

LABORATORY ASSIGNMENT №2

Development with ASP.NET MVC. Project configuration. Displaying data from DB on web-pages.

Goal: Get acquainted with the MVC architecture, acquire skills in creating the main components of the MVC architecture, learn how to configure database connections and execute data reading queries, and acquire skills in working with views.

Workflow:

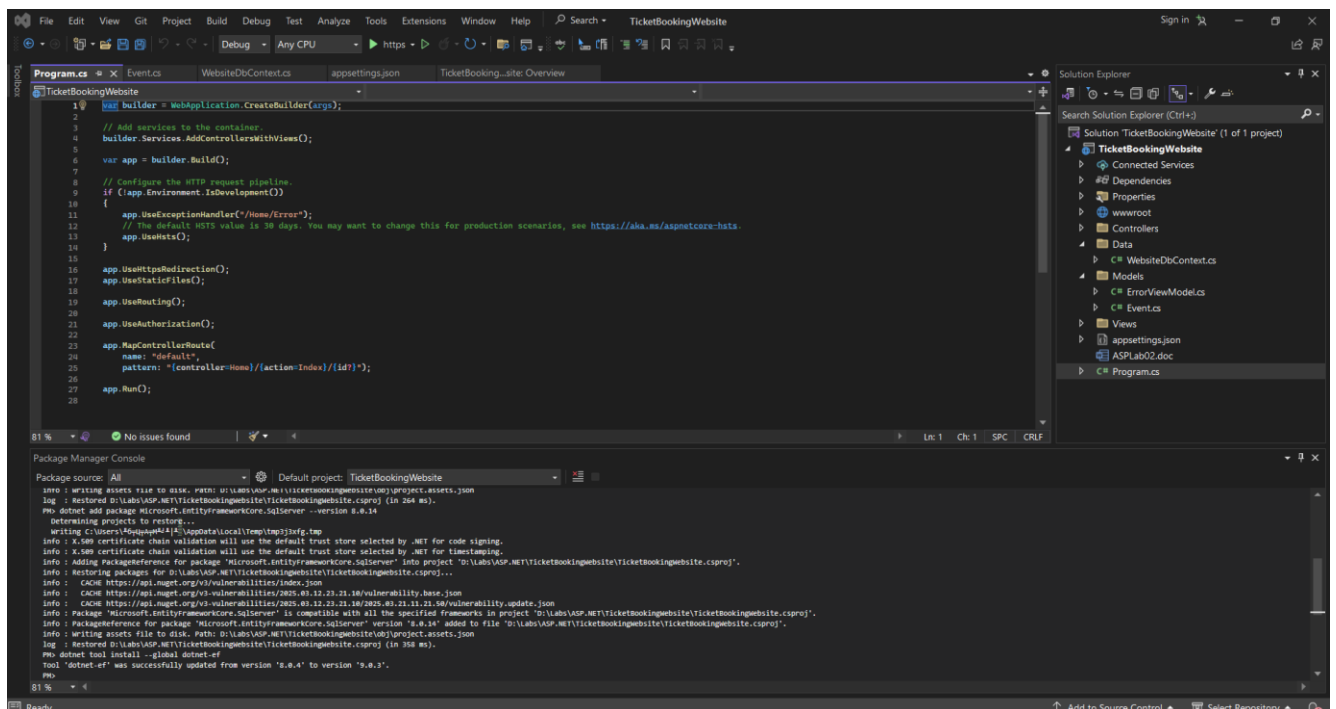
Task 1. Project creation and configuration

Number on group list	Topic	Short descriptions of functionalities
1	Event Ticket Booking Website	Users can search for events, book tickets, and manage bookings, while administrators can add and modify events.

Here is my topic

					ДУ «Житомирська політехніка».25.121.01.000 – Лр2								
Змн.	Арк.	№ докум.	Підпис	Дата									
Розроб.		Бейлах С.В.			Звіт з лабораторної роботи				Літ.	Арк.	Аркушів		
Перевір.		Українець М.О.									1	10	
Керівник									ФІКТ Гр. ІПЗ-23-3[1]				
Н. контр.													
Зав. каф.													

I've created project using visual studio gui.

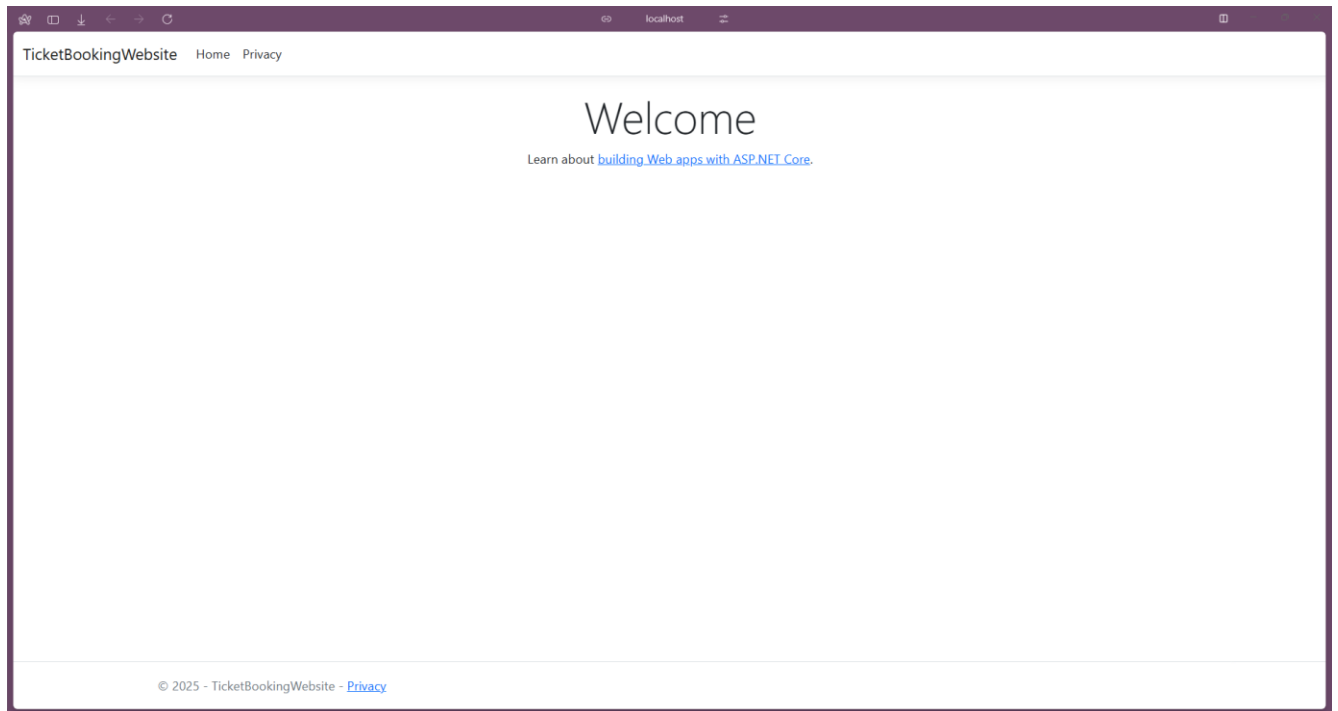


Here is project and its structure.

First model that I decided to create was an Event model, here is the code:

```
namespace TicketBookingWebsite.Models
{
    public class Event
    {
        public int Id { get; set; }
        public string Name { get; set; }
        public string Description { get; set; }
        public DateTime Date { get; set; }
        public string Location { get; set; }
        [Column(TypeName = "decimal(8, 2)")]
        public decimal Price { get; set; }
        public int AvailableTickets { get; set; }
    }
}
```

		Бейлах С.В.			ДУ «Житомирська політехніка».23.121.01.000 – Пр2	Арк.
		Українець М.О.				2
Змн.	Арк.	№ докум.	Підпис	Дата		



Running the project.

Task 2. ORM Setup

After setting up the project ive began setting up the orm. First step was creation of database in Microsoft SQL Server management studio. Then I’ve set up connection to database by adding connection string to config file:

```
"ConnectionStrings": {
  "SportsStoreConnection":
  "Server=(localdb)\\mssqllocaldb;Database=TicketBookingWebsite;MultipleActiveResultSets=true"
}
```

And then I’ve set up dbcontext:

```
using Microsoft.EntityFrameworkCore;
using TicketBookingWebsite.Models;

namespace SportsStore.Models
{
    public class WebsiteDbContext : DbContext
    {
        public WebsiteDbContext(DbContextOptions<WebsiteDbContext> options) :
        base(options) { }

        public DbSet<Event> Events => Set<Event>();
    }
}
```

		Бейлах С.В.			ДУ «Житомирська політехніка».23.121.01.000 – Лр2	Арк.
		Українець М.О.				3
Змн.	Арк.	№ докум.	Підпис	Дата		

After that I created repositories:

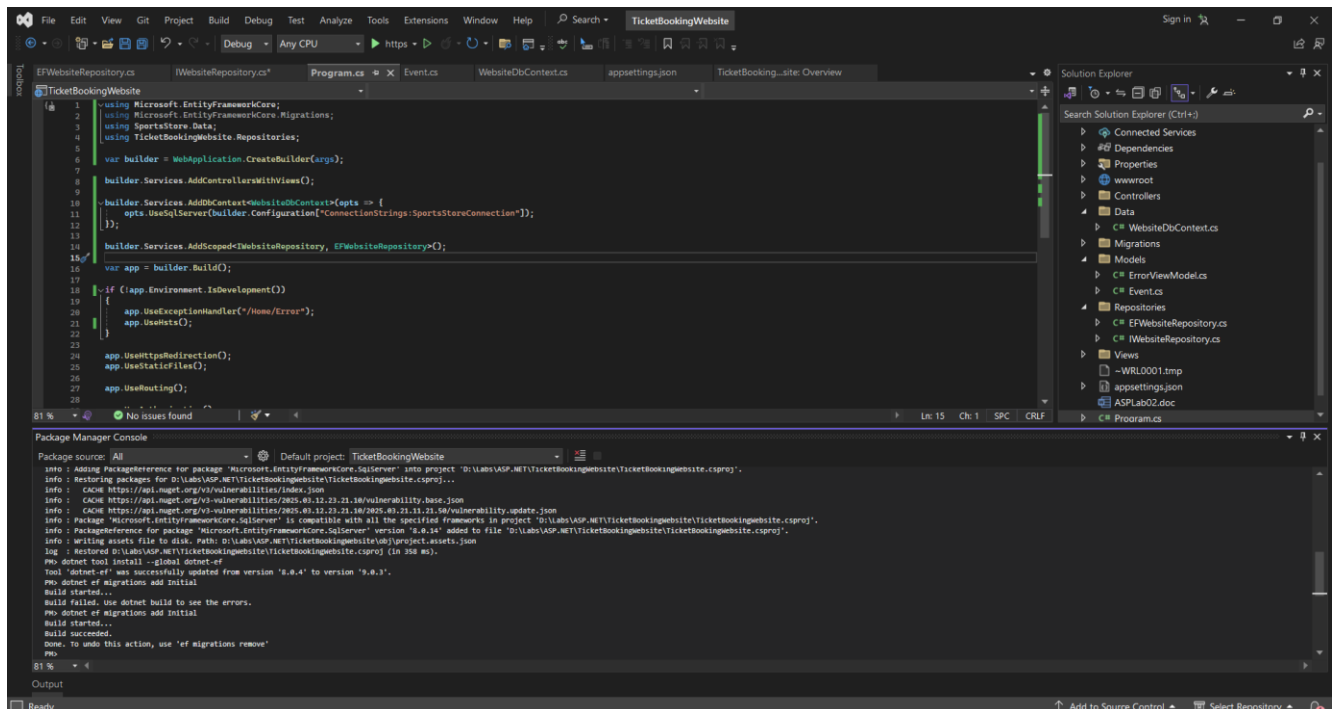
```
using Microsoft.EntityFrameworkCore.Migrations;
using SportsStore.Data;
using TicketBookingWebsite.Models;

namespace TicketBookingWebsite.Repositories
{
    public class EFWebsiteRepository : IWebsiteRepository
    {
        private WebsiteDbContext context;
        public EFWebsiteRepository(WebsiteDbContext ctx)
        {
            context = ctx;
        }
        public IQueryable<Event> Events => context.Events;
    }
}

using TicketBookingWebsite.Models;

namespace TicketBookingWebsite.Repositories
{
    public interface IWebsiteRepository
    {
        IQueryable<Event> Events { get; }
    }
}
```

Then I've applied some changes to program.cs and created migration:



		Бейлах С.В.			ДУ «Житомирська політехніка».23.121.01.000 – Пр2	Арк.
		Українець М.О.				4
Змн.	Арк.	№ докум.	Підпис	Дата		

Populate your database with sample data using the instructions below.

To complete this task I've created SeedData.cs:

```
using Microsoft.EntityFrameworkCore;
using Microsoft.Extensions.DependencyInjection;
using SportsStore.Data;
using System;
using System.Linq;

namespace TicketBookingWebsite.Models
{
    public static class SeedData
    {
        public static void EnsurePopulated(IApplicationBuilder app)
        {
            using (var serviceScope = app.ApplicationServices.CreateScope())
            {
                var context =
                    serviceScope.ServiceProvider.GetRequiredService<WebsiteDbContext>();

                if (context.Database.GetPendingMigrations().Any())
                {
                    context.Database.Migrate();
                }

                if (!context.Events.Any())
                {
                    context.Events.AddRange(
                        new Event
                        {
                            Name = "Rock Concert",
                            Description = "Live rock music event.",
                            Date = new DateTime(2025, 5, 20, 19, 0, 0),
                            Location = "Stadium A",
                            Price = 49.99m,
                            AvailableTickets = 200
                        },
                        new Event
                        {
                            Name = "Tech Conference",
                            Description = "Annual technology conference.",
                            Date = new DateTime(2025, 6, 10, 9, 0, 0),
                            Location = "Convention Center",
                            Price = 99.99m,
                            AvailableTickets = 500
                        },
                        new Event
                        {
                            Name = "Stand-up Comedy Show",
                            Description = "A night of laughter with top comedians.",
                            Date = new DateTime(2025, 4, 15, 20, 0, 0),
                            Location = "Comedy Club",
                            Price = 29.99m,
                            AvailableTickets = 150
                        },
                        new Event
                        {
                            Name = "Art Exhibition",
                            Description = "Showcasing modern art masterpieces.",
                            Date = new DateTime(2025, 7, 5, 10, 0, 0),
                            Location = "City Art Gallery",
                            Price = 15.00m,
                        }
                    );
                }
            }
        }
    }
}
```

		Бейлах С.В.			ДУ «Житомирська політехніка».23.121.01.000 – Пр2	Арк.
		Українець М.О.				5
Змн.	Арк.	№ докум.	Підпис	Дата		

}

After launching project my database was re-created and seeded with data:

[illegible]

Task 3. Displaying data on the web page

Firstly, to display data we need controller:

```
using Microsoft.AspNetCore.Mvc;
using Microsoft.EntityFrameworkCore.Migrations;
using TicketBookingWebsite.Repositories;

public class HomeController : Controller
{
    private IWebsiteRepository repository;

    public HomeController(IWebsiteRepository repo)
    {
        repository = repo;
    }

    public IActionResult Index() => View(repository.Events);
}
```

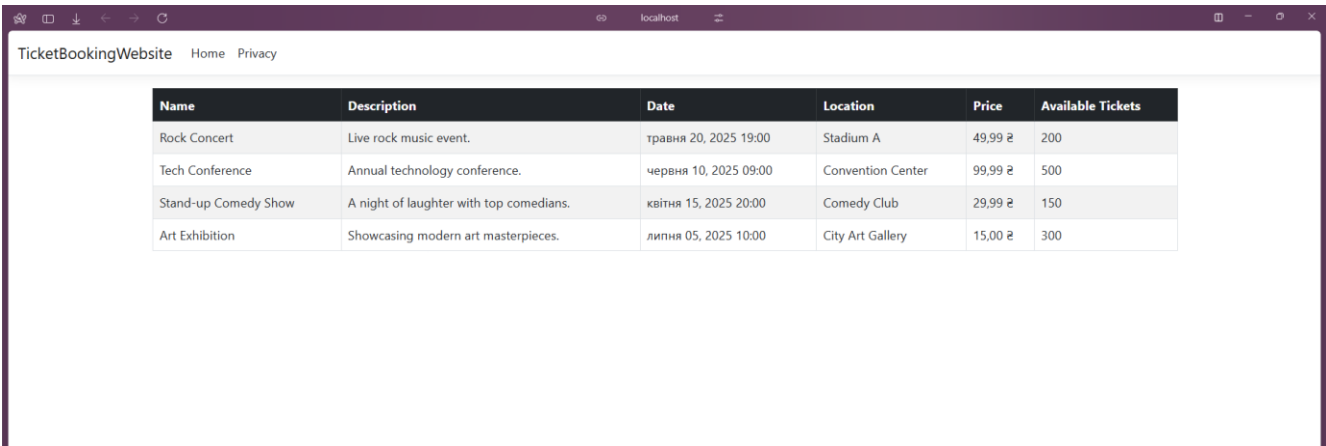
		Бейлах С.В.			ДУ «Житомирська політехніка».23.121.01.000 – Лр2	Арк.
		Українець М.О.				6
Змн.	Арк.	№ докум.	Підпис	Дата		

Then view:

```
@model IQueryable<Event>

<table class="table table-striped table-bordered">
  <thead class="table-dark">
    <tr>
      <th>Name</th>
      <th>Description</th>
      <th>Date</th>
      <th>Location</th>
      <th>Price</th>
      <th>Available Tickets</th>
    </tr>
  </thead>
  <tbody>
    @foreach (var e in Model ?? Enumerable.Empty<Event>())
    {
      <tr>
        <td>@e.Name</td>
        <td>@e.Description</td>
        <td>@e.Date.ToString("MMMM dd, yyyy HH:mm")</td>
        <td>@e.Location</td>
        <td>@e.Price.ToString("c")</td>
        <td>@e.AvailableTickets</td>
      </tr>
    }
  </tbody>
</table>
```

Here is the result:



Task 4. Implement paging

To do it I created 2 viewmodels:

```
using System.Collections.Generic;
using TicketBookingWebsite.Models;

namespace TicketBookingWebsite.ViewModels
{
    public class EventListViewModel
    {
        public IEnumerable<Event> Events { get; set; }
        public PagingInfo PagingInfo { get; set; }
    }
}
```

And

```
namespace TicketBookingWebsite.ViewModels
{
    public class PagingInfo
    {
        public int TotalItems { get; set; }
        public int ItemsPerPage { get; set; }
        public int CurrentPage { get; set; }
        public int TotalPages => (int)Math.Ceiling((decimal)TotalItems / ItemsPerPage);
    }
}
```

Then modified controller and view to ensure pagination:

```
using Microsoft.AspNetCore.Mvc;
using System.Linq;
using TicketBookingWebsite.Repositories;
using TicketBookingWebsite.ViewModels;

public class HomeController : Controller
{
    private readonly IWebsiteRepository repository;
    private const int PageSize = 2; // Change as needed

    public HomeController(IWebsiteRepository repo)
    {
        repository = repo;
    }

    public IActionResult Index(int page = 1)
    {
        var totalEvents = repository.Events.Count();

        var events = repository.Events
            .OrderBy(e => e.Date)
            .Skip((page - 1) * PageSize)
            .Take(PageSize);

        var viewModel = new EventListViewModel
        {
            Events = events,
            PagingInfo = new PagingInfo
            {

```

		Бейлах С.В.			ДУ «Житомирська політехніка».23.121.01.000 – Лр2	Арк.
		Українець М.О.				8
Змн.	Арк.	№ докум.	Підпис	Дата		


```

        CurrentPage = page,
        TotalItems = totalEvents,
        ItemsPerPage = PageSize
    };

    return View(viewModel);
}
}

```

```

@model TicketBookingWebsite.ViewModels.EventListViewModel
@using System.Globalization

```

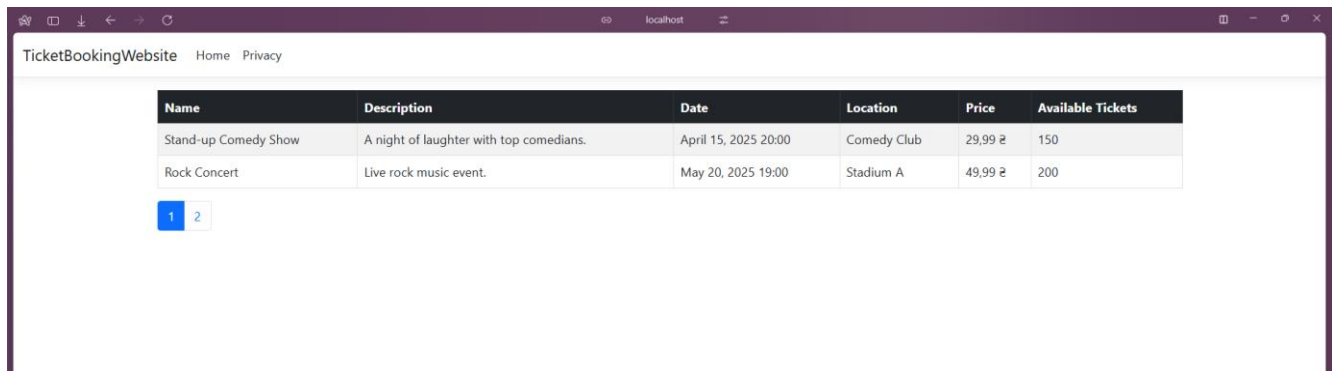
```

<table class="table table-striped table-bordered">
    <thead class="table-dark">
        <tr>
            <th>Name</th>
            <th>Description</th>
            <th>Date</th>
            <th>Location</th>
            <th>Price</th>
            <th>Available Tickets</th>
        </tr>
    </thead>
    <tbody>
        @foreach (var e in Model.Events)
        {
            <tr>
                <td>@e.Name</td>
                <td>@e.Description</td>
                <td>@e.Date.ToString("MMMM dd, yyyy HH:mm",
CultureInfo.InvariantCulture)</td>
                <td>@e.Location</td>
                <td>@e.Price.ToString("c")</td>
                <td>@e.AvailableTickets</td>
            </tr>
        }
    </tbody>
</table>

<!-- Pagination Controls -->
<nav>
    <ul class="pagination">
        @for (int i = 1; i <= Model.PagingInfo.TotalPages; i++)
        {
            <li class="page-item @(i == Model.PagingInfo.CurrentPage ? "active" : "")">
                <a class="page-link" asp-action="Index" asp-route-page="@i">@i</a>
            </li>
        }
    </ul>
</nav>

```

Result:



Name	Description	Date	Location	Price	Available Tickets
Stand-up Comedy Show	A night of laughter with top comedians.	April 15, 2025 20:00	Comedy Club	29.99 ₪	150
Rock Concert	Live rock music event.	May 20, 2025 19:00	Stadium A	49.99 ₪	200

1 2

Task 5. Styling

I've already had bootstrap installed and styling applied, so that's it.

Conclusion: I acquainted with the MVC architecture, acquired skills in creating the main components of the MVC architecture, learned how to configure database connections and executed data reading queries, and acquired skills in working with views.

		Бейлах С.В.			ДУ «Житомирська політехніка».23.121.01.000 – Лр2	Арк.
		Українець М.О.				10
Змн.	Арк.	№ докум.	Підпис	Дата		