



Universiteit Antwerpen
| Faculteit Wetenschappen

Introductie tot Git(Hub)

Hitchhikers Guide to CS

In case of fire



- 1. git commit
- 2. git push
- 3. leave building

Waarom GitHub?

Redenen om GitHub te gebruiken

Backups en version control

- `rm -rf .`
- "Ik heb deze functie compleet herschreven en nu werkt het niet meer. Hoe zag mijn vorige versie er ook al weer uit toen het wel werkte?"

Samenwerken

- Populaire filessharing technieken in eerste bachelor
 - Google Drive
 - Wetransfer
 - Email
 - iCloud/OneDrive
- "Ik heb 4 kopies als versies van deze file, welke moet ik gebruiken?"

Git vs GitHub

Git

- Tool voor version control
- Kan gebruikt worden zonder GitHub

GitHub

- Platform om samen te werken en backups in cloud
- Heeft git nodig voor version control



Git

Software

Version control

Maintained by Linux

Open-Source

No user management

Locally installed

Minimal external tool
configuration

Little to no competition



GitHub

Service

Git repository hosting

Maintained by Microsoft

Free or paid membership

Built-in user management

Hosted on the web

Active marketplace for
tool integration

High competition

Getting started

Installeer Git lokaal

- **Ubuntu:** `sudo apt install git-all`
- **MacOS:** installeer xcode commandline tools
- **Windows:** <https://git-scm.com/downloads/win>
- **Test of het werkt**
 - **Open Terminal of CMD**
 - Alternatief op Windows: rechtermuisknop in folder > git bash here
 - **Typ "git help"**

Account aanmaken op Github

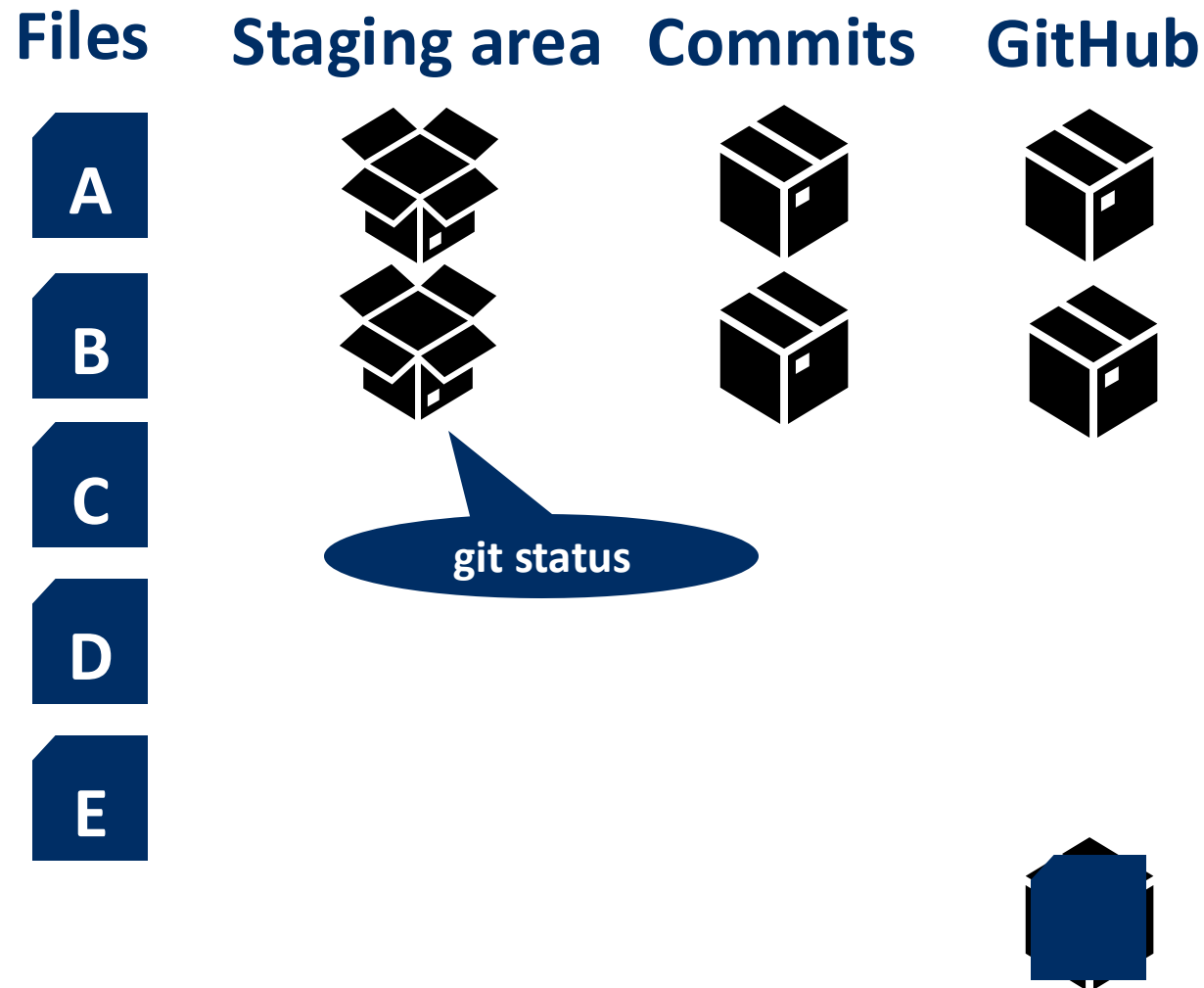
- Surf naar github.com
- Maak een account aan met je EIGEN privé email
- Voeg UA mail toe als extra mail voor gratis PRO licentie
 - Ga naar Settings
 - Emails
 - Voeg `voornaam.achternaam@student.uantwerpen.be` email toe

Maak een Git repository aan

Link GitHub repository aan lokaal project

Basis commando's

Visuele voorstelling



```
git add A
git add C
git commit -m "A en C"
git add E
git add B
git add D
git commit -m "E, B en D"
git push
git pull
```

Overzicht

- **git add <filename of directory>**
- **git status**
- **git commit –m “<omschrijving wijzigingen, zo concreet mogelijk>”**
- **git push**
- **git pull**

.gitignore

- Template om bepaalde files/folder te negeren
- Handige voorbeelden / files om te gebruiken
 - <https://github.com/github/gitignore>

Don't's

- Niet werkende code pushen
- Code pushen die afhankelijk is van absolute paden
 - (e.g., /home/<naam>/school/project/lib/file1.cpp -> ./lib/file1.cpp)
- Persoonlijke folders/files pushen -> .gitignore
 - .env (!!!)
 - .pycache

Do's

- **Pull voor je een push doet!**
 - Vermijd merge conflicts

Advanced commando's

git merge

git branch

