

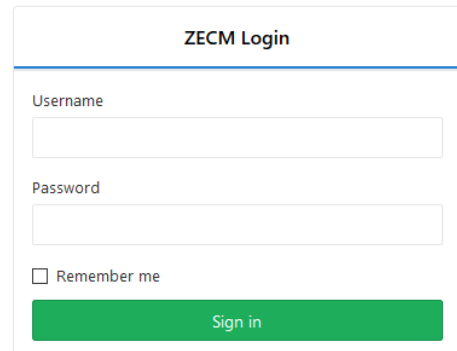
## TU Gitlab Schnelleinstieg

Als erstes loggt ihr euch unter [gitlab.tubit.tu-berlin.de] mit euren Tubit-Logindaten ein (die gleichen wie bei ISIS, vgl. Abbildung [1]).

### ZECM - GitLab Dienst

#### Open source software to collaborate on code

Manage git repositories with fine grained access controls that keep your code secure. Perform code reviews and enhance collaboration with merge requests. Each project can also have an issue tracker and a wiki.



ZECM Login

Username

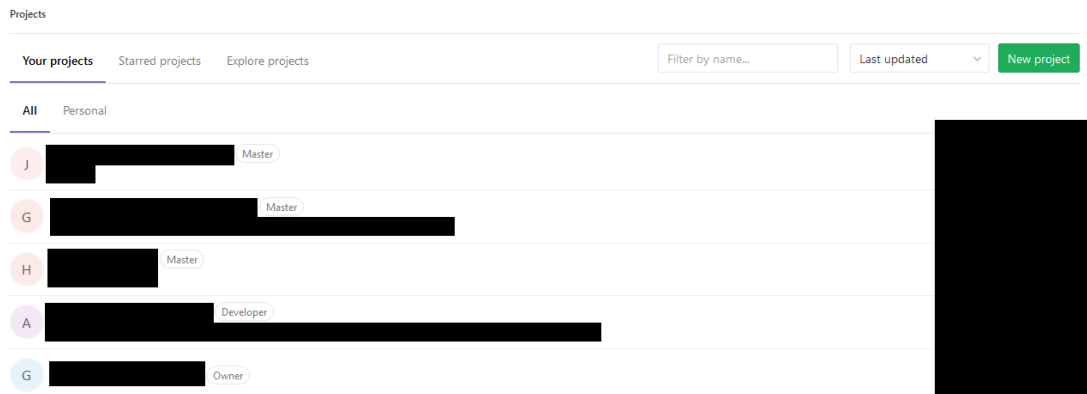
Password

Remember me

Sign in

Abbildung 1: [1] Login Screen.

Sobald ihr eingeloggt seid werden euch eure vergangenen bzw. aktiven Projekte angezeigt. Ihr wollt nun ein neues Projekt erstellen und klickt dazu auf den „New Project“ Button oben rechts (vgl. Abb. [2]).



Projects

Your projects Starred projects Explore projects

Filter by name... Last updated New project

All Personal

J [redacted] Master

G [redacted] Master

H [redacted] Master

A [redacted] Developer

G [redacted] Owner

Abbildung 2: [2] Home Screen mit Projektliste.

Nun müsst ihr den Namen eures Projekts angeben und könnt eine Beschreibung hinzufügen (optional). Außerdem könnt ihr euch entscheiden ob das Repository öffentlich (public), nur für TU-Angehörige (internal) oder nur für euch (private) zugänglich sein soll. Für unsere Zwecke wählen wir private. Bestätigt nun eure Einstellungen mit „Create project“ (vgl. Abb. 3).

## New project

A project is where you house your files (repository), plan your work (issues), and publish your documentation (wiki), among other things.

All features are enabled when you create a project, but you can disable the ones you don't need in the project settings.

Blank project
Create from template
Import project

**Project path**

**Project name**

Want to house several dependent projects under the same namespace? [Create a group](#)

**Project description (optional)**

beschreibung.

**Visibility Level**

**Private**  
Project access must be granted explicitly to each user.

**Internal**  
The project can be accessed by any logged in user.

**Public**  
The project can be accessed without any authentication.

Create project
Cancel

Abbildung 3: [3] Anlegen eines neuen Projekts.

Nun werdet ihr auf die in Abbildung [4] gezeigte Seite weitergeleitet. Ihr müsst nun für die folgenden Schritte Git bereits auf eurem Computer installiert haben. Wenn nicht dann ladet es unter [\[https://git-scm.com/downloads\]](https://git-scm.com/downloads) herunter. Stellt sicher, dass ihr vom SSH Protokoll auf HTTPS

testtest > Details

T

**testtest**

beschreibung.

☆ Star
0
HTTPS
https://gitlab.tubit.tu-berl:
+
Global

**The repository for this project is empty**

If you already have files you can push them using command line instructions below.

Otherwise you can start with adding a [README](#), a [LICENSE](#), or a [.gignore](#) to this project.

You will need to be owner or have the master permission level for the initial push, as the master branch is automatically protected.

**Command line instructions**

**Git global setup**

```
git config --global user.name " "
git config --global user.email " @campus.tu-berlin.de"
```

**Create a new repository**

```
git clone https://gitlab.tubit.tu-berlin.de/ /testtest.git
cd testtest
touch README.md
git add README.md
git commit -m "add README"
git push -u origin master
```

**Existing folder**

```
cd existing_folder
git init
git remote add origin https://gitlab.tubit.tu-berlin.de/ /testtest.git
git add .
git commit -m "Initial commit"
git push -u origin master
```

**Existing Git repository**

```
cd existing_repo
git remote rename origin old-origin
git remote add origin https://gitlab.tubit.tu-berlin.de/ /testtest.git
git push -u origin all
git push -u origin --tags
```

Abbildung 4: [4] Initialisierung des Repositorys.

umgeschaltet habt. Danach führt die bei euch unter „Git global setup“ und „Create a new repository“ aufgeführten Befehle in der Git-Bash aus (nicht vergessen vorher in den euren Projektordner zu navigieren!) um das Repository zu initialisieren. Nun könnt ihr alle weiteren Git Befehle in der Git-Bash benutzen und euer Repository benutzen.

Ihr müsst nun noch eure Gruppenpartner hinzufügen, damit diese auch Zugriff auf das Repository bekommen. Dafür geht ihr links im Menü auf „Settings“ und dann auf den Unterpunkt „Members“ (vgl. Abbildung [5]).

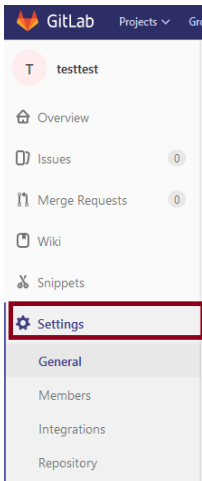


Abbildung 5:  
[5] Menü im  
linken Bereich  
der Website.

Die Oberfläche sollte nun wie in dem Screenshot in Abbildung [6] aussehen. Ihr könnt in der Zeile „Select members to invite“ nach euren Gruppenmitgliedern suchen und diese als Developer oder Master Hinzufügen. Um das Hinzufügen abzuschliessen klickt auf „Add to project“.

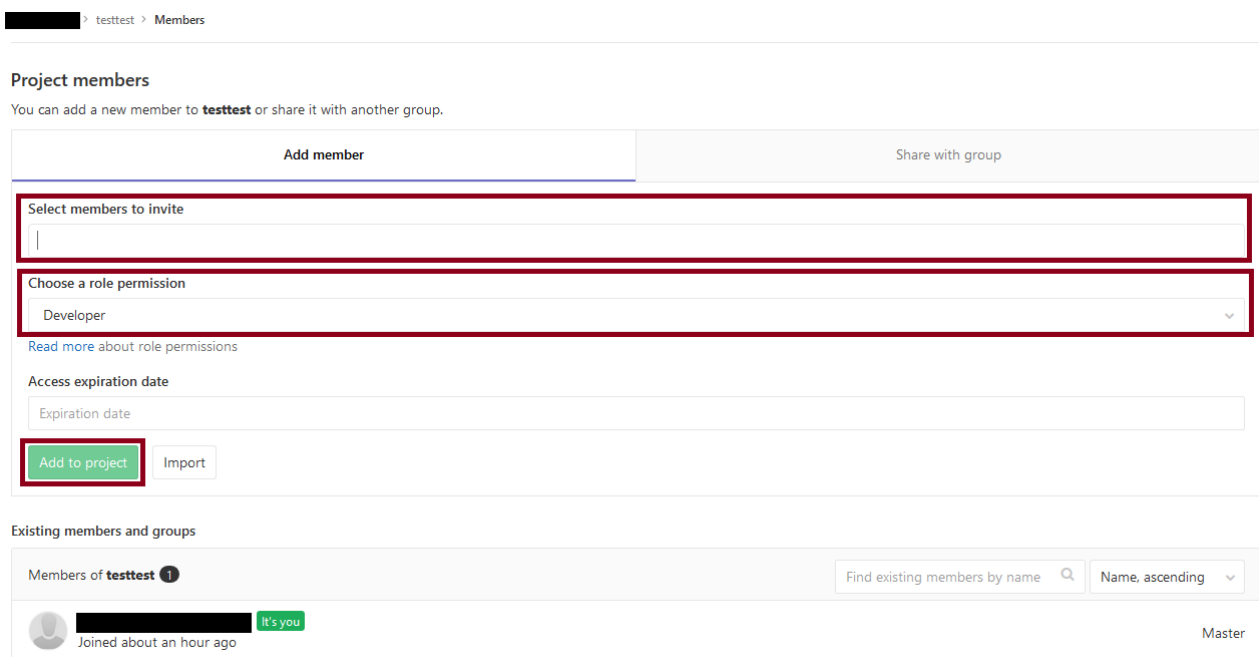


Abbildung 6: [6] Hinzufügen von Gruppenpartnern.

Nun sollte jeder von euch Zugriff auf das gemeinsame Repository haben und in der Lage sein Dateien zu verändern und zu schreiben. Den Link eures Repositorys findet ihr immer unter dem Reiter „Overview“, wenn ihr bei Gitlab eingeloggt seid und euer Projekt ausgewählt habt.