

GIT/GITLAB: Fragen und Übungen

Firaz Ilhan fi007*,

Maximilian Dolbaum md127*

Jens Schlegel js414*

1. Was genau sind die Gründe um Gitlab zu verwenden?

Vorteile von Git:

- Gruppenarbeit vereinfachen
- dezentralisiertes Arbeiten
- merge konflikte lösen
- Zugriffsrechte
- Variantenbildung (Branches)
- Man braucht kein MainV1.java MainV2.java usw.
- Rollback-Möglichkeit
- Dem Repository bereits bekannte Dateizustände werden nicht erneut gespeichert, sondern werden mit Hashcodes referenziert -> bessere Performance
- Gitlab und Github bieten eine Open source platform

2. Welche Daten gehören nicht ins Repo?

Ins Repo	Nicht ins Repo
.java	Zeichnungen
.xml	Passwörter für Cloud-Services
.json	Passwörter für lokale Services (Self-hosted)
Bilddateien für Spiel	Notizen
Musikdateien für Spiel	Logfiles
UML	Kapitel eines Buches (Copyright-Verletzung)
Dokumentation	Messdaten vom Profiling
.project Files (Endung je nach Tool)	
Konfigurationsfiles	
Bachelorarbeit	

3. Was soll der Mist mit den Stages (dass add/commit nur lokal wirken)?

- Veränderungen Lokal behandeln ohne Andere direkt zu beeinflussen (Konflikte)
- Änderungen (Commits) können damit logisch strukturiert werden

4. Würden Sie in einer Firma Gitlab selber hosten oder GIT als Service im Netz?

Begründung.

- Privatsphäre wenn selber hosten
- Einfacher für internes Teammanagement
- Der Git-Server muss selber verwaltet werden, was Zeit kostet
- Volle Kontrolle über den Server
- Github bzw. Gitlab können jederzeit das Projekt entfernen (siehe youtube-dl <https://github.com/ytdl-org/youtube-dl>)

5. Verwenden Sie Branches im Projekt oder arbeiten alle Teammitglieder auf dem Master Branch?

Zeigen Sie Vor- und Nachteile der Verfahren

Vorteile	Nachteile
weniger merge Konflikte	Verschiedene Sachen gleichzeitig checken
Unabhängigkeit	Nicht gesamtes Projekt als Ganzes vorliegen
Für Experimentelle Funktionen	
Einzelne Funktionen können unterteilt werden z.B. feature/Addition	
Fehlersuche spezifischer	

6. Wie veröffentlichen Sie Ihre Änderungen auf dem globalen Repository?

- git add/commit/push(merge)?

Wie oft checken Sie Ihre Änderungen im globalen Repo ein?

Was ist besser: Nach jeder Änderung, nach einigen Änderungen,

wenn Sie ein zusammenhängendes Stück fertig haben, wenn Sie eine Änderung machen die viele Kollegen betrifft, einmal am Tag, einmal die Woche. Wieso?

Häufige push, sodass keine Merge Conflicts entstehen und die anderen Programmier auf Basis des neuen Code ihre Arbeit fortsetzen können. Dabei besteht die Gefahr, dass nicht lauffähiger Code die Arbeit der anderen Programmier beeinträchtigt.

7. Eine Version eines Files im Repo sieht so aus:

„There are two versions of GitLab: Community Edition (CE) and Enterprise Edition (EE)“

Ihre Version die Sie hochladen hat an dieser Stelle:

„There are two versions of GitLab: CE and EE.“

Wie löst Gitlab diesen Konflikt?

-merge konflikt check

8. Hausaufgabe für jedes Team: Es muss eine chain-story (Kettengeschichte) erzählt werden bei der

ALLE Teammitglieder jeweils immer einen Satz erfinden und dann einchecken. Ich will von ALLEN

Teammitgliedern Sätze im Repo finden! Danach sagen Sie uns wie es ging. Haben Sie über externe

Kanäle kommuniziert (Slack?). Anderweitig abgesprochen?

siehe Github für die Chainstory

<https://github.com/Firaz-Ilhan/math-trainer>

Einladung wurde an <https://github.com/wkriha> versendet

Kommuniziert haben wir über Discord