

public IActionResult Deleter()

{

return View();

}

[HttpPost]

public IActionResult DeleterPost(Order order)

{

string sqlCommand = "DELETE FROM orderinfo WHERE OrderId = @idOrder";

using (MySqlConnection connectionCurrent = new MySqlConnection(connection))

{

connectionCurrent.Open();

MySqlCommand commandCurrent = new MySqlCommand(sqlCommand, connectionCurrent);

commandCurrent.Parameters.AddWithValue("@idOrder", order.IdOrder);

commandCurrent.ExecuteNonQuery();

connectionCurrent.Close();

}

return RedirectToAction("Deleter", "Delete");

}

}

}

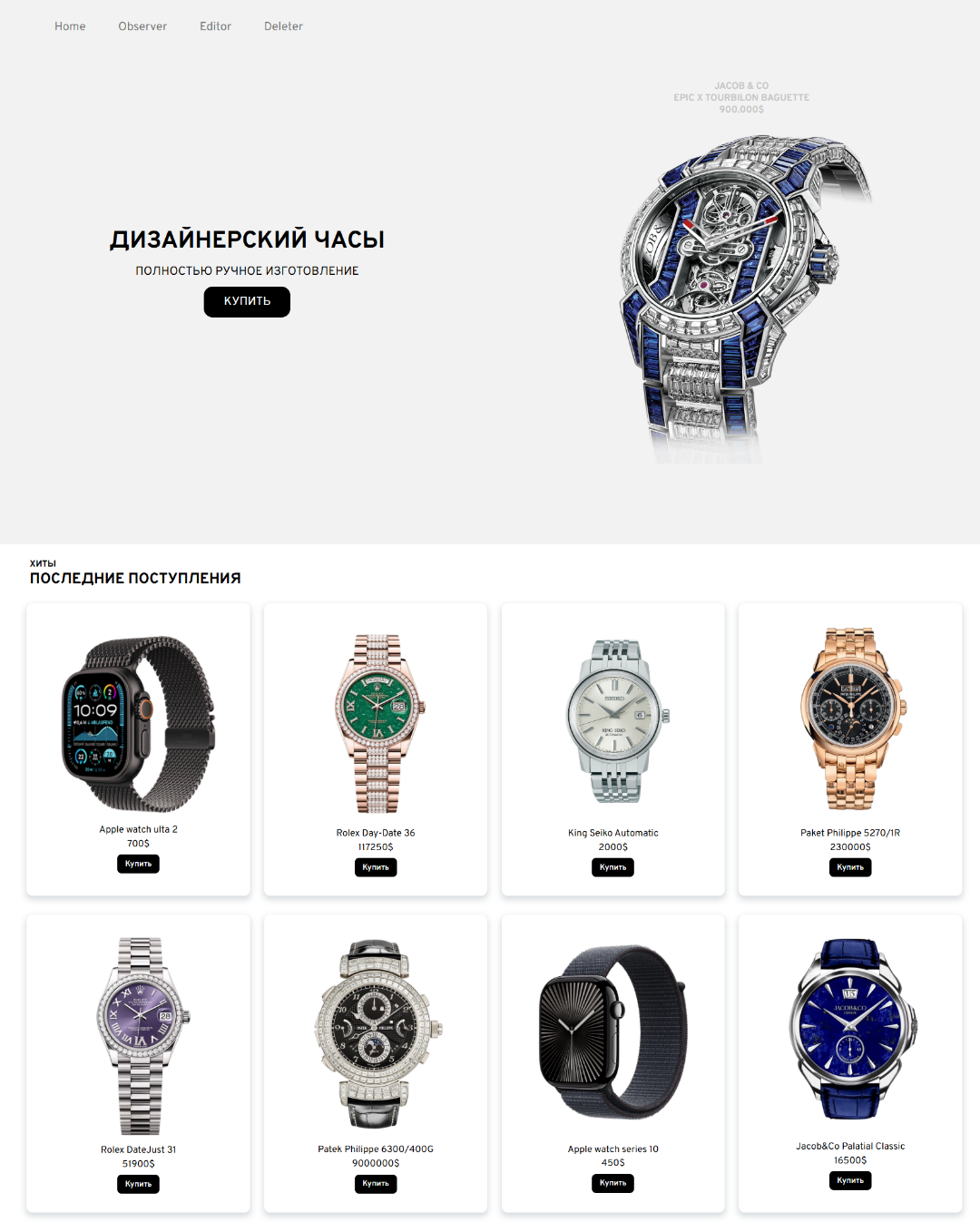


Рисунок 2. Созданное веб-приложение. Главная страница.

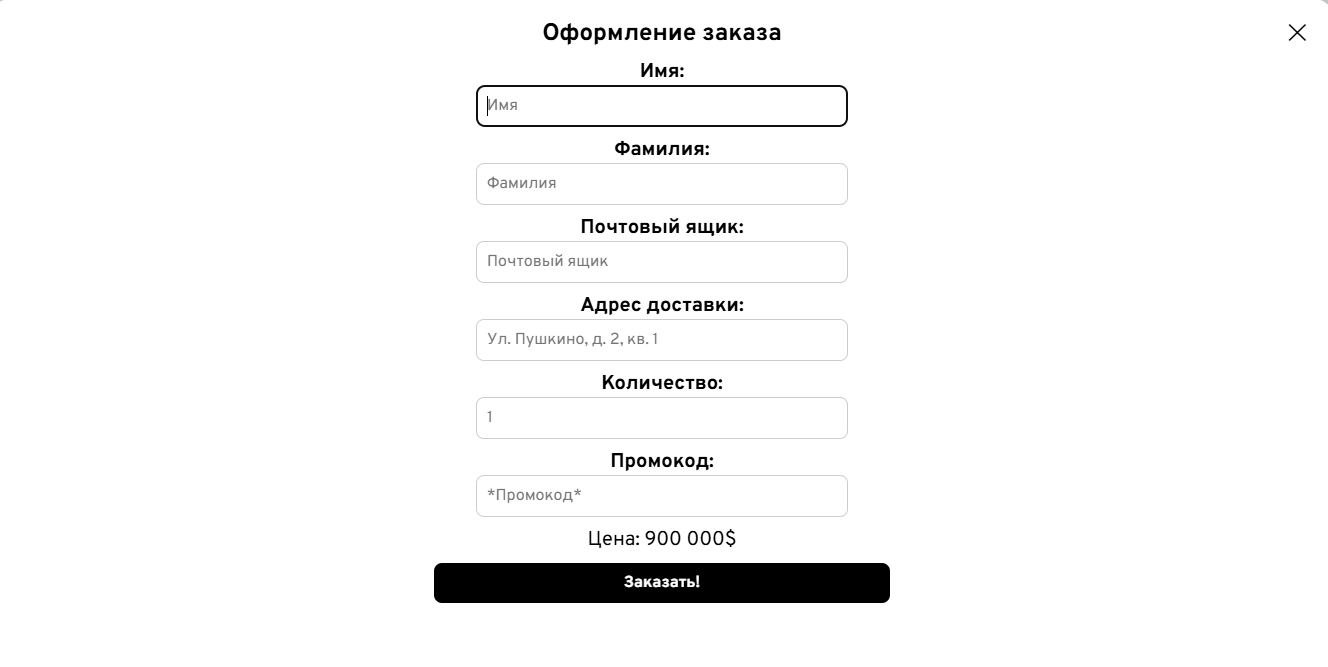


Рисунок 3. Форма добавления нового заказа.



Рисунок 4. Страница детального просмотра элемента.

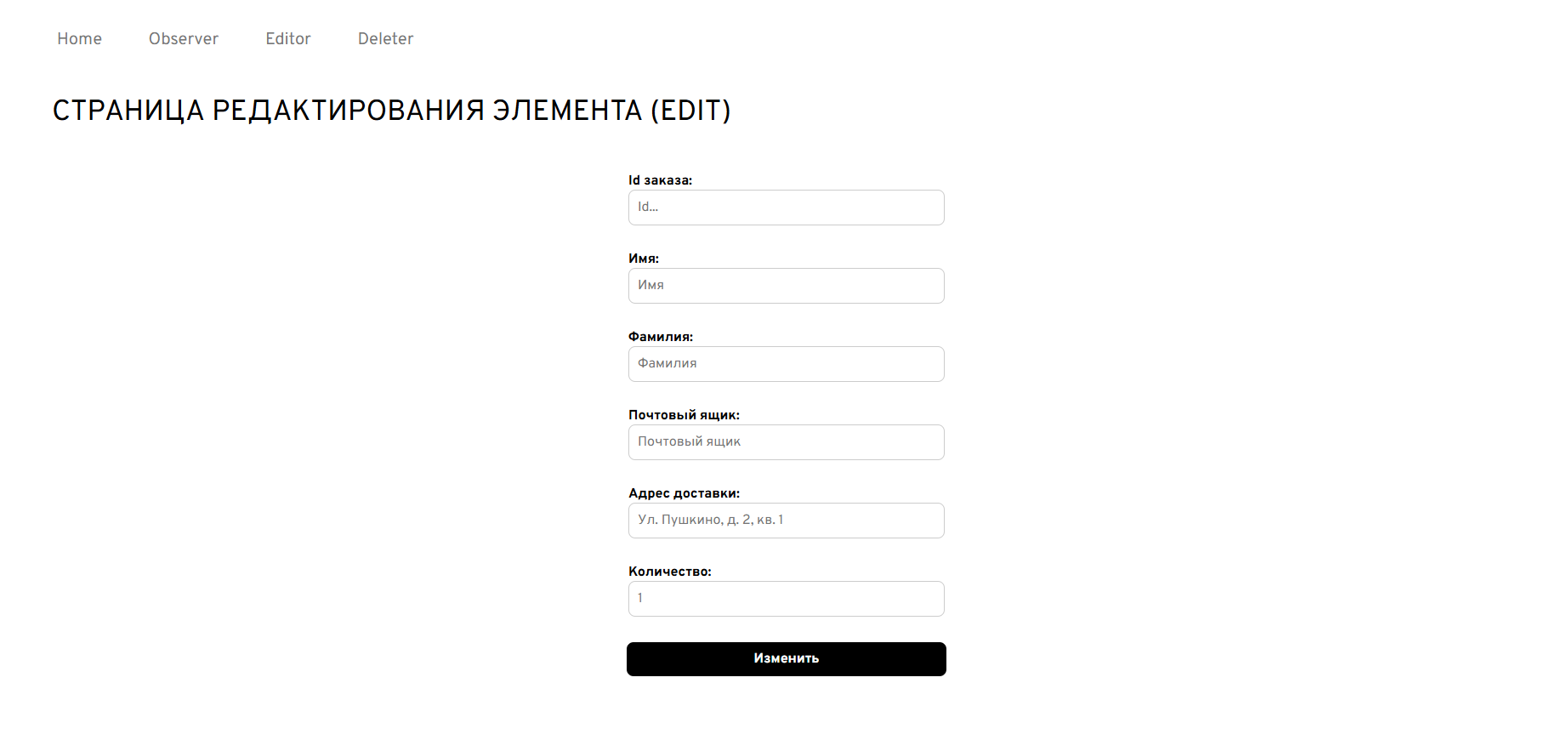


Рисунок 5. Страниц редактирования элемента.

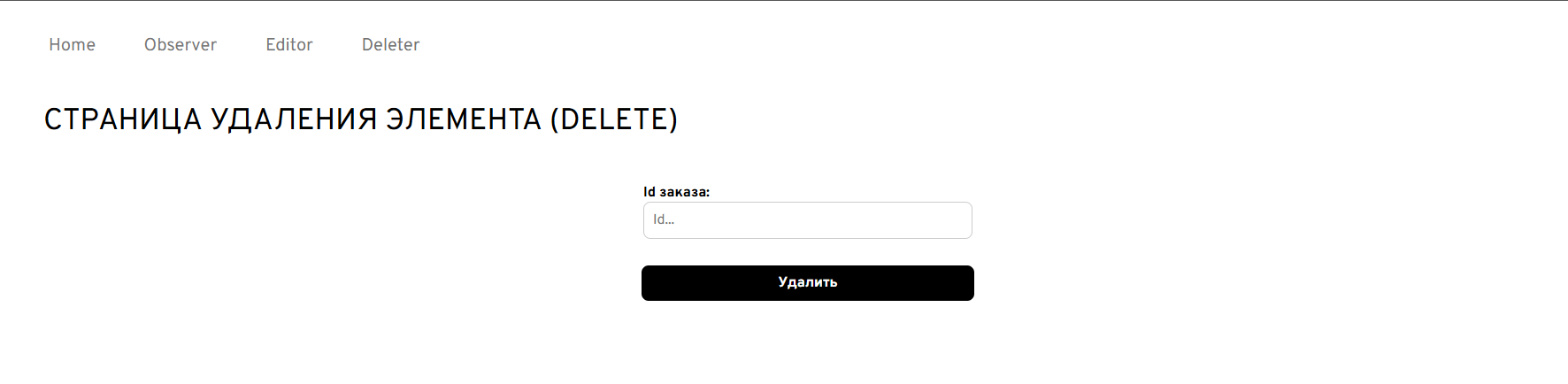


Рисунок 6. Страниц удаления элемента.

ВЫВОД К РАБОТЕ:

Я познакомился с технологией создания и использования шаблона проектирования MVC на платформе ASP.NET.