

## Modul 02: ASP.NET Core - Konfiguration

*Ziel: Verwende Lebenszyklen für Objekten innerhalb ASP.NET Core*

*Systemvoraussetzungen: .Net 6.0*

*Umsetzbar: RazorPages, MVC, Web API, gRPC, Blazor Server, SPAs*

*Tools: Visual Studio 2022*

*Dauer: ~30min*

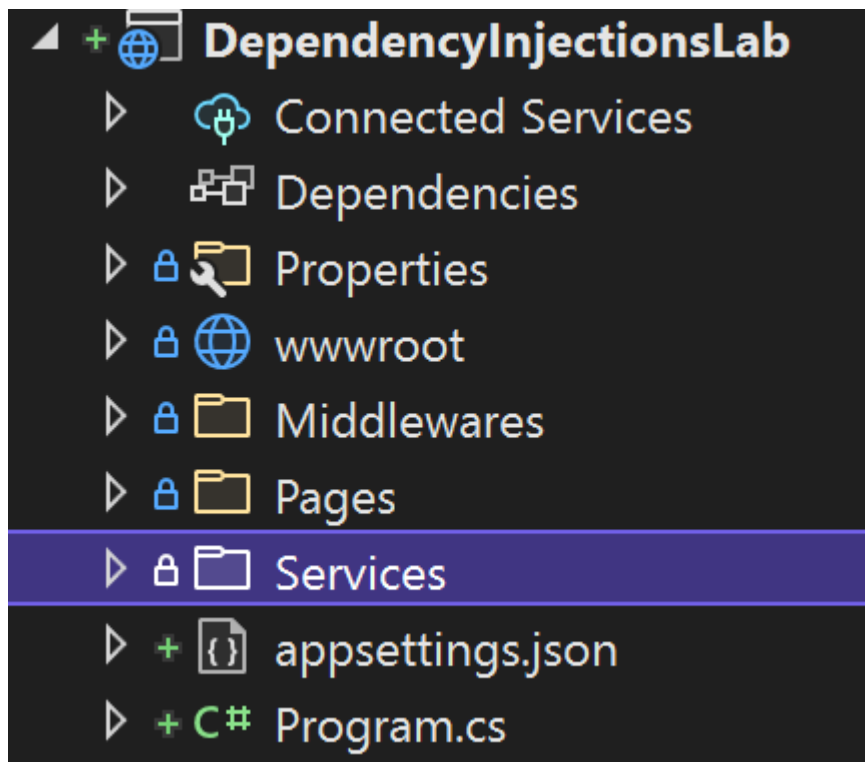
*Autor: Kevin Winter*

*Letzte Änderung: 16.12.21*

### 1 Vorbereitung

Das Beispiel demonstriert eine ASP.NET Core Funktion und kann daher mit allen ASP.NET Core-Templates umgesetzt werden.

- 1.) Erstelle in deinem ASP.NET Core Projekt das Verzeichnis Services



- 2.) Erstelle innerhalb des Verzeichnis Services ein Interface mit dem Namen IOperation.cs und füge folgende Interfaces ein:

```

3 references | 0 changes | 0 authors, 0 changes
public interface IOperation
{
    8 references | 0 changes | 0 authors, 0 changes
    string OperationId { get; }
}

6 references | 0 changes | 0 authors, 0 changes
public interface IOperationTransient : IOperation { }
5 references | 0 changes | 0 authors, 0 changes
public interface IOperationScoped : IOperation { }
6 references | 0 changes | 0 authors, 0 changes
public interface IOperationSingleton : IOperation { }

```

- 3.) Erstelle innerhalb des Verzeichnis Services eine Klasse mit dem Namen Operation.cs und füge folgende Klasse ein:

```

public class Operation : IOperationTransient, IOperationScoped, IOperationSingleton
{
    0 references | 0 changes | 0 authors, 0 changes
    public Operation()
    {
        OperationId = Guid.NewGuid().ToString()[^4..];
    }

    8 references | 0 changes | 0 authors, 0 changes
    public string OperationId { get; }
}

```

## 2 Aufgabenstellung

- 1.) Registriere die Interface:
- a. IOperationSingleton als Singleton Lifetime
  - b. IOperationSclope als Sclope Lifetime
  - c. IOperationTransient als Transient Lifetime

in der Program.cs

- 2.) Zu Programmieren ist eine Methode, die ein Request Verarbeiten kann:
  - a. WebAPI -> WeatherForecast-Controller -> Get-Methode
  - b. MVC -> HomeController->Index-Methode
  - c. RazorPage->IndexModel->OnGet()
  - d. Blazor->
  - e. gRPC->Partial Service-Klasse -> RPC Service Methode
- 3.) Pro Aufruf soll jedes Interface die Property->**OperationId** in den Default-Logger schreiben
- 4.) Wenn der Logger nicht verfügbar ist, soll dieser auch via Dependency Injection hinzugefügt und verwendet werden.