

## ASP.NET MVC Labo Week 4

### Doelstelling:

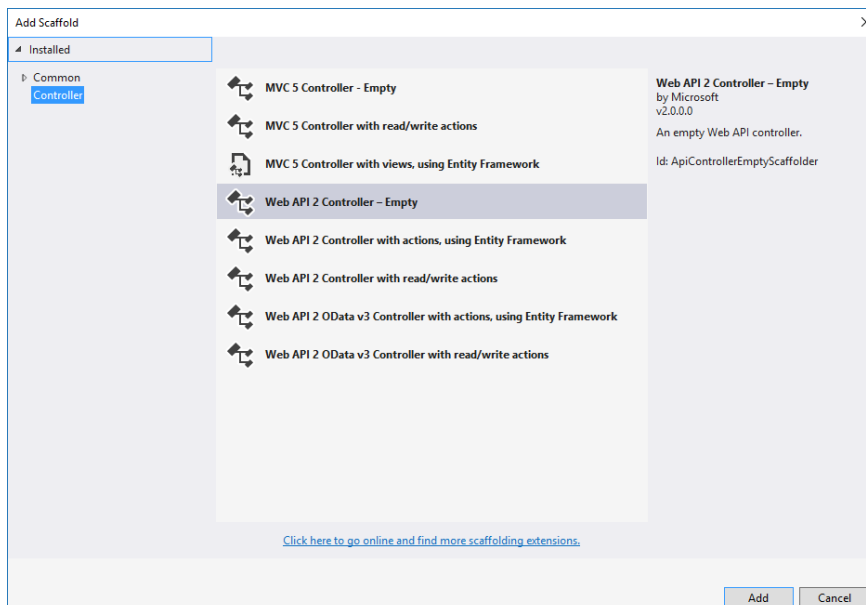
- Kennis maken met ASP.NET MVC Web API

### Oefening Web API

#### Configuratie

We gaan de oefening over sessies van vorig week uitbreiden met een REST API.

Kies in de map controllers rechtermuis => Add => Controller. Daarna kiezen we voor Web API 2 Controller en we noemen deze MobileController.



Daarna kan u volgende melding te zien krijgen. Gelieve deze te volgen.

Visual Studio has added the full set of dependencies for ASP.NET Web API 2 to project 'Oefening 2'

The Global.asax.cs file in the project may require additional changes to enable ASP.NET Web API.

1. Add the following namespace references:

```
using System.Web.Http;
using System.Web.Routing;
```

2. If the code does not already define an Application\_Start method, add the following method:

```
protected void Application_Start()
{
}
```

3. Add the following lines to the beginning of the Application\_Start method:

```
GlobalConfiguration.Configure(WebApiConfig.Register);
```

Daarna zou er een controller aanwezig zijn die er als volgt uitziet:

```
public class MobileController : ApiController
{
}
```

## GET Requests

De eerste methode die we toevoegen is

0 references

```
public List<Sessie> GetSessions()
{
    List<Sessie> all = new List<Sessie>();
    all.AddRange(Data.GetSessions(1));
    all.AddRange(Data.GetSessions(2));
    all.AddRange(Data.GetSessions(3));
    return all;
}
```

Deze methode zal alle sessies terugkeren. Test deze uit in de browser. Test zowel in Chrome als IE (of Edge). Welke URL moeten we gebruiken om deze methode te testen? Bekijk de route in "WebApiConfig.cs"

Als dit werkt gaan we de wijzigen. We gaan namelijk een route toevoegen die veel duidelijker is (en meer RESTFull). We kunnen dit doen door het attribuut [Route].

```
[HttpGet]
[Route("api/sessions")]
0 references
public List<Sessie> GetSessions()
{
    List<Sessie> all = new List<Sessie>();
    all.AddRange(Data.GetSessions(1));
    all.AddRange(Data.GetSessions(2));
    all.AddRange(Data.GetSessions(3));
    return all;
}
```

Je kan nu de REST service aanroepen via <http://localhost:19442/api/sessions/> wat veel duidelijker is. Test dit terug in de browser.

Maak nu volgende uitbreiding:

URL voor

- <http://localhost:19442/api/sessions/slot/1>
- <http://localhost:19442/api/sessions/slot/2>
- <http://localhost:19442/api/sessions/slot/...>
- <http://localhost:19442/api/organizations>
- <http://localhost:19442/api/registrations>

**Tip: pas eventueel uw code aan om registraties bij te houden. Maak hiervoor in Data.cs een lijst aan die static is en waarin je registraties kan opslaan.**

## POST Request

We wensen de mogelijkheid te voorzien in de API om organisaties toe te voegen. Toevoegen doen we via een Http POST verb. De route die we gebruiken is "api/organizations". Deze route bestaat reeds (zie hierboven), mag dit ? Wat is het verschil met de reeds bestaande?

We introduceren hier ook de HttpResponseMessage. Hiermee kunnen we een statuscode terugsturen. De keuze van de statuscode is niet altijd eenvoudig. Implementeer onderstaande code. Maak eventueel gebruik van <http://www.restapitutorial.com/httpstatuscodes.html>

```
[HttpPost]
[Route("api/organizations/")]
0 references
public HttpResponseMessage AddOrganization(Organization newOrganization)
{
    if (newOrganization == null)
        return new HttpResponseMessage(HttpStatusCode.InternalServerError);

    Data.AddOrganization(newOrganization);
    return new HttpResponseMessage(HttpStatusCode.Created);
}
```

Maak gebruik van Fiddler om dit te testen (weet u waarom? ik hoop van wel). Hoe moet je de nieuwe organisatie doorsturen, welk formaat?

## DEL Requests

We wensen de mogelijkheid om een organisatie te verwijderen? Denk na over volgende zaken

- Welk HTTP verb moeten we hiervoor gebruiken?
- Wat is de route?
- Hoe geven we mee welke organisatie we wensen te verwijderen?
- Wat is de terugkeerwaarde van deze actie?
- Wat als we een organisatie wensen te verwijderen die niet bestaat, wat moeten we dan terugkeren?

Probeer dit zelfstandig te implementeren. Probeer dit te testen in de browser, werkt dit? Indien dit niet werkt wat kan je dan wel gebruiken om te testen?

## PUT Request

Werk nu volledig zelfstandig de mogelijkheid uit om een organisatie te wijzigen.