

Was ist ein git-Repository?

Ein git-Repository ist eine Sammlung von Dateien, die alle gemeinsam verwaltet werden. Dabei spricht man von einer so genannten "Versionsverwaltung" — in einem git-Repository wird zu jeder Datei genau aufgezeichnet, wann eine Änderung gemacht wurde und wer sie gemacht hat. Wenn man Dateien in einem git-Repository verwaltet, kann man immer wieder zu dem Stand eines bestimmten Zeitpunkts zurückkehren.

Wozu benutzt Teckids git?

Die gesamte öffentliche Vereinsarbeit findet in einem git-Repository statt, so dass alle Mitglieder immer Zugriff auf die Daten haben und auch Nichtmitglieder jederzeit in die Arbeit hineinsehen können. Zum Beispiel unsere Website, die Protokolle unserer Sitzungen und das freie Unterrichtsmaterial liegen in unserem git-Repository.

Wie bekommt man git?

Für die meisten Betriebssysteme, die wir unterstützen, gibt es git in einem Paket. Unter Debian zum Beispiel reicht der folgende Befehl, um git zu installieren:

```
sudo apt-get install git
```

Das Vereins-Repository bekommen

Als erstes muss man sich eine Kopie des Vereins-Repositorys, einen so genannten Klon, besorgen. Mitglieder benutzen hierfür den SSH-Zugang auf terra.teckids.org. Damit das richtig funktioniert, muss zunächst der SSH-Client richtig konfiguriert werden. In der Datei `~/.ssh/config` muss der folgende Eintrag gemacht werden:

```
Host terra
  Hostname terra.teckids.org
  User <dein Benutzername>
```

Danach kann das Repository geklont werden:

```
git clone terra:/srv/git/verein.git teckids
```

Das Kommando legt ein neues Verzeichnis namens `teckids` an und lädt darin das Repository herunter.

Bevor du aber etwas im Repository änderst, musst du noch deinen Namen und deine E-Mail-Adresse für die Aufzeichnung der Änderungen konfigurieren:

```
git config --global user.name "Vorname Nachname"  
git config --global user.email "vorname.nachname@teckids.org"
```

Änderungen im Repository machen

Nun kannst du Dateien im Repository hinzufügen, ändern oder löschen. Dein Klon ist ein ganz normales Verzeichnis mit ganz normalen Dateien. Bitte überlege dir, wenn du neue Dateien anlegst, gut, in welches Unterverzeichnis sie gehören.

Bevor du Änderungen machst, solltest du deinen Klon zunächst auf den neuesten Stand bringen. Das machst du ganz einfach mit dem Kommando:

```
git pull
```

Danach machst du deine Änderungen einfach mit deinen Lieblingsprogrammen. Jede Datei, die du hinzufügst oder veränderst, musst du git danach bekanntgeben:

```
git add dateiname.txt
```

Immer, wenn du einen *logischen Schritt* gemacht hast—zum Beispiel alle Änderungen gemacht hast, die zu einer einzelnen Aufgabe oder Idee gehören, musst du git dazu auffordern, deine Änderungen in einem so genannten Commit aufzunehmen. Dabei ist es besonders wichtig, dass du eine aussagekräftige Nachricht zu deinem Commit angibst.

```
git commit
```

Nach diesem Kommando öffnet sich dein Texteditor, in dem du ganz oben deine Nachricht eingibst. Schau auch genau hin, ob wirklich genau die Dateien im Commit aufgenommen wurden, die du verändert hast!

Vergiss bitte, wenn du mit deiner Arbeit fertig bist, nicht, die Commits aus deinem Klon auch wieder auf den Server hochzuladen. Das machst du mit:

```
git push
```

Manchmal kommt es vor, dass zwischendurch jemand seine eigenen Commits auf den Server hochgeladen hat. Dann gibt dir git eine Fehlermeldung aus und fordert dich dazu auf, diese Commits erst herunterzuladen. Meistens klappt das ganz einfach mit:

```
git pull --rebase
```

Sollte dabei ein weiterer Fehler auftreten, kannst du bei einem erfahrenen git-Benutzer um Hilfe bitten.

Was gehört ins Repository und was nicht?

Im Repository müssen alle Ergebnisse der Vereinsarbeit aufgenommen werden, aber es dürfen dabei keine persönlichen Daten veröffentlicht werden!

Solltest du Zweifel haben, melde dich bitte bei uns.