



Git - intro

Web Essentials

DE HOGESCHOOL MET HET NETWERK

Hogeschool PXL – Elfde-Liniestraat 24 – B-3500 Hasselt
www.pxl.be - www.pxl.be/facebook



Version Control System

- Systeem om veranderingen in files bij te houden
- Samenwerken aan dezelfde code
- Historiek van files
 - Wie heeft aan wat gewerkt?
 - Uitleg bij elke *commit* is belangrijk!
 - Veranderingen ongedaan maken

Version Control System

- Terminologie:
 - **Repository:**
Plaats waar bestanden én informatie worden bewaard
 - **Master repository:**
Meestal op een externe server
Bevat de huidige (gedeelde) versie van de bestanden in het project
 - **Working copy:**
Lokale kopie van de master repository waar de gebruiker wijzigingen in kan brengen
 - **Working directory:**
Lokale map op systeem van de gebruiker waar hij effectief in werkt en test

Version Control System

- Terminologie:
 - **Commit:**
Veranderingen in de lokale bestanden toevoegen aan een repository
 - **Push:**
Veranderingen in de working copy updaten naar de master repository
 - **Pull:**
Veranderingen in de master repository overbrengen naar de working copy
 - Merge:** Veranderingen van de master worden meestal automatisch verwerkt en samengevoegd in de lokale
 - Conflict:** als de veranderingen niet automatisch opgelost kunnen worden resulteert dit soms in een conflict dat opgelost moet worden.

Version Control System

- Verschillende soorten
 - Non-distributed:
 - Server heeft master repository
 - Alle *commits* naar de server
 - Voorbeelden: SVN, CVS
 - Distributed:
 - Server heeft master repository
 - Ontwikkelaars hebben een lokale kopie
 - Voorbeelden: Mercurial, **Git**

A Distributed Revision Control System

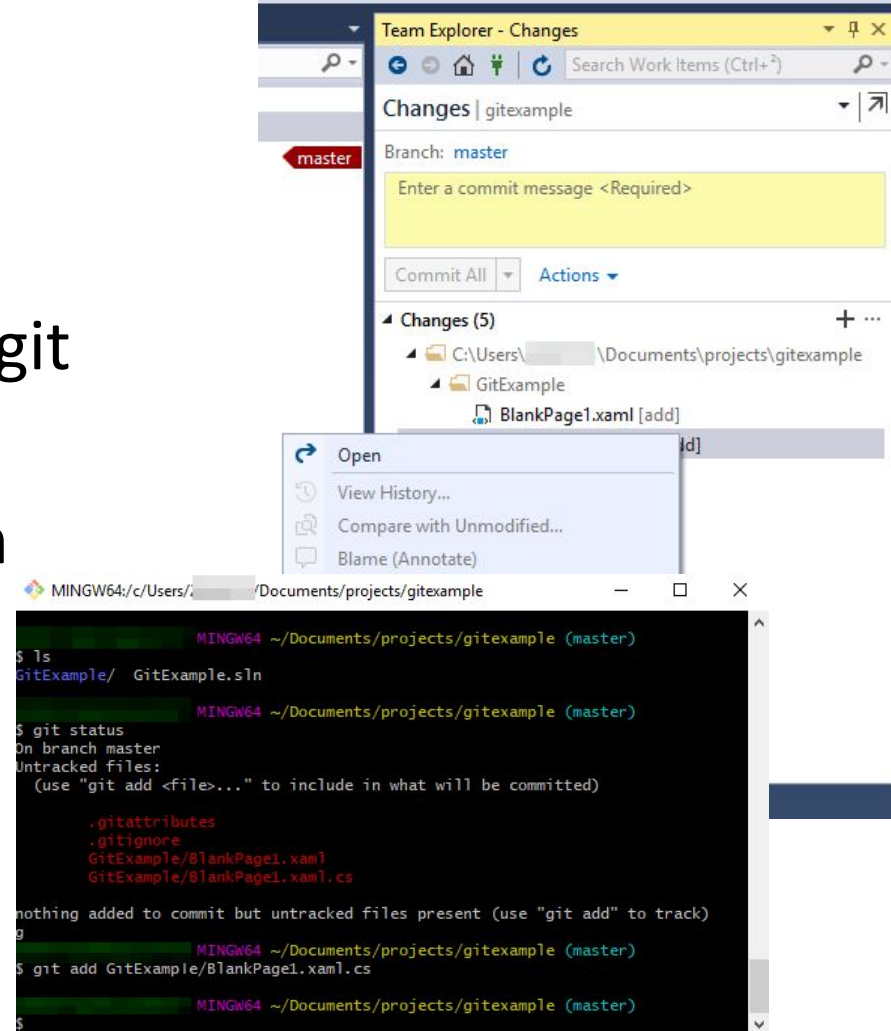
Git

Git

- Open source project gestart door Linus Torvalds
- Heel populair
- Lokale kopie van de master repository
 - Veranderingen lokaal toevoegen
 - Doorsturen naar server kan later

Git operaties

- Interactie met Git
 - Via de command line: git <command>
 - Via GUI of integratie in development tool (bijvoorbeeld VS)

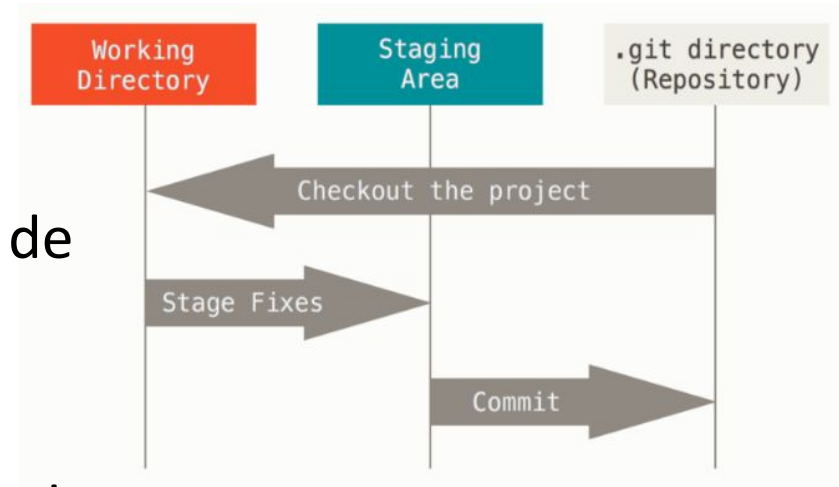


Initialisatie

- Nieuwe repository starten (lokaal):
 - `git init`
- Bestaande repository ophalen:
 - `git clone /path/to/repository`
 - `git clone`
`username@host:/path/to/repository`

Toestand data

- Data bevindt zich in 3 toestanden:
 - Modified:
 - De data is aangepast in de working directory
 - Staged:
 - De data staat klaar voor de volgende commit
 - Committed:
 - Aanpassingen zijn doorgegeven naar de repository



Status

- Toestand van files/data bekijken:
 - git status

```
$ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    new file:   GitExample/BlankPage1.xaml.cs

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

    modified:   GitExample/BlankPage1.xaml.cs
    modified:   GitExample/MainPage.xaml

Untracked files:
  (use "git add <file>..." to include in what will be committed)

    .gitattributes
    .gitignore
    GitExample/BlankPage1.xaml
```

The diagram illustrates the output of the `git status` command. It highlights three categories of file changes:

- Staged:** A new file, `GitExample/BlankPage1.xaml.cs`, is ready to be committed.
- Modified:** Two files, `GitExample/BlankPage1.xaml.cs` and `GitExample/MainPage.xaml`, have been modified but are not yet staged for commit.
- (untracked):** Files `.gitattributes`, `.gitignore`, and `GitExample/BlankPage1.xaml` are present in the working directory but have not been tracked by Git.

Stage

- File/data klaar zetten voor een commit:
 - `git add <filename>`
- Controleerbaar via `git status`
- Kunnen ook meerdere files tegelijk zijn
- Opnieuw iets gewijzigd? -> opnieuw add!

Commit

- Commando:
 - `git commit -m "Boodschap die in de log getoond wordt"`
- ALTIJD een goede message voorzien bij een commit
- Na een commit zitten de veranderingen in de repository (lokaal)
- Controleerbaar via:
 - `git log`

```
$ git log
commit 175c6f10933856ebbd645ff3ddf9d1256c0a4753 (HEAD -> master)
Author: 
Date: Thu Jun 7 12:06:05 2018 +0200

    Initial commit
```

SHA-1 Hash

- Elke commit is een toevoeging van veranderingen
- Een change is een reeks bytes
- Elke reeks bytes kan omgezet worden in SHA-1 hash
- Deze hash is uniek en is de 'naam' van de commit en kan bijvoorbeeld gebruikt worden om terug te keren naar een vorige versie
- Bijvoorbeeld:

```
$ git log
commit 175c6f10933856ebbd645ff3ddf9d1256c0a4753 (HEAD -> master)
Author: 
Date: Thu Jun 7 12:06:05 2018 +0200

    Initial commit
```

Push

- Commando:
 - `git push origin`
- Brengt de veranderingen van de lokale repository over naar de remote repository

Pull

- Commando:
 - `git pull`
- Ophalen van veranderingen op de master repository naar de lokale

Installatie & tutorials

Installatie:

<https://git-scm.com/downloads>

Informatie:

<http://rogerdudler.github.io/git-guide/>

Oefenen:

<https://try.github.io/>

<https://www.codecademy.com/courses/learn-git>

t

PE - week 3

- Maak GitHub account <https://github.com/>
 - voornaamnaamPXL
 - PXL email adres
- Klik op de link op BlackBoard
 - Join classroom
- Volg de instructies in je persoonlijke repository!