AppDbContext.cs

```
using Application.Common.Interfaces;
using Domain;

using MediatR;

using Microsoft.AspNetCore.Http;
using Microsoft.EntityFrameworkCore;

namespace Infrastructure.Common.Persistence;

public class AppDbContext(DbContextOptions options, IHttpContextAccessor
_httpContextAccessor, IPublisher _publisher) : DbContext(options), IAppDbContext
{
    public virtual DbSet<ConfiguredPokemon> ConfiguredPokemons { get; set; }

    public virtual DbSet<ConfiguredMove> ConfiguredPokemonMoves { get; set; }

    public virtual DbSet<Team> TeamPresets { get; set; }

    public async override Task<int> SaveChangesAsync(CancellationToken
cancellationToken = default)
    {
        return await base.SaveChangesAsync(cancellationToken);
    }

    protected override void OnModelCreating(ModelBuilder modelBuilder)
    {

modelBuilder.ApplyConfigurationsFromAssembly(typeof(AppDbContext).Assembly);

        modelBuilder.Entity<ConfiguredPokemon>(entity =>
        {
            entity.ToTable("ConfiguredPokemon");

            entity.HasKey(e => e.Id);

            entity.Property(e => e.Id).HasColumnName("ConfiguredPokemonId");
            entity.Property(e => e.Name).HasColumnName("Name")
                .HasMaxLength(50)
                .IsUnicode(false);
            entity.Property(e => e.HP).HasColumnName("HP");
            entity.Property(e => e.Attack).HasColumnName("Attack");
            entity.Property(e => e.Defense).HasColumnName("Defense");
            entity.Property(e => e.SpecialAttack).HasColumnName("SpecialAttack");
            entity.Property(e =>
e.SpecialDefense).HasColumnName("SpecialDefense");
            entity.Property(e => e.Speed).HasColumnName("Speed");
            entity.Property(e => e.AbilityId).HasColumnName("AbilityId");
            entity.Property(e => e.TeamId).HasColumnName("TeamPresetId");
            // IV Properties
            entity.Property(e => e.HpIv).HasColumnName("HpIv");
            entity.Property(e => e.AttackIv).HasColumnName("AttackIv");
            entity.Property(e => e.DefenseIv).HasColumnName("DefenseIv");
            entity.Property(e =>
e.SpecialAttackIv).HasColumnName("SpecialAttackIv");
            entity.Property(e =>
e.SpecialDefenseIv).HasColumnName("SpecialDefenseIv");
            entity.Property(e => e.SpeedIv).HasColumnName("SpeedIv");

            // EV Properties
            entity.Property(e => e.HpEv).HasColumnName("HpEv");
            entity.Property(e => e.AttackEv).HasColumnName("AttackEv");
            entity.Property(e => e.DefenseEv).HasColumnName("DefenseEv");
            entity.Property(e =>
e.SpecialAttackEv).HasColumnName("SpecialAttackEv");
            entity.Property(e =>
e.SpecialDefenseEv).HasColumnName("SpecialDefenseEv");
            entity.Property(e => e.SpeedEv).HasColumnName("SpeedEv");

            entity.HasMany(p => p.Moves)
                .WithOne()
                .HasForeignKey(m => m.ConfiguredPokemonId);
        });

        modelBuilder.Entity<ConfiguredMove>(entity => {
                entity.ToTable("ConfiguredPokemonMoves");

                entity.HasKey(e => e.Id);

                entity.Property(e => e.Id).HasColumnName("PokemonMoveId");
                entity.Property(e => e.MoveId).HasColumnName("MoveId");
                entity.Property(e =>
e.ConfiguredPokemonId).HasColumnName("ConfiguredPokemonId");
            }
        );


        modelBuilder.Entity<Team>(entity =>
        {
            entity.ToTable("TeamPreset");

            entity.HasKey(e => e.Id);

            entity.Property(e => e.Id).HasColumnName("TeamPresetId");
            entity.Property(e => e.Name).HasColumnName("PresetName");
            entity.Property(e => e.TrainerId).HasColumnName("UserId");

            entity.HasMany(e => e.Pokemon).WithOne().HasForeignKey(p=> p.TeamId);
        });


        base.OnModelCreating(modelBuilder);
    }
```

```
}
```

Path: `./Backend/Infrastructure/Common/Persistence/AppDbContext.cs`