

---

Vorname, Name, Matrikelnummer

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

*Die Arbeit mit mehreren Personen wird durch kontrollieren von Änderungen unterstützt. Änderungen werden überwacht, können wenn nötig rückgängig gemacht werden, bereit gestellt werden und Aktualisierungen eingeholt werden. Zudem kann es benutzt werden um Versionierungen bereit zu stellen.  
Im Allgemeinen ist Git ein sehr gutes Tool um die Entwicklung einer Applikation zu dokumentieren und zu unterstützen, sei es durch Analysen, vorgefertigte Templates etc.*

2. Welche Daten gehören (nicht) ins Repo?

- .java Files
- .xml Files
- .JsonFiles
- Bilddateien für Gameprojekte
- Musikdateien für Projekte
- .project Files (Endung je nach Tool)
- UML Modelle
- Zeichnungen
- Notizen
- Dokumentation
- Configurationsfiles
- *Kapitel eines Buches*
- *Eine Bachelorarbeit*
- *Passwörter für Cloud-Services*
- *Passwörter für lokale Services (Self-hosted)*
- logfiles
- Messdaten vom Profiling

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

*Dieser Mechanismus ist zur Absicherung da, damit man nicht ausversehen fehlerhaften Code direkt in das remote repository pusht. So können große Fehler verhindert werden, wenn z.B. bugs einem erst später auffallen gibt es immer noch den Fail-Save, dass es noch nicht online ist.*

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

*Sollte die Firma die nötigen Server zur Verfügung haben, welche auch 24/7 online sind sollte man es darüber machen, da man so besser Zugriffsrechte und Management von Projekten hat. Obwohl GitLab noch nie durch geklauten Code etc. aufgefallen ist, ist es natürlich sicherer es selbst zu hosten. Sollten die Kapazitäten dafür jedoch nicht ausreichen ist der Service online im Netz auf jeden Fall keine schlechte Alternative.*

5. Verwenden Sie Branches im Projekt oder arbeiten alle Teammitglieder auf dem Master Branch? Zeigen Sie Vor- und Nachteile der Verfahren

**Master:**

- +keine Merge Konflikte wenn dann lokal
- + 'pull' -> direkt auf neuestem Stand
- +für 'Anfänger' einfacher zu bedienen
- unübersichtlicher workflow (wer macht was)?

**Branches:**

- +einfache Aufgabenverteilung (GUI-Branch, Logik-Branch,...)
- +workflow gut ersichtlich (z.B. wo Fehler entstanden sind etc.)
- Merge Konflikte können entstehen sollten die Branches miteinander agieren
- bei großen Projekten teils komplex zu managen

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

*Durch den push - Befehl werden alle Änderungen, die mit commit vorher auf dem lokalen Repository definiert wurden auf das globale Repository gepusht. Wie oft dies passieren soll kann für jedes Projekt bzw. Arbeitsgruppe anders sein, man sollte jedoch nicht zu lange mit dem pushen warten, da sonst die Wahrscheinlichkeit auf Konflikte mit Änderungen von Kollegen steigt. Bei uns haben wir es so gehandhabt, dass komplett neue Code-Stücke erst nach fertigstellen gepusht wurden und Änderungen an vorhandenen Code direkt, sofern diese funktionieren.*

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?

*Sollte der obere Stand gleich mit dem unmodified lokalen Repo sein wird die Änderung einfach normal gepusht ohne Probleme zu bereiten.*

*Wenn aber der obere Stand erst nach dem letzten 'pull' vom lokalen Repo kam, wird der 'push' abgewiesen mit dem Hinweis, dass man die remote changes erst mergen soll.*

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?