



Marp

# Marp

Markdown to slides test voor  
GitItDone

<https://marp.app/>

# How to write slides

Split pages by horizontal ruler ( `---` ). It's very simple! 😊

```
# Slide 1
```

```
foobar
```

```
---
```

```
# Slide 2
```

```
foobar
```

# Dit is een kersverse slide!

Je kan dus eenvoudig een slide toevoegen hé! Zeer handig wel denk ik dan

```
Dit is een markdown kadertje voor geen specifieke reden (noch onspecifieke reden...)
```

Hier is nog wat tekst onderaan!

# Wat is git nu precies?

- Version control system!!
- Sprekend voorbeeldje toevoegen met visuele weergave

# Een beetje geschiedenis...

- The man, the myth, the legend: Linus Torvalds...

# Git installeren

- Stappen per OS kort eens toelichten (niet zo moeilijk)

# Git configureren

- Email en naam goedzetten
- verschil tussen globaal en lokaal configureren (context) - zeker voor geavanceerde gebruikers interessant

# Git configureren - SSH sleutels

- Waarvoor dienen SSH-sleutels? (kort)
- Hoe genereer ik zo'n sleutelpaar?
- Public key toevoegen aan GitHub (demo?)
- Extra: je kan verschillende keypairs gebruiken voor verschillende accounts/repo's! (voorbeeld Emilia)



# Git workflow

- Overzicht van local flow en met remote erbij (image toevoegen)
- Veranderingen bijhouden en zo doen aan version control!
- Met remote: zelfde concept, maar collaboratief!
- Meerdere slides hiervan maken hoor!

# Git - wat is een commit nu precies?

- Uitleg dat alle veranderingen worden bijgehouden in een 'commit'
- Elke commit krijgt dan ook een commit hash (lange en korte eens uitleggen)
- In een commit kunnen meerdere veranderingen tegelijk zitten -> we proberen ze allemaal atomair te houden!
- Commit messages - BELANGRIJK!!!
- Wat is HEAD en HEAD(~1)

# Git - wat is een .gitignore bestand?

- Soms wil je sommige zaken niet bijhouden in jouw version control (denk aan wachtwoorden, sensitieve data,...)
- .gitignore definieert wat je niet wilt tracken (voorbeeld met demo!)

# Git - command line versus GUI

- Git kan je zowel CLI als GUI gebruiken
- CLI: vrij snel en efficient
- GUI: handig voor een beter overzicht soms, zit in veel IDE's ingebakken (voorbeeld GitKraken)

# Git - command line en basiscommando's

- git clone
- git fetch
- git pull
- git add
- git commit
- git push (remote nodig!)

# Git - meer basiscommando's

- git status (BELANGRIJK)
- git log

# Git - branches

- Wat zijn branches?
- Gerelateerde commando's

# Git - merge conflicten

- Wat zijn merge conflicten?
- Wanneer ontstaan deze?
- Hoe kan je deze oplossen (BELANGRIJK - demo!!)



# Git - samenwerken

- Nood aan een remote repository (niet hetzelfde als lokaal)
- GitHub is een voorbeeld van zo'n provider/platform
- Verschill tussen Git en GitHub kort toelichten

# GitHub specifiek

- Remote repositories, platform om samen te werken
- Heel wat specifieke mogelijkheden (Actions, Pages,...)
- Pull requests!!

# Git - advanced topics

## TO DO

- git rebase: in detail eens overgaan
- Meer advanced commando's?
- GitKraken of andere GUI tool demonsteren?
- Website hosten met GitHub pages -> kan gratis voor statische sites (zie huidige slides)
- DEMO's van vanalles moeten nog toegevoegd worden! (repo's, specifieke flows, use cases,...)

# Handige/interessante links

- Git(Hub) guide van HOGENT - <https://hogenttin.github.io/git-hogent-gids/>
- Leren branchen - <https://learngitbranching.js.org/>
- Wat als het allemaal in de soep loopt? - <https://ohshitgit.com/>