

## Installation GIT

Installieren Sie bitte GIT auf Ihrem Laptop bzw. Computer. Sie finden GIT unter

<https://git-scm.com/>

Diese Anleitung verwendet die Git-Bash (windows). Die Konsolen-Befehle müssten auch in einem Linux-Terminal oder MacOS-Terminal funktionieren. Grafische Werkzeuge aus Ihrer Entwicklungsumgebung bzw. Tortoise müssten dieselben Funktionen über deren jeweilige Oberfläche bereitstellen.

Starten Sie die GitBash wechseln Sie in ein Verzeichnis ihrer Wahl (das \$ kommt von der GitBash), nachfolgend ein Beispiel auf dem D-Laufwerk unter Windows:

```
$ cd /d/sandbox/vvprojekt
```

dort werden Sie Ihr Repository anlegen. Sollten Sie GIT noch nicht konfiguriert haben, bitte konfigurieren Sie Ihren Namen und Ihre Email Adresse

```
$ git config --global user.name "Ihr Name"
$ git config --global user.email "Ihre FH(!) Mailadresse"
```

## Repository Clonen und einrichten

Dann Clonen Sie das Repository in das Verzeichnis in dem Sie sich befinden:

```
$ git clone https://inf-git.fh-rosenheim.de/vv-ss18/IhrName.git .
```

Eine erste Datei ergänzen Sie wie folgt. Das sichert ab, dass Sie einen Master-Branch haben. Eine Readme-Datei für die Inhalte Ihres Repositories ist immer eine gute Idee.

```
$ touch README.md
$ git add README.md
$ git commit -m "add README"
$ git push -u origin master
```

## dev-Branch erzeugen

Sie arbeiten nun mit einem dev-Branch weiter. Dieser wird wie folgt angelegt. Das Kommando legt den Branch „dev“ an und wechselt für Sie in diesen Branch.

```
$ git checkout -b dev
```

In diesem Verzeichnis können Sie nun Ihr Projekt anlegen, beispielsweise mit IntelliJ. Sorgen Sie möglichst auch für eine sinnvolle .gitignore-Datei.

Programmieren Sie nun Ihre Lösung mithilfe Ihrer Entwicklungsumgebung. Und committen Sie diese auf der Bash bzw. in ihrer Entwicklungsumgebung

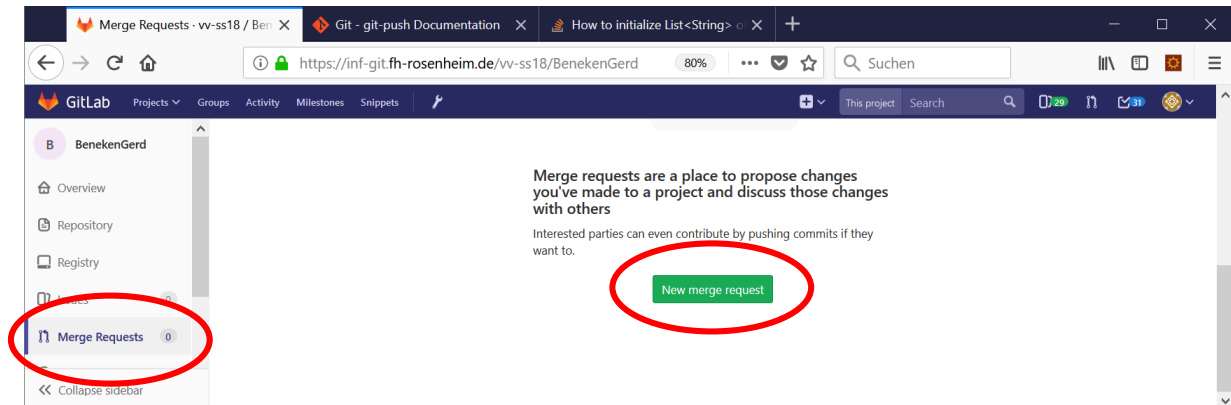
```
$ git add --all
$ git commit -m "Erste Version"
```

Ihre Lösung muss jetzt in das zentrale Repository gepushed werden. Dabei ist es wichtig, dass der Branch mit genannt wird:

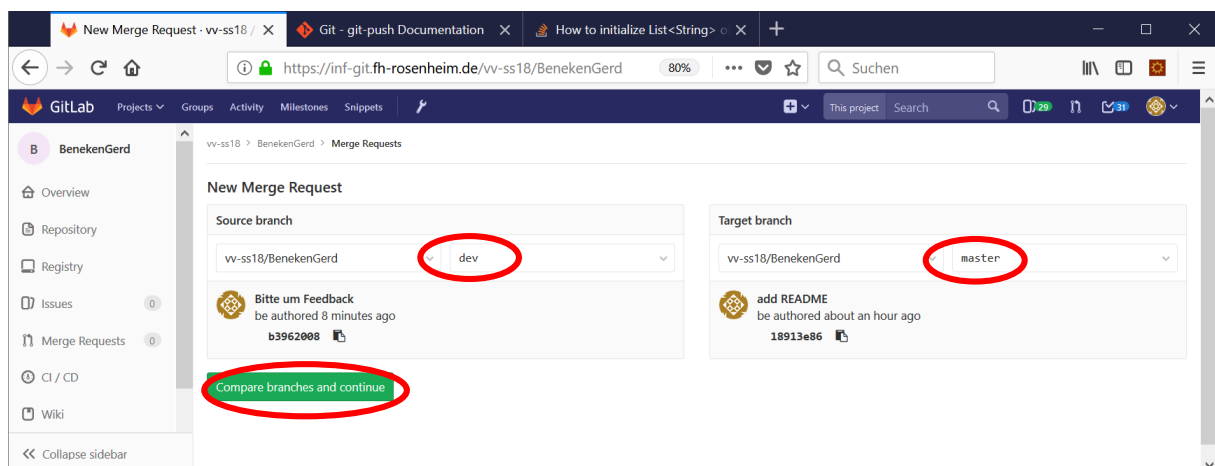
```
$ git push -u origin dev
```

## Ersten Merge-Request in Gitlab erzeugen

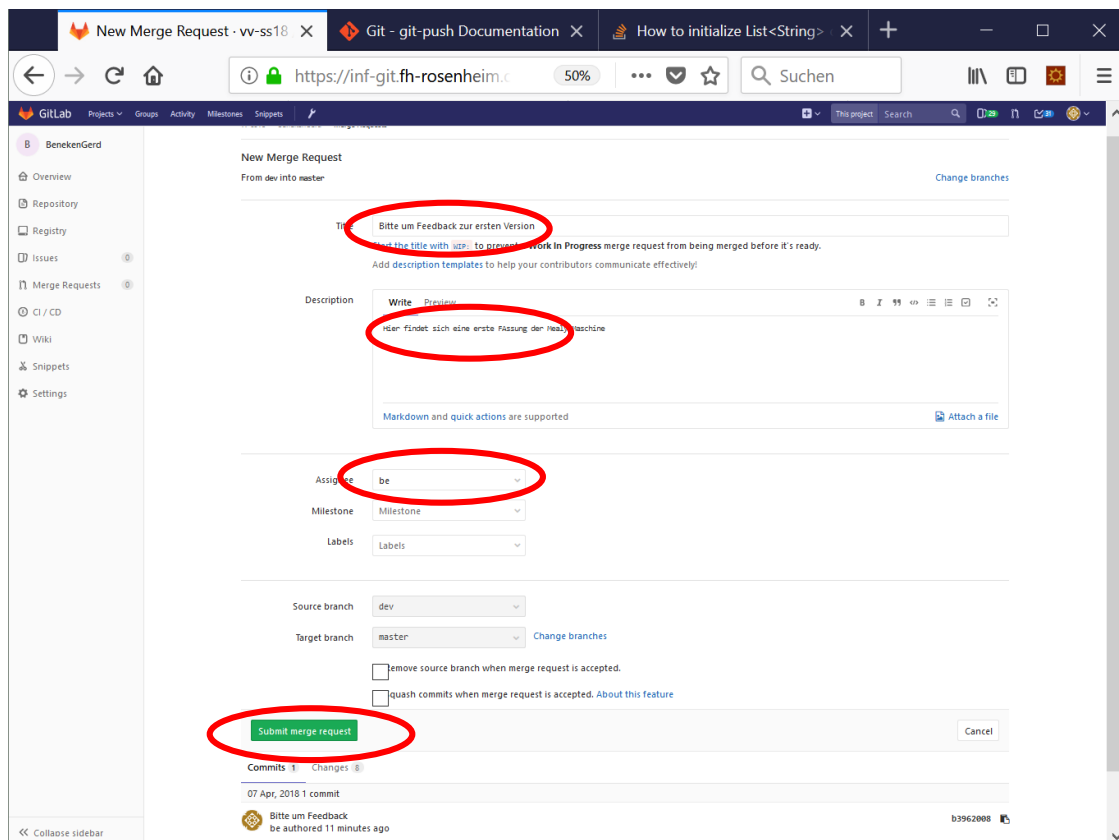
Nun gehts auf Gitlab weiter: Bitte melden Sie sich an und wechseln auf Ihr Projekt. Dann wählen Sie „Merge Requests“:



Und erzeugen einen neuen Merge Request. Dort wählen Sie bitte den gerade erzeugten Commit aus. Wählen Sie als Source branch unseren gerade angelegten „dev“-Branch. Als Target branch bitte „master“



Im Formular zum Merge-Request weisen Sie bitte ihrem jeweiligen Tutor diesen Request zu. Eventuell können Sie noch Fragen oder Hinweise im Kommentar ergänzen.



## Beispiel für Feedback

Die Tutoren geben Ihnen dann Hinweise zu Ihrem Code, der sich in den Commits zum Merge-Request befindet. Sie erhalten dann entsprechend kommentierte Quelltexte. Nachfolgend sehen Sie ein Beispiel:

