

# Tutorial GIT

CHEAT SHEET

VALENTIN.GREGOIRE@GMAIL.COM

## **1    *Inhoudstafel***

1	Inhoudstafel.....	1
2	Algemene informatie.....	2
3	Stappenplan.....	2
4	Commando's.....	2
4.1	Algemeen.....	2
4.2	Getting started.....	2
4.3	Toevoegen/wijzigen/verwijderen.....	2
4.4	Ignore.....	3
4.5	Branching.....	3
4.6	Commit/checkout/push/pull.....	3

## 2 *Algemene informatie*

- <https://git-scm.com/documentation>
- [https://www.youtube.com/watch?v=SWYqp7iY\\_Tc&t=8s](https://www.youtube.com/watch?v=SWYqp7iY_Tc&t=8s)
- <https://www.youtube.com/watch?v=HVsySz-h9r4>

## 3 *Stappenplan*

1. Installeer GIT  
<https://git-scm.com>
2. Gebruik GIT

## 4 *Commando's*

### 4.1 Algemeen

```
git <command> --help
```

Manual van git command.

### 4.2 Getting started

```
git init
```

Initialiseert de huidige folder tot een GIT-project. Merk op dat de (verborgen) map .git toegevoegd wordt.

```
git config --global user.name '<name>'  
git config --global user.email '<e_mail_address>'
```

Voeg naam en e-mailadres toe aan project.

```
git config --list
```

Lijst alle configs van huidige GIT.

### 4.3 Toevoegen/wijzigen/verwijderen

```
git add <file_or_folder_name>
```

Voegt een file of map toe aan de git repository. Na elke change moet opnieuw geadd worden!!

```
git status
```

Toont welke bestanden klaar staan om gecommitt te worden.

```
git rm (--cached) <file_or_folder_name>
git reset (<file_or_folder_name>)
```

Verwijdert files/folders uit de staging.

```
git diff
```

Toont alle differences (wijzigingen) die gemaakt zijn sinds laatste commit.

## 4.4 Ignore

```
touch .gitignore
```

Maakt een ignore file aan. Plaats deze in de root van de repository. Iedere lijn kan een patroon bevatten van te ignore files/folders.

## 4.5 Branching

```
git branch (-a)
```

Toont alle branches voor huidige repository. Flag -a zorgt ervoor dat ook de remote repository opgelijst wordt.

```
git branch --merged
```

Toont alle gemerge branches.

```
git branch <branch_name>
```

Creëert branch met als naam branch\_name.

```
git branch -d <branch_name>
```

Deletet branch **lokaal** met als naam branch\_name.

## 4.6 Commit/checkout/push/pull

Alvorens we kunnen pushen naar een remote repository, moeten we een publieke repository aanmaken. Voorbeelden van gratis repository leveranciers:

- <https://github.com/>
- <https://bitbucket.org/>
- ...

```
git commit -m '<message>'
```

Commit code naar de lokale branch met als boodschap message.

```
git push origin --delete <branch_name>
```

Deletet branch met naam `branch_name` op remote repository.

```
git checkout <branch_name>
```

Checkt de branch uit en zorgt ervoor dat er gewerkt wordt op de branch.

```
git remote -v
```

Toont alle remote repositories voor huidige repository.

```
git remote add origin  
https://github.com/<account_name>/<repository_name>.git
```

Voegt de lokale repository toe aan de remote repository. Soort synchronisatie. “origin” is de naam van de remote repository, “master” is de naam van de branch.

```
git remote
```

Toont alle remote repositories voor de huidige repository.

```
git pull <remote_repository>
```

Pullt alle changes van de remote repository.

```
git clone <url_repository_to_clone> <path_to_clone_to>
```

Kloont een repository naar een willekeurig pad. Om in het huidige pad te klonen, gebruik als destination “.”.