

.....
Vorname, Name, Matrikelnummer

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

- **Being able to work with other developers on one project.**
- **Easy version controll.**
- **Being able to work seperately from your team members.**

2. Welche Daten gehören (nicht) ins Repo?

- .java Files
- .xml Files
- .Jason Files
- Bilddateien für Gameprojekte
- Musikdateien für Projekte
- .project Files (Endung je nach Tool)
- UML Modelle
- Zeichnungen **Does not belong**
- Notizen **Does not belong**
- Dokumentation
- Configurationsfiles
- Kapitel eines Buches **Does not belong**
- Eine Bachelorarbeit
- Passwörter für Cloud-Services **Does not belong**
- Passwörter für lokale Services (Self-hosted) **Does not belong**
- logfiles
- Messdaten vom Profiling

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

With stages you are able to split up your commits. For example:

You work on a new feature and while working on that you fixed a bug. Instead of committing both changes at once you can git add the feature and commit it separately from the bug fix.

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

You should rather host Gitlab yourself. Using Git as a service on the web results in the company showing off their projects during development. Due to security reasons you do not want that.

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

Pros for using branches:

- Team members can work on a new feature without disrupting the main project.

Cons for using branches:

- Branching can get complex and convoluted.

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?

a) You push your changes to the global repository by using git push.

b) You should always check for changes in the global repo.

c) You should do atomic commits.

→ **Commit often but only if have a block of work done.**

→ **Ensures that you can revert your project to the point you want.**

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 wollen hat an dieser Stelle:

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

Wie löst Gitlab diesen Konflikt?

You talk with your team members about which version you want to keep,

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?