

Git

Was ist version Control?

- Ein System um Veränderungen aufzubewahren.
- Etwas was uns erlaubt Zusammen mit anderen zu arbeiten.
- Ein Tool mit dem wir sehen können wer wann welche Veränderungen vorgenommen hat.
- Version Control erlaubt uns **Veränderungen rückgängig zu machen und zurück zu einem älteren Zustand des Projekts zu gelangen.**

Was ist git?

Ein solches Version Control System (VCS)

- Jeder Programmierer innerhalb eines projektes kann auf alle alten Zustände des Projektes auf dem eigenen Computer zugreifen.
- Programmierer können ohne Internetverbindung veränderungen durchführen.
- (Allerdings geht dann *pushen* und *pullen* von einem remote Server nicht.

Kurze git Geschichte

- Gibt es seit 2005
- Erfunden durch Linus Torvald, dem Linux Erfinder.
- Es gab und gibt noch andere VCS . z.B. Perforce und Subversion aber Git ist am weitesten verbreitet..

Wie funktioniert git?

Snapshots (Commit)

- Damit verfolgt git alle alten Zustände deines Projekts.
- Protokolliert wie deine Dateien wann ausgesehen haben.
- Du kannst selber entscheiden wann du einen Snapshot machst, und welche Dateien da drin vorkommen sollen.
- Du kannst jederzeit zurück zu einem alten Snapshot gehen.
- Aber die späteren Snapshots bleiben trotzdem vorhanden.

Commit

- Die Art mit der wir einen Snapshot machen.
- Kann ein Nomen oder ein Verb sein.
- Ich habe gerade code committed.
- Ich habe gerade einen neuen Commit erstellt.
- Ein Projekt beinhaltet eine Sammlung an Commits.

Repository

- Oft auch einfach kurz: Repo
- Eine Sammlung an Dateien und deren Bearbeitungsgeschichte.
- Hier sind all deine commits drinnen.

Repository

- Wenn man Commits die du auf deinem Rechner nicht hast von dem Remote Server runterlädst, nennt man das pulling.
- Wenn du lokale Commits zu dem Remote Server hinzufügst, nennt man das pushing

Branches

- Alle Commits in Git leben in einer Branch.
- Es kann viele solche Branches geben.
- Der Hauptbranch in einem projekt heißt: main.

Wie sieht so eine Branch aus?

- Commits die in einer Kette verlinkt sind
- <http://git-school.github.io/visualizing-git/>
 - HEAD ist die Referenz auf den letzten commit
 - master ist der primäre Branch in deinem Repository.

Vokabular Time-Out

Repository

- Arbeitsverzeichnis, Euer Projektordner
- GIT beaufsichtigt die Änderungen im Repository

Commit

- Git speichert die Änderungen in der History nur ab, wenn wir Git proaktiv dazu auffordern
- = "Git's way of saving"
- In einem Texteditor sagen wir ctrl+s und dieser speichert die Datei ab
- In git passiert dies nur, wenn wir einen commit machen

Stage

- STAGING = wir bereiten etwas vor,
- Wir bereitein den commit vor

Der Git Workflow

Git States

**Working
Directory**

**Modified
Files**

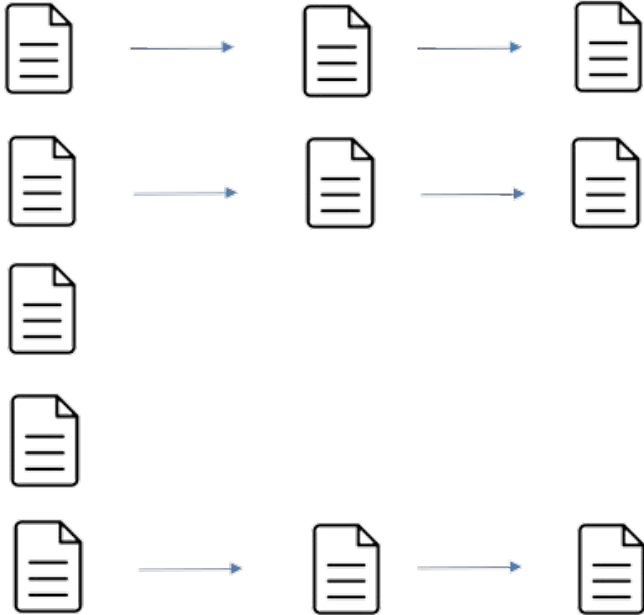


Git States

**Working
Directory**

**Modified
Files**

**Staged
Files**



Git States

**Working
Directory**

**Modified
Files**

**Staged
Files**

**Committed
Files**



Git States



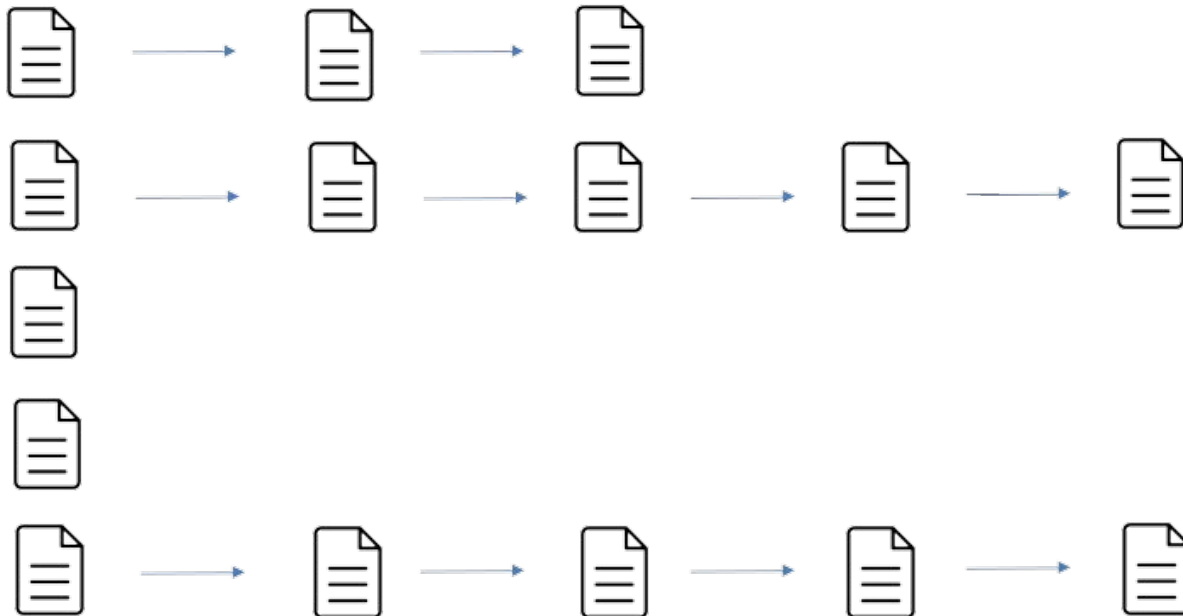
**Working
Directory**

**Modified
Files**

**Staged
Files**

**Committed
Files**

**Remote
Repository**



Git States

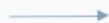
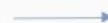
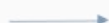
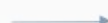
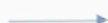
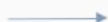
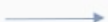
LOCAL

Working
Directory

Modified
Files

Staged
Files

Committed
Files



REMOTE



Remote
Repository

Merging

<http://git-school.github.io/visualizing-git/>