

Git (tool) si GitHub (cloud)

1. BASICS

- Ce este GitHub
- Este primul loc unde se uita un angajator după ce vede CV-ul
- Este interesant de văzut ceea ce știe aplicantul practic: ce și cum scrie un cod/proiecte/site-uri
- Este un loc public, un cloud, unde ne putem afișa munca
- La munca acesta reprezintă codul comun la care lucrează mai mulți colegi
- Este acel loc din hub unde vei scrie proiectul și unde angajatorul poate să vadă ceea ce faci
- Repo = repository - loc în GitHub unde ne putem pune proiectul
- Comanda clone = fișierul îl poate lua orice coleg/angajator, poate să-l verifice, îmbunătățească etc
- Push/pull/add/commit/clone - add împachetează fișierele, commit le și salvează
- Ramura principala se numește main
- Remote repo - Git pull/Clone - Work Direct - Git add (sunt salvate temp pt a fi pus în repo) - Stage Area - Git commit - Local Repo - Git Push - Remote Repo - Git pull again pentru actualizare fișiere în local repo (daca lucrează mai multi)

2. SETUP - instalare Git și GitHub;

- show hidden files MacOS cmd+option+. (punct)
- Open terminal: git -- version, git init
- Git config - - global user.email "email"
- Git config - - global unser.name "username"

3. CUM SA INCARCI UN PROIECT IN GITHUB

- Terminal => git init => face un .Git în folderul cu proiectul
- Terminal => git status => arata fișierele din folderul respectiv
- Terminal => git add . =>încărcate fișierele în git
- Terminal => git commit -m "nume proiect sat .."
- Git branch - arata ca suntem pe ramura buna
- Git push
- Git status
- git remote add origin https://github.com/Dragosne/First_repo.git
- Il iei din noul repository

1. CUM MENTINEM PROIECTUL LA ZI

- Git commit - m "nume proiet"
- Git push/
- status
- Git clone încărcător un repo remote pe local

2. CUM IGNORAM FISIERELE NEDORITE

- Fișiere pe care dorim sa le ascundem în GitHub

- Terminal => touch .gitignore
- Open .gitignore in python
- Search chrome .gitignore
- git rm -r --cached . (Daca adaugi noi fisiere în .gitignore ulterior trebuie data comanda asta în terminal ca se le ignore)

Ordinea este pentru proiect repository nou: git init/git add ./git commit -m "name"/git remote add origin https://github.com/Dragosne/First_repo.git/git push -u origin main

Git rm "nume fisier" șterge un fișier dintr-o ramura de ex

- Touch readme.md (\ pentru rând nou sau lăsată linie goală)

6. CUM LUCRAM CU RAMURI

- Git checkout -b ramura01
- Git checkout main ("nume branch")
- Git branch
- Git fetch aduce la zi info din remote
- Git merge aducem codul dintr-o ramura în main

7. CUM REZOLVAM CONFLICTELE

- Git diff arata diferențele într-un
- fișier editat local și remote
- Touch nume_fisier Cream un fisier nou

8. CUM TE PREZINȚI UNUI ANGAJATOR?

- linkedIn + gitHub
- Poza profil GH
- Readme se deschide prima data (ABOUT ME + Certificates)
- coursera udemy
- About me/Cv/Skills/Tools/Courses
- <https://github.com/esbazhenova/portfolio>

Utile:

- **"Testing dot com"** R.Savin 2017
- **"A Practitioner's Guide to Software Test Design"**
- Curs python free pe LinkedIn
- Site? Blog? Carti citite. Grupuri de FB
- <https://github.com/search?q=%2Fjulai215&type=repositories>
- Cv/linkedin/github/website

Nicolae Pavel linkedin

<https://www.markdownguide.org/cheat-sheet/>