



# Anleitung: Git und Gitlab

## 1 Git

Git ist ein verteiltes Versionsverwaltungssystem. Sie können es lokal auf Ihrem Rechner nutzen oder aber z. B. mit anderen Mitgliedern der Universität gemeinsam Ihre Projekte im vom Rechenzentrum bereitgestellten Webfrontend Gitlab verwalten. Mit Git arbeiten Sie vorwiegend offline. Dies unterscheidet Git von dem bisher vom Rechenzentrum angebotenen Versionsverwaltungssystem Subversion.

**Eine ausführliche Dokumentation zu Git finden Sie auf [git-scm.com/book/de](https://git-scm.com/book/de).**

Einen Überblick über die Kommandozeilenbefehle erhalten Sie auf [git-scm.com/docs](https://git-scm.com/docs).

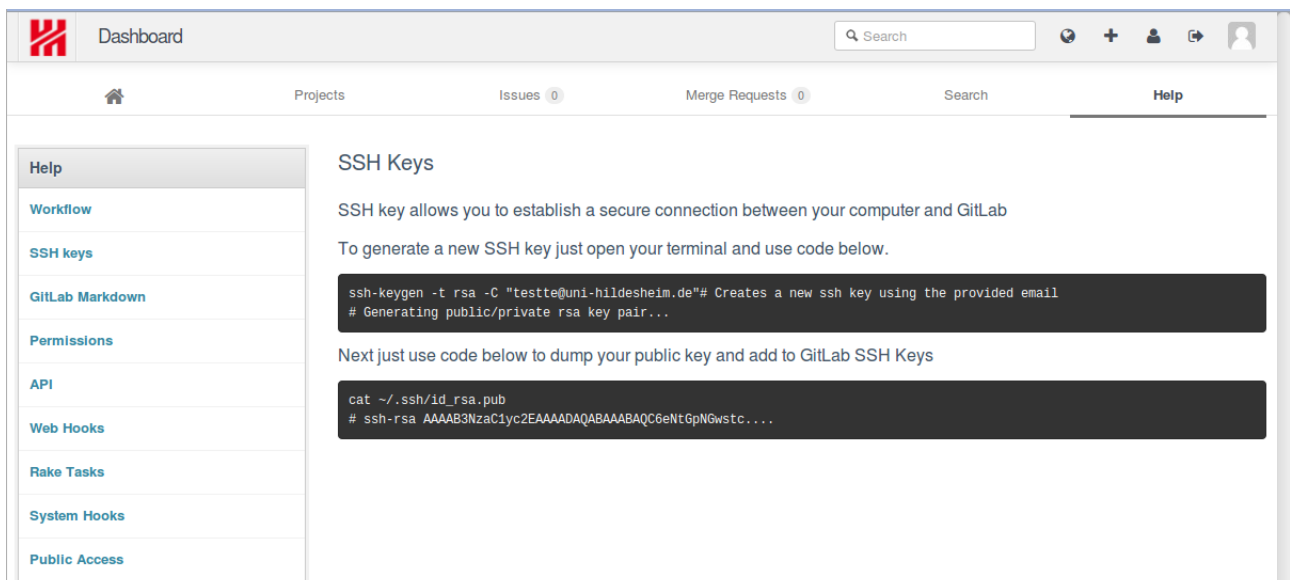
Git selbst können Sie von [git-scm.com/downloads](https://git-scm.com/downloads) herunterladen bzw. im Paketverwaltungsprogramm Ihres freien Betriebssystems finden.

## 2 SSH-Schlüssel erzeugen und ablegen

Auf [uni-hildesheim.de/gitlab](https://uni-hildesheim.de/gitlab) loggen Sie sich mit Ihrer RZ-Kennung ein.



Als Erstes müssen Sie einen SSH-Schlüssel in Gitlab ablegen. Erzeugen Sie ggf. einen mit den auf [uni-hildesheim.de/gitlab/help/ssh](https://uni-hildesheim.de/gitlab/help/ssh) genannten Befehlen und fügen Sie ihn auf [uni-hildesheim.de/gitlab/keys/new](https://uni-hildesheim.de/gitlab/keys/new) hinzu.



Profile

Search

Account Notifications **SSH Keys 0** Design History

### Add an SSH Key

Title

Key 

```
ssh-rsa
ZZZZB3NzaC1yc2EAAAADAQABAAQCrSePp8ADohvV0kYUnDN++/q7C85EH+XaSd
qSugMM7gf43kiYFBiKHvs3dlXYpWQXYQ32OcgDcHkIZgHlghddXUuPywgtSqp7IU9LcpJ
VJfleMy8kDJVZiJ4S1vuGFRSvTGKojGvCGis8PQs56JsvCAKq7RB4RYPJe8iA18KE71fJ+
4hjUN4pRxk8Urw57W4Rcn799d9hfZG399LSXAcC3ApCMiniJrBhIE1c1gsu31jAT7Xm8Uy
AG95SP7rw35uTLNJuEUR+0YoMV7DxBd+cjvhj0gF8MB10BKjOQM0PpmhnrVl8S0yRO5
.MVk/WhE8P5QAhc1jVQJUBr7cgV testte@uni-hildesheim.de
```

Paste your public key here. Read more about how generate it [here](#)

### 3 Projekt in Gitlab erstellen

Ein neues Git-Projekt legen Sie unter [uni-hildesheim.de/gitlab/projects/new](https://uni-hildesheim.de/gitlab/projects/new) an.

Dashboard

Search

Projects Issues 0 Merge Requests 0 Search Help

### New Project

Project name is

[Import existing repository?](#)

New projects are private by default. You choose who can see the project and commit to repository.

Need a group for several dependent projects?

Want to share a project between team?

Mit den auf [uni-hildesheim.de/gitlab/kennung/projektname](https://uni-hildesheim.de/gitlab/kennung/projektname) genannten Befehlen richten Sie bei sich das entsprechende Repository ein. Es empfiehlt sich, zunächst ein Git-Verzeichnis anzulegen und dann darin zu arbeiten („mkdir gitrepos“, „cd gitrepos“, „mkdir testprojekt“ usw.).

testprojekt

Search

Issues 0 Merge Requests 0 Wiki Settings

SSH HTTPS `ssh://git@www.uni-hildesheim.de:7920/testte/testprojekt.git` private

Git global setup:

```
git config --global user.name "Test Test"
git config --global user.email "testte@uni-hildesheim.de"
```

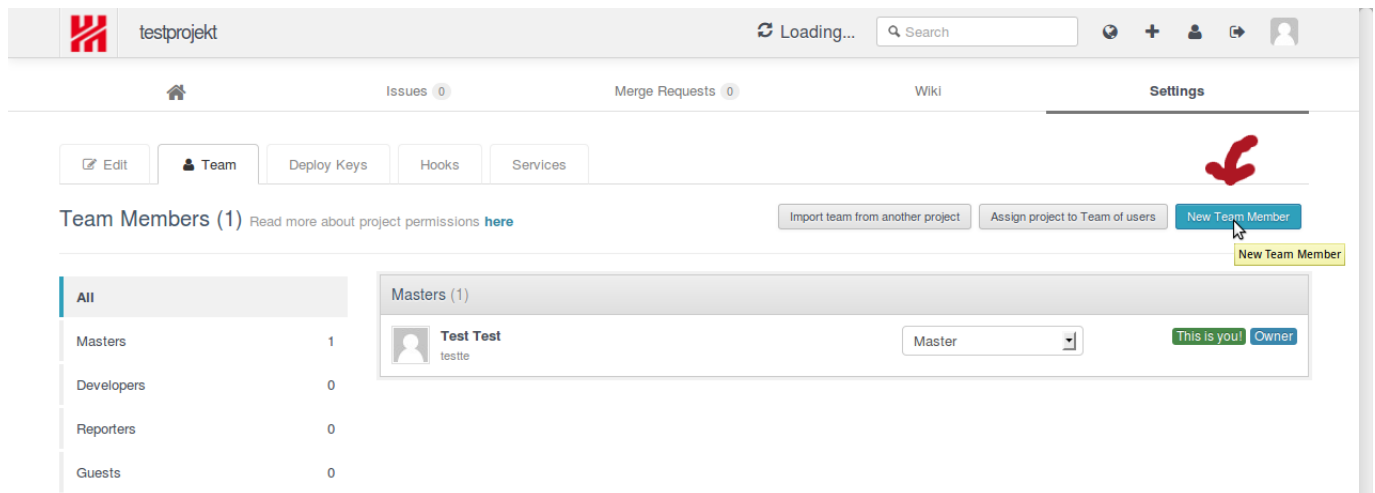
Create Repository

```
mkdir testprojekt
cd testprojekt
git init
touch README
git add README
git commit -m "first commit"
git remote add origin ssh://git@www.uni-hildesheim.de:7920/testte/testprojekt.git
git push -u origin master
```

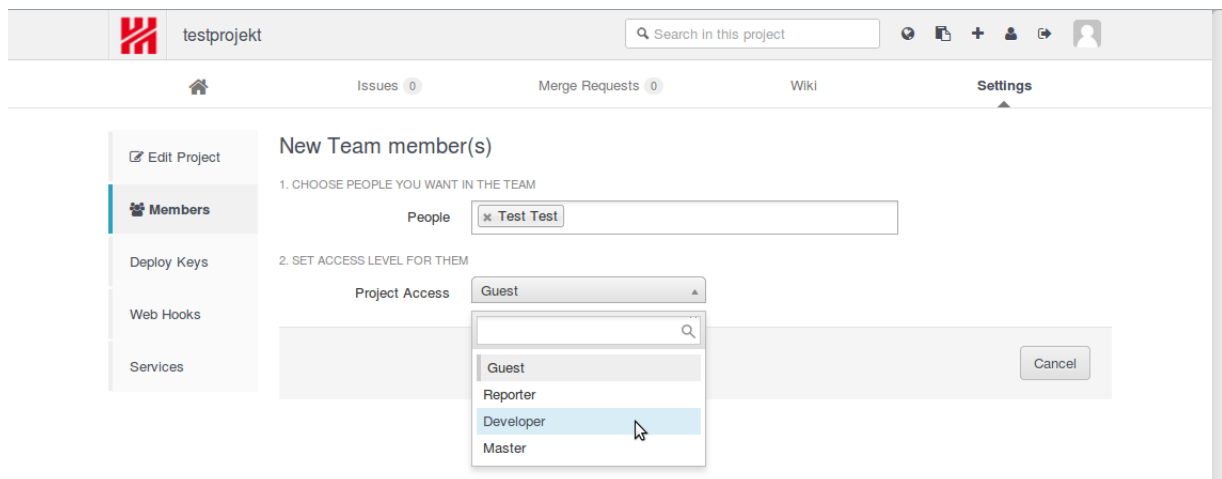
Existing Git Repo?

```
cd existing_git_repo
git remote add origin ssh://git@www.uni-hildesheim.de:7920/testte/testprojekt.git
git push -u origin master
```

Auf [uni-hildesheim.de/gitlab/kennung/projektname/team](https://uni-hildesheim.de/gitlab/kennung/projektname/team) können Sie nun Ihrem Projekt weitere Mitglieder hinzufügen.



Geben Sie die RZ-Kennungen der gewünschten Mitglieder in das Suchfeld ein und legen Sie im Ausklappmenü deren Rollen fest. **Wichtig: Die entsprechenden Mitglieder müssen schon einmal in Gitlab eingeloggt gewesen sein, sonst können sie nicht gefunden werden.**



## 4 Alltag mit Git

Ab jetzt gehen Sie mit Ihren Projektdateien ähnlich vor wie mit der eingangs benutzten „README“:

1. ins Repository wechseln
2. ggf. Stand des Projekts herunterladen („pullen“)
3. Dateien erstellen oder ändern
4. vormerken („adden“)
5. speichern („committen“)
6. hochladen („pushen“)

Dabei benötigen Sie erst im letzten Schritt eine Netzwerkverbindung. Gitlab ermöglicht Ihnen dabei einen grafischen Überblick über die Entwicklung Ihres Projekts.

Für die weitergehende Verwendung von Git (Änderungsanalyse, Zurücksetzen, Klonen, Abzweigen und Zusammenfügen ... ) sei nochmals auf die ausführliche Dokumentation auf [git-scm.com/doc](https://git-scm.com/doc) verwiesen. Außerdem existieren einige Podcasts, die sich bereits informativ mit Git beschäftigen haben.