```csharp
using Microsoft.AspNetCore.Builder;
using Microsoft.AspNetCore.Hosting;
using Microsoft.AspNetCore.HttpsPolicy;
using Microsoft.EntityFrameworkCore;
using Microsoft.Extensions.Configuration;
using Microsoft.Extensions.DependencyInjection;
using Microsoft.Extensions.Hosting;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using UI.BL.Inerface;
using UI.BL.Mapper;
using UI.BL.Repoistery;

using UI.DAL.DataBase;

namespace UI
{
    public class Startup
    {
        public Startup(IConfiguration configuration)
        {
            Configuration = configuration;
        }

        public IConfiguration Configuration { get; }

        // This method gets called by the runtime. Use this method to add
services to the container.
        public void ConfigureServices(IServiceCollection services)
        {
            services.AddControllersWithViews();
services.AddControllersWithViews();
            services.AddScoped<IUser, UserRep>();
            services.AddScoped<IViewer, Viewer>();
            services.AddScoped<IViewer, Viewer>();
            services.AddScoped<IFavorit, FavouritRep>();
            services.AddScoped<IFavorit, FavouritRep>();
            services.AddScoped<IAdmin, AdminRep>();
            services.AddScoped<IAdmin, AdminRep>();


            services.AddScoped<IBlockUserRep, BlockUserRep>();




            services.AddAutoMapper(x => x.AddProfile(new DomainProfile()));
            services.AddDistributedMemoryCache();
            services.AddSession();
```

```csharp
        services.AddDbContextPool<Dbcontiner>(opts =>
        opts.UseSqlServer(Configuration.GetConnectionString("SQlcon"))
        );


    }

    // This method gets called by the runtime. Use this method to configure
the HTTP request pipeline.
    public void Configure(IApplicationBuilder app, IWebHostEnvironment env)
    {
        if (env.IsDevelopment())
        {
            app.UseDeveloperExceptionPage();
        }
        else
        {
            app.UseExceptionHandler("/Home/Error");
            // The default HSTS value is 30 days. You may want to change this
for production scenarios, see https://aka.ms/aspnetcore-hsts.
            app.UseHsts();
        }
        app.UseHttpsRedirection();
        app.UseStaticFiles();
        app.UseSession();

        app.UseRouting();

        app.UseAuthorization();

        app.UseEndpoints(endpoints =>
        {
            endpoints.MapControllerRoute(
                name: "default",
                pattern: "{controller=User}/{action=Index}/{id?}");
        });
    }
}
}
```