

Anleitungen zur technischen Einrichtung:

- Kapitel 1: SSH
- Kapitel 2: GIT
- Kapitel 3: TRAC

Kapitel 1: SSH:

Bei uns laufen die Verbindungen zu den GIT Repositories über SSH-Verbindungen. Damit ihr also auch Zugriff auf euer GIT Repository habt, müsst ihr euch zunächst SSH-Schlüssel generieren. Dabei ist folgendes zu beachten:

1. die Keys sollen ohne (bzw. leeres) Passwort/Passphrase und mit SHA-2 und 2048 bits erzeugt werden.
2. bei EUERNAME im Konsolenbefehl setzt ihr bitte euren Benutzernamen ein, den ihr wie folgt definieren müsst:

Anfangsbuchstabe des Vornamens + Nachname (alles klein geschrieben)

Für Max Muster wäre das also mmuster.

3. Es werden mehrere Dateien generiert. den **Public-Key (mit der Endung *.pub)** müsst ihr unserem Admin Frank Schuh entweder per Mail (schuh@informatik.uni-siegen.de) zuschicken oder auf einem USB-Stick bei ihm vorbei bringen (H-C 8302). Gebt ihm auch bitte euren **Namen** und eure **Emailadresse!**

Wichtig: Den anderen Key (Private-Key) gebt ihr bitte nicht raus!!

4. Bei euch lokal sollten die Keys liegen in:
 - Windows: C:\Benutzer\DeinWindowsBenutzerName\.ssh\
 - Mac/Linux: ~/.ssh/

5. Danach im Betriebssystem ab- und neu anmelden.

Generierung unter Linux/Mac in der Konsole:

```
ssh-keygen -q -t rsa -b 2048 -P "" -f EUERNAME
```

(zu EUERNAME siehe Hinweise oben)

Generierung unter Windows mit separaten Tools (z.B. PuTTY):

Download: <http://www.putty.org/>

Anleitung: <https://www.howtoforge.de/anleitung/key-basierte-ssh-logins-mit-putty/2/>

Kapitel 2: GIT:

Sobald ihr eure SSH-Keys eingerichtet habt und die Public-Keys von unserem Admin zusammen mit euren Namen und Emailadressen hinterlegt wurden, könnt ihr GIT einrichten.

In den neueren Eclipse-Versionen ist GIT inklusive der Benutzerschnittstelle „egit“ direkt mit an Bord.

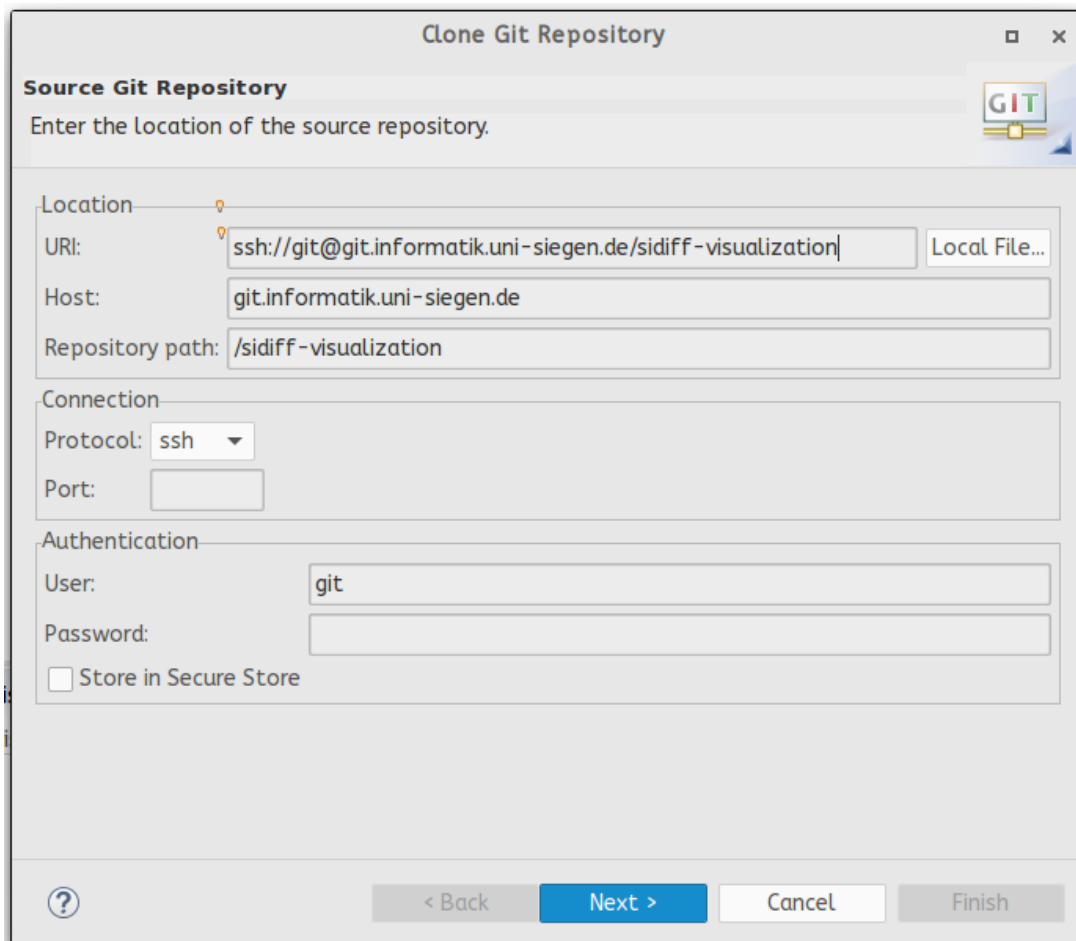
Zunächst muss die GIT-Perspektive in Eclipse geöffnet werden:

Window > Perspectives > Open Perspectives > Other > GIT.

Danach könnt ihr rechts in der Repository-View auf das folgende Symbol zum „clonen“ des Repositories klicken:



Den Dialog könnt ihr wie folgt konfigurieren:



Kleine Anmerkungen zu GIT:

(1) Ihr solltet euch mit der Funktionsweise von GIT vertraut machen. Da gibt es diverse Tutorials im Netz. GIT unterscheidet sich wesentlich vom CVS und SVN.

Ein großer Unterschied ist z.B. dass das komplette Repository auf eure Festplatte **ge-cloned** (also kopiert) wird und ihr damit ein lokales, eigenes Repository besitzt (das **Working Directory**). Wenn Änderungen an den Remote-Server gesendet werden sollen, müssen eure Änderungen immer erst lokal **comitted** in das eigene lokale Repository und anschließend in das Remote-Repository **gepusht** werden.

(2) Das geklonte GIT Repository sollte bei euch niemals gleich eurem Eclipse-Workspace Ordner sein! Legt es in einem separaten Benutzerordner an z.B. c:\Benutzer\ICH\git\sidiff-visualization. Sonst comitted man u.U. allen möglichen Projekte, die man nur zum Testen temporär angelegt hat.

(3) Weitere Infos:

http://wiki.eclipse.org/EGit/Git_For_Eclipse_Users

http://wiki.eclipse.org/EGit/User_Guide

Kapitel 3: TRAC

TRAC ist ein Verwaltungssystem, welches mit Repositories verknüpft werden kann. In TRAC kann man zum Beispiel Tickets mit Aufgaben oder zu lösenden Bugs eintragen, den Projektablauf mit Meilensteinen planen, ein Wiki mit relevanten Informationen für alle Projektteilnehmer anlegen und die eigenen Arbeitszeiten erfassen. Genau diese Dinge sollt ihr mit TRAC machen und werden von uns gefordert!

Das TRAC für eurer Projekt ist erreichbar mit dem Browser unter:

https://trac.informatik.uni-siegen.de/stp_ws1516_2

Dort könnt ihr euch mit den von euch gewählten Login-Daten anmelden, die ihr auch vorher unserem Admin gegeben habt.

Eine direkte Integration in Eclipse ist auch möglich, über das MyLyn Framework:

<http://www.vogella.com/tutorials/Mylyn/article.html>

Als Beispiel könnt ihr das TRAC der vorigen Projektgruppe anschauen:

https://trac.informatik.uni-siegen.de/pg_svw