



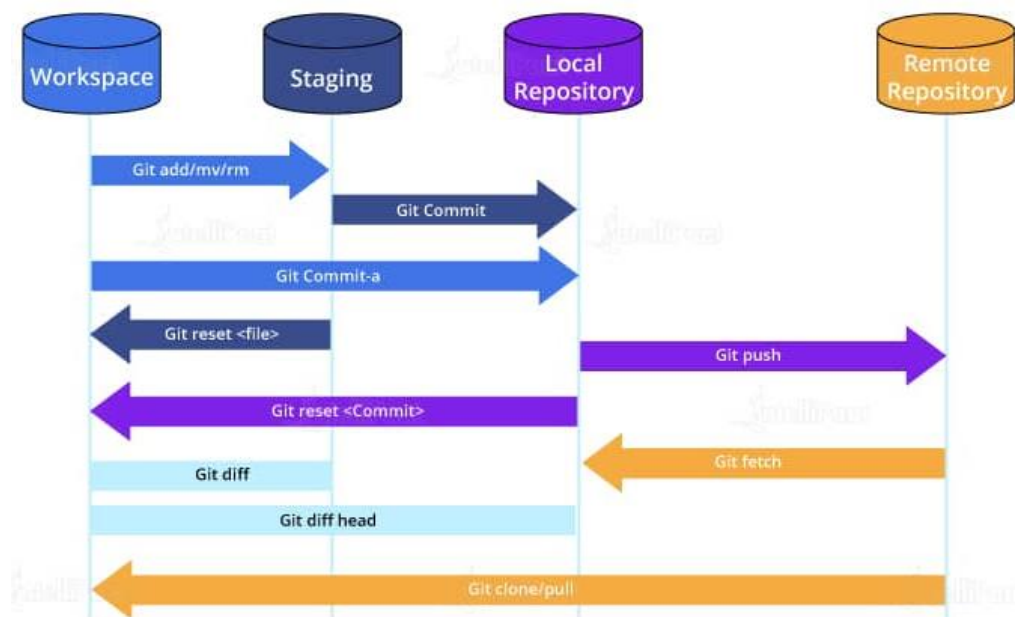
Operativni Sistemi

II Semestar – 2022/23 – Vježbe

Git Osnovne Naredbe

Najpoznatiji sistem za verzioniranje koda (eng. VSC-System Control Version) → GIT

- Struktura Git-a



Postoje 4 stepena Git-a (tokom kontrole verzija projekta, datoteka):

- **WORKING DIRECTORY:** Folder na lokalnom računaru koji pohranjuje projektne datoteke. Instalirani Git je „svjestan“ tome folderu, može upravljati s njime, a i ne mora.
- **STAGING AREA:** (također se zove Git Index ili pre-commit holding area) Promjene u verzijama koje će se predati će se prvo zadržati na ovom stanju.
- **COMMIT:** Ovaj stepen predstavlja „Git repository“ ili historiju promjena. Sve finalizirane promjene u svakoj verziji projekta su zauvijek pohranjene ovdje.
- **REMOTE REPOSITORY:** Cloud, GitHub (Kreira se preko <https://github.com/>)

BRANCH – „timeline“ pohranjenih promjena. Glavni branch u svakom GitHub repositoryju je **main** (naziv se može promijeniti).

Postoje dva načina korištenja Git-a:

- 1) Git Desktop s grafičkim korisničkim okruženjem
- 2) Git Bash kao komandna linija (slično cmd-u) za Git naredbe (pruža potpune mogućnosti)

Osnovne komande:

- **\$ git version** za provjeru verzije Git-a
- **\$ git exit** za izlaz iz komandnog okruženja
- **\$ pwd** za provjeru u kojem direktoriju se nalazimo na računaru
- **\$ mkdir naziv-projekta** za kreiranje novog projekta
- **\$ cd** za navigaciju na posebni folder
- **\$ git config --global user.name „vaš-username“** za postavljanje user name za pristup
- **\$ git config --global user.email „vaša-mail-adresa-na-github“**
- **\$ git config --global --list** za izlistavanje username i email adrese
- **\$ git clone url-github-repo** za kloniranje/kopiranje remote repositoryja na lokalni PC
- **\$ ls** za izlistavanje repositoryja
- **\$ git status** za provjeru na kojem branch-u se nalazimo, također javlja da li postoje nove promjene napravljene u *Working directory* koje trebaju preći *Staging* i *Commit* status
- **(main) naziv-projekta \$ echo „Neki tekst“ >> start.txt** za kreiranje nove datoteke u lokalni repo
- **\$ cat proba.txt** za prikaz sadržaja datoteke proba.txt
- **\$ git add proba.txt** za prebacivanje datoteke proba.txt iz *Working dir.* u *Staging area*
- **\$ git commit -m „naziv-datoteke“** za prebacivanje datoteke koja se nalazi u *Staging area* u *Committed status* (nakon čega naredba **git status** javlja da je *Working dir.* clean)
- **\$ git push origin main** za slanje ažuriranih promjena (*committed changes*) na remote Github repository
- **\$ git init -b main naziv-brancha** za promjenu/specifikaciju zadanog branch-a za lokalni repo

- **\$ git config --global init.defaultBranch novi-naziv** za konfiguraciju/preimenovanje novog branch-a
- **\$ git init novi-primjer** za kreiranje novog lokalnog repositorija novi-primjer
- **\$ git branch novi-branch** za kreiranje novog branch-a u trenutnom direktoriju
- **\$ git checkout novi-branch** za prelazak/navigaciju na novi branch
- **\$ git checkout -b novi-branch** za simultano kreiranje i navigiranje na novokreirani branch
- **\$ git push -u origin naziv-branch** za slanje lokalnog branch-a na remote repository
- **\$ git push -all -u** za slanje svih branches

Napomena: Možemo imati više repositorija na remote/GitHub lokaciji. **Main** branch ukazuje na jednog.

- **\$ git remote -v** se koristi za provjeru na koji repo pokazuje main branch
- **\$ git remote set-url origin URL-adresa-repo** za navigaciju main-a na drugi remote repository (čiji URL proslijedujemo u komandi)

Kontakt:

Narcisa.hadzajlic@size.ba (A Grupa)

Adin.jahic2019@size.ba (B Grupa)