Tp 3 – Logical Layer

Exercice

- Créer une business layer pour votre jeu
  - Faire en sorte que les plateaux soit « mis en mémoire » uniquement dans cette layer
  - Mettre en place un « système » pour que votre partie n'accepte QUE 2 joueurs

Exercice (optionnel)

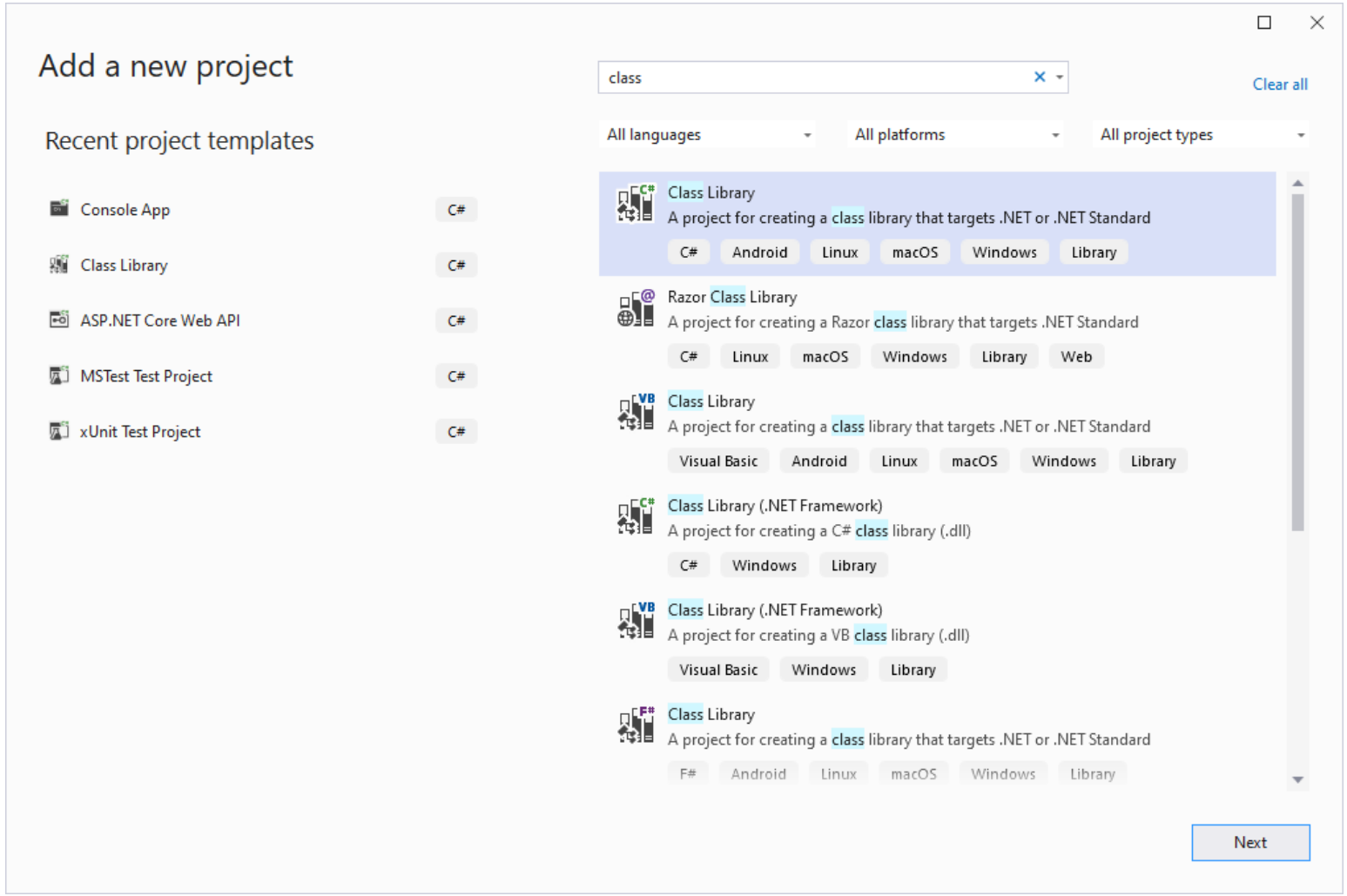
- Créer une bibliothèque de Test Unitaire
  - Tester que toutes vos méthodes dans la business layer précédemment créée.

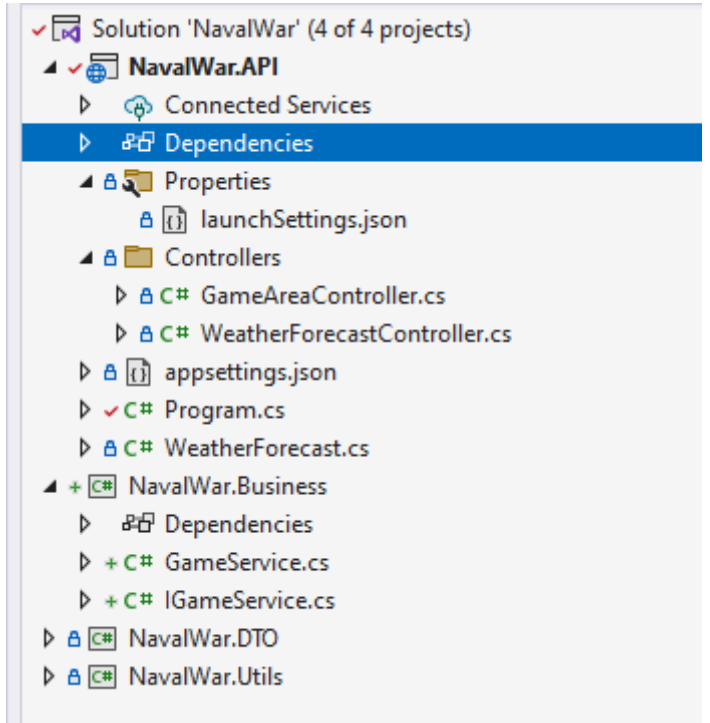
Autres librairies

- Add FluentValidation Nuget
- Pour chaque action critique, ajouter une étape de validation via FluentValidation afin d'assurer que les critères sont bons avant l'action.
- https://fluentvalidation.net/

Tips

BusinessLayer




Solution 'NavalWar' (4 of 4 projects)
NavalWar.API
Connected Services
Dependencies
Properties
launchSettings.json
Controllers
GameAreaController.cs
WeatherForecastController.cs
appsettings.json
Program.cs
WeatherForecast.cs
NavalWar.Business
Dependencies
GameService.cs
IGameService.cs
NavalWar.DTO
NavalWar.Utils

```
using NavalWar.Business;

var builder = WebApplication.CreateBuilder(args);

// Add services to the container.

builder.Services.AddControllers();
// Learn more about configuring Swagger/OpenAPI at https://aka.ms/aspnetcore/swashbuckle
builder.Services.AddEndpointsApiExplorer();
builder.Services.AddSwaggerGen();

//add your dependencies
builder.Services.AddScoped<IGameService, GameService>();

var app = builder.Build();

// Configure the HTTP request pipeline.
if (app.Environment.IsDevelopment())
{
    app.UseSwagger();
    app.UseSwaggerUI();
}

app.UseHttpsRedirection();

app.UseAuthorization();

app.MapControllers();

app.Run();
```

```
namespace NavalWar.API.Controllers
{
    [Route("api/[controller]")]
    [ApiController]
    1 reference | Pierre-Loic Chevillot, 32 days ago | 1 author, 2 changes
    public class GameAreaController : ControllerBase
    {
        private GameArea _area = new GameArea();
        private readonly IGameService _gameService;

        0 references | 0 changes | 0 authors, 0 changes
        public GameAreaController(IGameService gameService)
        {
            _gameService = gameService;
        }
```

IActionResult

- https://docs.microsoft.com/fr-fr/aspnet/core/web-api/action-return-types?view=aspnetcore-5.0