

Praktikum „SEP: Java-Programmierung“ SS 2019

Git und Gitlab

Karlheinz Friedberger und Thomas Bunk

All You Need

- ▶ Simple words for a GitLab Newbie:
<https://docs.gitlab.com/ee/gitlab-basics/>
- ▶ Udacity Course: Version Control with Git
<https://eu.udacity.com/course/version-control-with-git--ud123>
- ▶ Buch: Scott Chacon, Ben Straub. Pro Git
<https://git-scm.com/book/en/v2>

1. Grundlagen Version-Control-Systems (VCS)
2. Git: Konzept + Verwendung
3. Projektverwaltung mit Gitlab

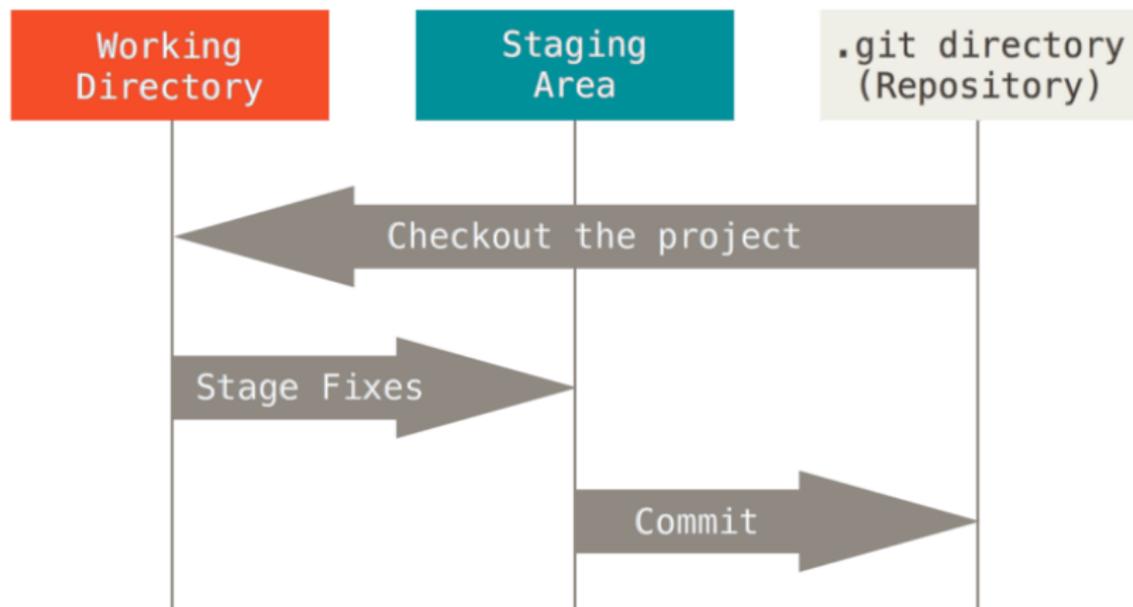
- ▶ Versionierung
 - ▶ Revisionen
 - ▶ Branches
 - ▶ Merges
- ▶ Kollaboration
- ▶ Zentrale oder dezentrale Lösungen möglich

- ▶ Zentral:
 - ▶ Versionierung zentral auf Server (Verbindung notwendig)
 - ▶ Keine lokale Versionierung
 - ▶ Beispiele: Mercurial, CVS, SVN

- ▶ Dezentral:
 - ▶ Jeder Nutzer eigenes, lokales Repository (werden untereinander synchronisiert)
 - ▶ Lokale Versionierung möglich
 - ▶ Beispiel: Git

- ▶ Synchronisierung zwischen Repos mit `pull` und `push`
 - ▶ Initiales „Herunterladen“ eines Repo mit `git clone`
- ▶ Besonderheit: Staging Area zum Vormerken von Änderungen
- ▶ Download: <https://git-scm.com/downloads>

Git: Staging Area



from Pro Git book v2, licensed under CC-BY-NC-SA 3.0

- ▶ Befehl: 'git'
- ▶ Wichtigste Befehle:
 - ▶ 'git clone'
 - ▶ 'git pull (-rebase)'
 - ▶ 'git diff'
 - ▶ 'git add'
 - ▶ 'git commit'
 - ▶ 'git push'
 - ▶ 'git reset'

Für Interessierte:

<https://www.atlassian.com/git/tutorials/merging-vs-rebasing>

- ▶ Übersicht: <https://git-scm.com/downloads/guis/>
- ▶ z.B. Gitkraken, Sourcetree
- ▶ IDE-Integration: Bei Eclipse + IntelliJ dabei

- Repository/Repo** Datenbank mit Dateien des VCS, all seinen Versionen und Branches
- Working Copy** lokale Kopie der Dateien eines Repo (kann sich von neuester Version im Repo unterscheiden)
- Revision** Committeter Zustand einer Datei (CVS), eines Branch (Git) oder des ganzen Repo (SVN)
- Change Set** Veränderungen an Dateien im VCS
- Commit** Schreiben eines Change Set von der Working Copy ins Repo
- Update** Updaten der Working Copy mit Änderungen an Dateien im Repo seit dem letzten Update

Branch	Menge von Datei(versionen) im VCS die sich unabhängig von anderen Dateien oder Datei-versionen entwickeln
Master	Haupt-Branch der Entwicklung
Fork	Abgespaltener Klon eines Repo, der unabhängig verwaltet und weiterentwickelt wird
Tag	Menschenlesbarer Link zu bestimmter Revision(z.B. festlegen von Versionsnummern des Programms)
Merge	Zusammenführen der Änderungen in mehreren Branches
Switch	Wechsel der Working Copy zwischen Branches
Conflict	Zwei parallele Änderungen an einer Datei, die nicht eindeutig zusammengeführt werden können

Live-Demo