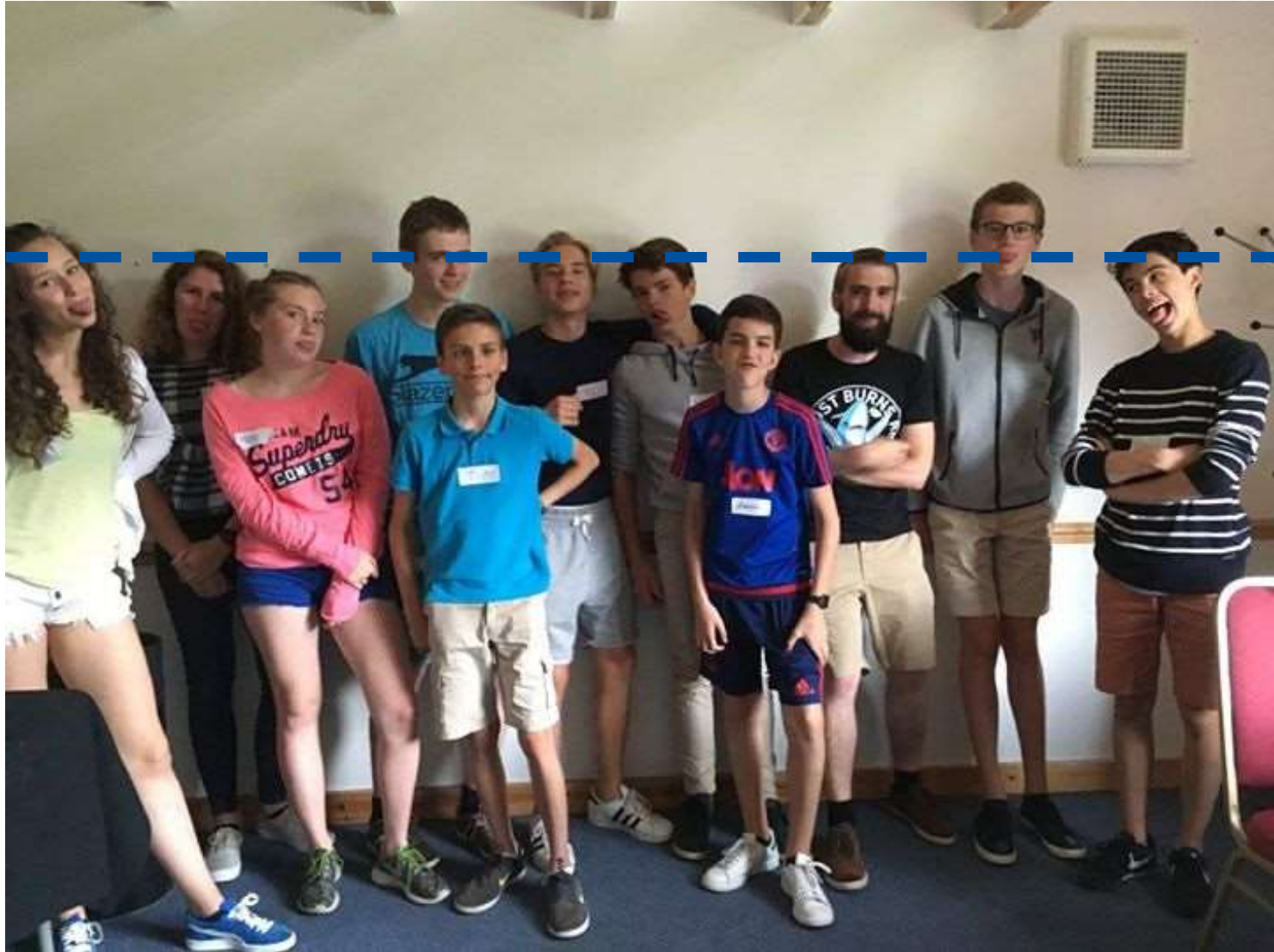


**MVC**

**#ISKDG**









# Disclaimer



**HTML**



TypeScript



ANGULARJS

by Google



git



UNIVÉ  
VERZEKERINGEN



attentia



Napoleon Games



ergo:.



National Bank  
OF BELGIUM

PETRONAS



erkend sociaal secretariaat  
info@sodiwe.be - www.sodiwe.be



Nerd? I prefer the term  
"Intellectual Badass"



som<sup>ee</sup>cards  
user card





# Wat is jouw favoriete technologie/framework



## Start the presentation to activate live content

If you see this message in presentation mode, install the add-in or get help at [Pollev.com/app](https://Pollev.com/app)

# Content

---

- Tips and tricks voor de setup
- Razor to the max!
- CQRS
  - Alternatief voor de laagjes architectuur
- MVC pipeline
  - Of .. hoe kan ik nog meer uit MVC halen

# Wat is MVC?

**Start the presentation to activate live content**

If you see this message in presentation mode, install the add-in or get help at [PollEv.com/app](https://PollEv.com/app)

# MVC

- Simpel
  - C# -> request afhandelen
  - Razor -> Html maken met C#
- Pro tip:
  - UPDATE NUGET PACKAGES ASAP!
- Gebruik T4MVC
  - Weg met magic strings!





# Razor

- C# component om html te generen
  - Strongly typed
  - Build errors
- Tip:
  - MVC build views!

```
<TargetFrameworkVersion>v4.6.1</TargetFrameworkVersion>  
<UseIISExpress>false</UseIISExpress>  
<MvcBuildViews>false</MvcBuildViews>  
<IISExpressSSLPort />  
<IISExpressAnonymousAuthentication />
```

- IIS > IIS express (sneller debuggen)
  - Web platform installer -> zoek naar 'IIS recommended configuration'

# DisplayTemplates

- Magic folder
  - DisplayTemplates
- Scope afhankelijk van de locatie
  - Shared vs Controller folder
- Op basis van conventie
  - @model <TYPE> bovenaan
  - Filename == typename
- Gebruik
  - @Html.DisplayFor(m => m.MyProp)
  - @Html.DisplayForModel()

# EditorTemplates

- Zelfde systeem
  - Supersnel formulieren bouwen
  - @Html.IdFor/@Html.NameFor gebruiken!
  - Gebruik
    - @Html.EditorFor(m => m.MyProp)
    - @Html.EditorForModel()
- Ga slim om met het bouwen van ViewModels  
(focus op hergebruik)

- Command Query Responsibility Segregation
- TL;DR
  - Queries apart
  - Wijzigingen apart



# MVC Pipeline

# Filters

- Herbruikbaar
- Voor en/of na elke request
- Scope goed te kiezen
- Breed inzetbaar
  - Logging
  - Redirects
  - Authenticatie
  - ....

# Filters

- **OnActionExecuting**
  - Voor de actie
- **OnActionExecuted**
  - Na de actie
- **OnResultExecuting**
  - Voor het uitvoeren van het ActionResult
- **OnResultExecuted**
  - Na het uitvoeren van het ActionResult (bv view renderen)

# EF Migrations

- Sql update scripts genereren op basis van code
- Elke migration heeft een snapshot van de code
  - Wat met team?  
Add-migration MergeMigration –IgnoreChanges
- Houdt rekening met alle conventions en config van je DbContext
- Voor je eigen peace of mind  
→ in apart project



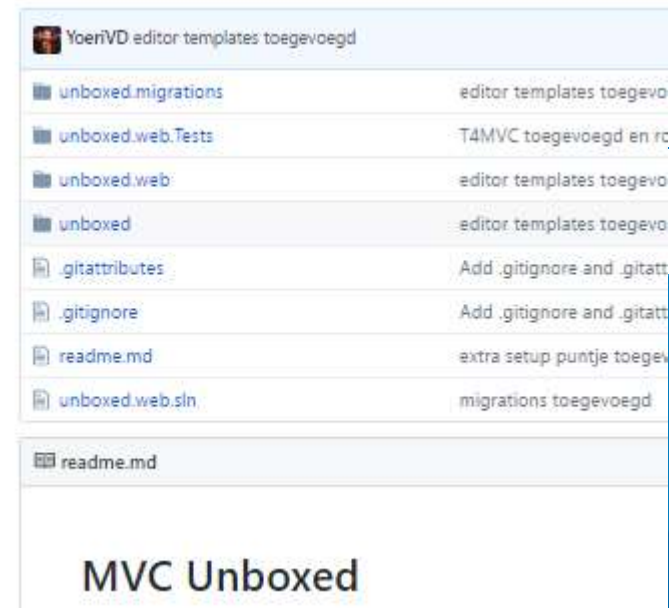
# En wat hebben we vandaag geleerd?

- T4MVC
- ViewModels + EditorTemplates
- Command en Query scheiden



# Resources

- [github.com/yoerivd/mvc-unboxed](https://github.com/yoerivd/mvc-unboxed)
- [mvc-unboxed.azurewebsites.net](http://mvc-unboxed.azurewebsites.net)
- [twitter.com/yoerivd](https://twitter.com/yoerivd)





Wat vond je zelf het meest interessant deel?



**Start the presentation to activate live content**



If you see this message in presentation mode, install the add-in or get help at [Pollev.com/app](https://Pollev.com/app)

# Zien we het project zitten?!



**Start the presentation to activate live content**

If you see this message in presentation mode, install the add-in or get help at [Pollev.com/app](https://Pollev.com/app)

# Vragen?

---