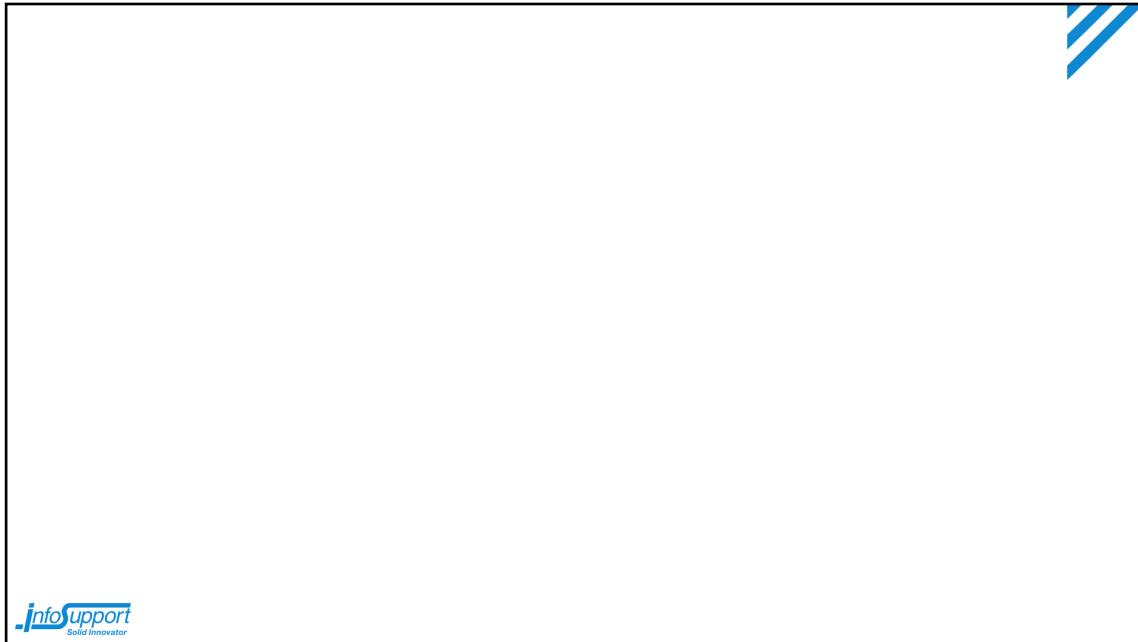


MVC, Git it?

The name is Yoeri

@yoerivd

everything web - git - CI/CD - javascript - C#





Contents

- MVC Hidden gems!
- Git? Pull request? Branches? REBASE? MERGE??
- Build your own Dependency Injection Container!

Getting started

New project

- file -> new project
- MVC
- Update NuGet packages
 - templates kunnen achterlopen

Voting buttons!

the plan

1. maak een feature branch
2. git push --set-upstream origin <branch>
3. pas de code aan
4. test
5. merge branch

EditorTemplates

1. maak een folder EditorTemplates in eender welke view (of in Shared voor elke view)
2. maak een nieuwe view: naamModel.cshtml
3. declareer het model type in de view

```
@model mvc_demo.web.Models.MyModel
```

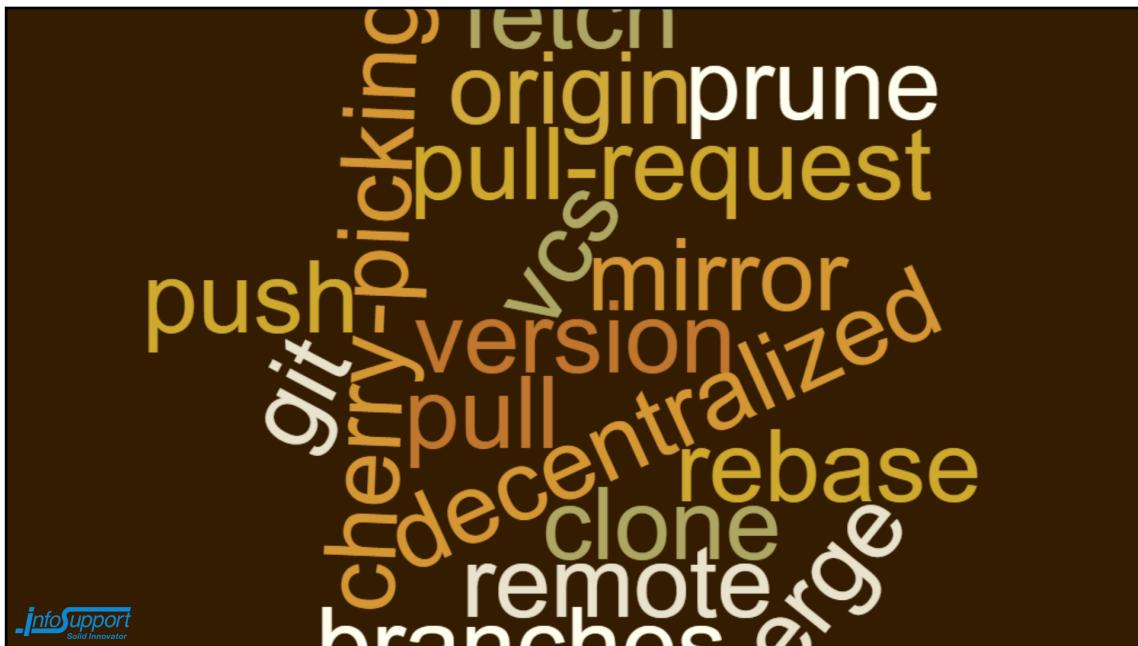
4. Gebruiker volgende helper waar gewenst:

```
@Html.EditorFor(model => model.PropertyVanHetTypeMyModel)
```

life saving tip #3

Voorzie het `id` en `name` attribuut van `input`-velden
altijd van een waarde met `@Html.NameFor()` en
`@Html.IdFor()`

Git



waarom VCS ?

versies beheren
samen werken
deployment

waarom git en geen
ander?

decentralized

snel!

geen connectie met server nodig

gebaseerd op commit history, niet op file history

freedom of branches!



DEMO

A cartoon illustration of Mr. Flanders, a balding man with glasses and a white lab coat, gesturing with his hands as if giving a presentation.

Life saving tip # 1

-> werk nooit rechtstreeks op de master branch
conflicten
lelijke history
master zou ten allen tijden moeten werken

*meeste VCS systemen kennen
'protected branches'*

Dure woordjes

Git	git it
repository	het gehele project dat in VCS zit
clone	een lokale kopie maken van het repo (om zelf te bewerken)
remote	git kan sync'en met verschillende 'remote' repositories tegelijk. Meestal github, gitlab, bitbucket,..
commit	wanneer je een wijziging aanbrengt kan je die opslaan in git, dit noemt men een commit

Git	git it
merge	wijzigingen van een andere branch ook op de huidige branch toepassen
rebase	alternatief voor merge, negeren tot later :-)
upstream	de naam van de branch op de remote die getracked mag worden
pull	wanneer je niet zelf een branch mag wijzigen, kan je een aanvraag doen
request	naar iemand die het wel mag

Life saving tip # 2

-> gebruik git cmd, geen ui

Getting sources

command	uitleg
git pull	get all commits from master and merge with the active local branch
git stash	"shelve" all pending changes (useful before git pull)
git stash pop	"unshelve" previously stashed changes
git remote	lokale remotes branches die niet meer op de server bestaan ook <u>lokale verwijderen</u>

Branching

command	uitleg
git checkout -b someFeatureOrBugFixName	create a new branch from the current branch and switch to the new branch
git checkout master	switch to a branch with the name master
git branch	list all branches present on this machine"
git branch -d someFeatureOrBugFixName	delete a branch with name "someFeatureOrBugFixName"

Committing changes

command	uitleg
git status	show unstaged and staged changes
git add *.cs	stage all unstaged files ending in ".cs"
git commit -am "some message"	add all staged changes, commit with a message

Merging

command	uitleg
git pull origin master	huidige branch up to date brengen met master
git mergetool	=> fix conflicts
git commit [...]	=> check in the merge
git push	=> sync with the master repository

Pull requests

pull request

- niet mogelijk vanuit cmd
- handig voor reviews
- build status



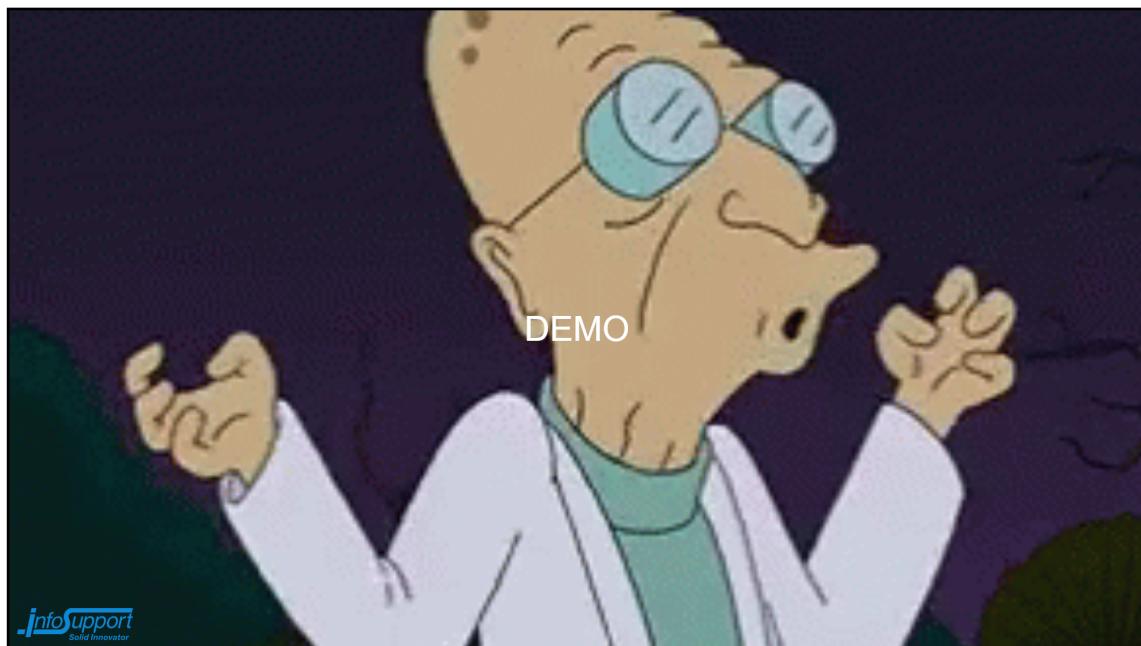
pull request

1. complete zorgt voor een merge op de server
2. life saving tip #4: delete source branch!

Display Templates

the plan

1. maak een feature branch
2. git push --set-upstream origin
<branch>
3. Beantwoorde vraag tonen
4. pull request maken



DisplayTemplates

1. maak een folder DisplayTemplates in eender welke view (of in Shared voor elke view)
2. maak een nieuwe view: naamModel.cshtml
3. declareer het model type in de view

```
@model mvc_demo.web.Models.MyModel
```

4. Gebruiker volgende helper waar gewenst:

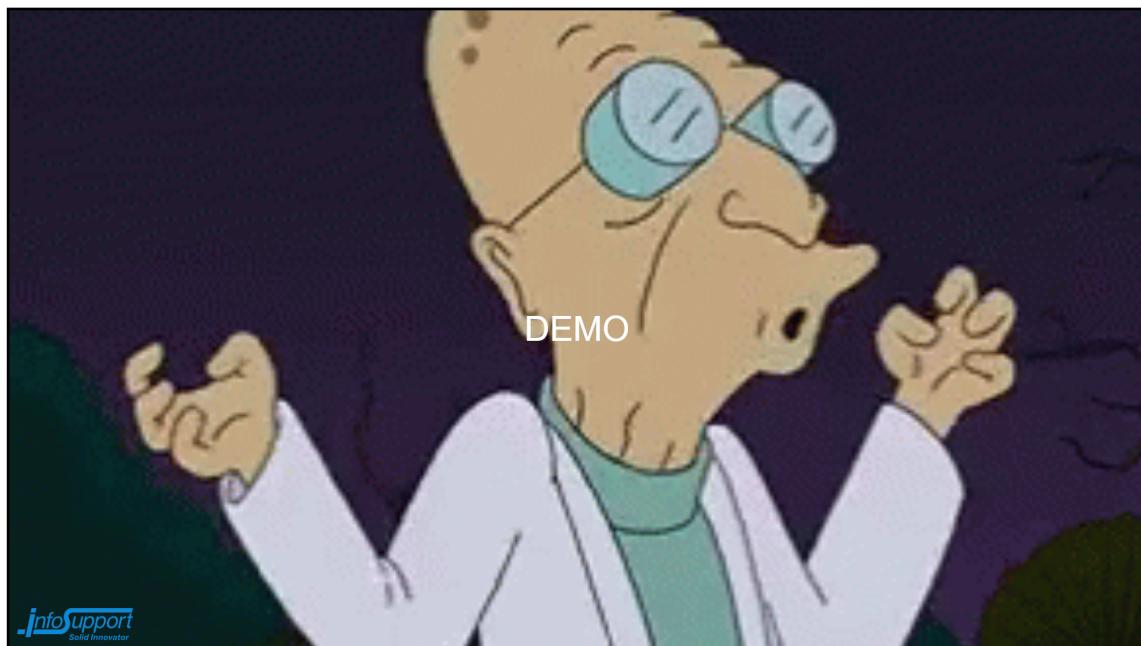
```
@Html.DisplayFor(model => model.PropertyVanHetTypeMyModel)
```

Tag helper

Easy to build :-)

the plan

1. maak een feature branch
2. git push --set-upstream origin <branch>
3. een attribuut schrijven dat knoppen toont/verbergt op basis van user



How to?

1. voeg je namespace toe aan _ViewImports
2. overerven van TagHelper
3. override Process of ProcessAsync

Dependency Injection

<https://medium.com/@yoerivd/how-does-dependency-injection-really-work-b92199694206>



Vakantiewerk?

- weer rekken vullen?
- weer ploegen?
- of .. deze keer iets wat je graag doet?

neem een flyer mee!