

# Versionsmanagement

EIN VERGLEICH ZWISCHEN GIT  
UND SUBVERSION (SVN)

# Inhalt

- Versionsmanagement allgemein
  - Was ist Versionsmanagement
  - Gründe für Versionsmanagement
- Kurzeinführung GIT
  - Beispiel mit GIT
- Kurzeinführung SVN
- Unterschiede der Tools
- Fazit: Bevorzugtes System

# Versionsmanagement allgemein

WAS IST VERSIONSMANAGEMENT

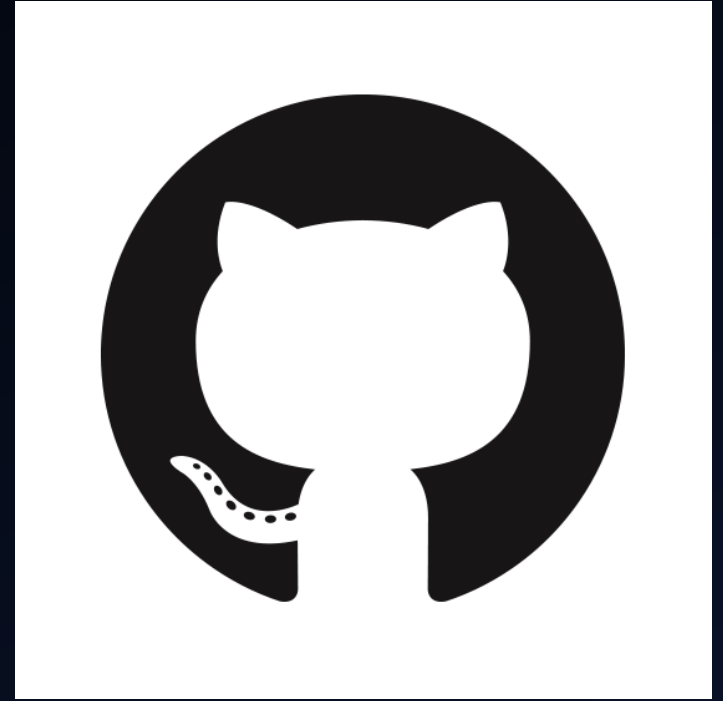
GRÜNDE FÜR VERSIONSMANAGEMENT

# Was ist Versionsmanagement?

- Werkzeug zum Verwalten von Dokumenten, Codes o.ä.
  - Beispiel: datei\_finale\_version\_5\_2.doc
- Aufgaben
  - Protokollieren von Änderungen
  - Archivierung von Momentanzuständen eines Projekts
  - Wiederherstellung alter Stände einzelner Dateien
  - Koordination mehrerer Teammitglieder auf ein Projekt

# Gründe für Versionsmanagement

- Geeignet für Arbeiten mit mehreren Entwicklern
  - Zuteilen von Rechten (Lese-, Schreibe-, Adminrechte)
- Arbeiten an Quelltexten und Programmcodes
  - Änderungen durch Admin rücksetzbar
- Private Versionsverwaltung mit mehreren Systemen
  - Z.B. bei Gebrauch von Desktop-PC (zu hause) – Laptop (unterwegs)

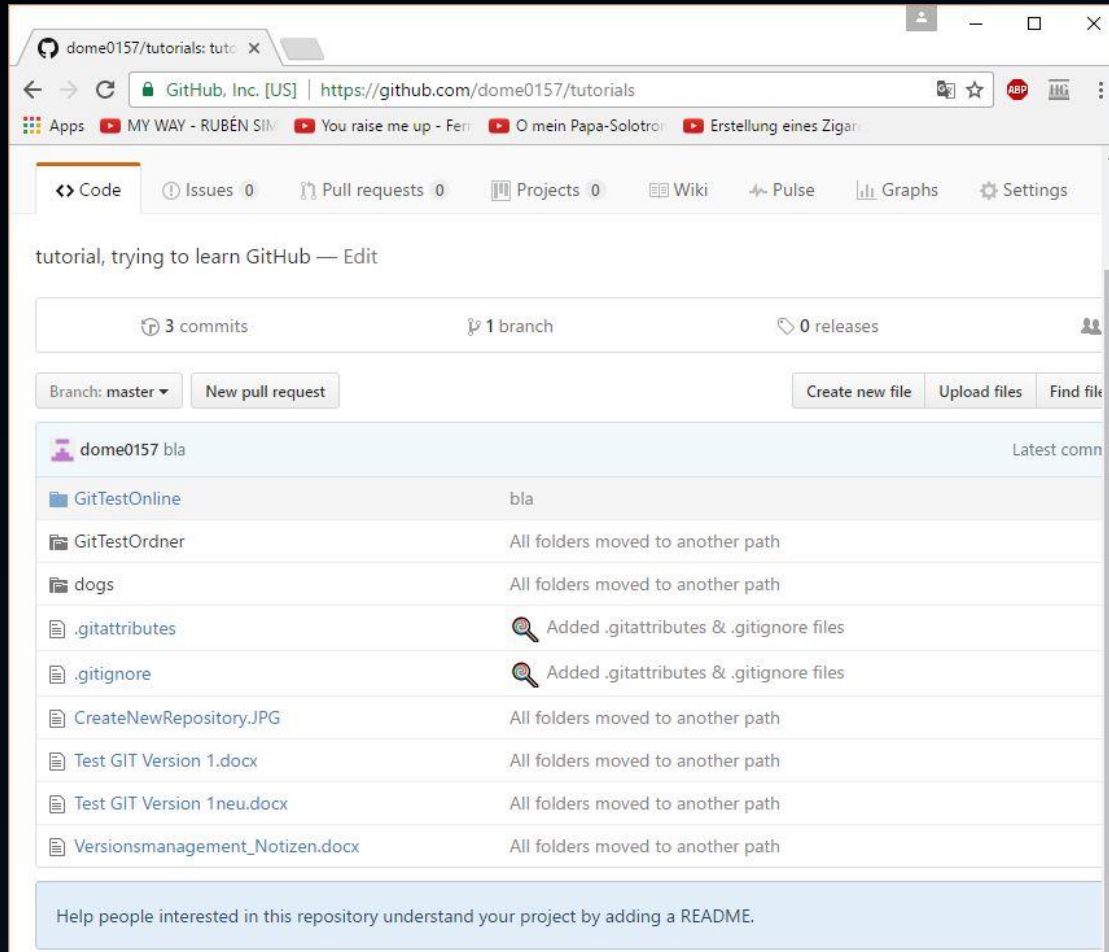


# Kurzeinführung GIT

# Kurzeinführung GIT

- Mächtiges Werkzeug zur Verwaltung von Dateien
- 3 Systeme
  - Kommandobox
  - Onlinezugang
  - GitHub Desktop
- Team & Organisationsverwaltung online möglich
  - Rechtezuteilung

# Kurzeinführung GIT



```
MINGW64:/g/Arbeit/HS Hannover/Semester 6/Software Engineering/Vortra...
Domenic@DomenicPC MINGW64 /g/Arbeit/HS Hannover/Semester 6/Software Engineering
/Vortrag Versionsmanagement (master)
$ git remote add githubRepo https://github.com/dome0157/tutorials.git

Domenic@DomenicPC MINGW64 /g/Arbeit/HS Hannover/Semester 6/Software Engineering
/Vortrag Versionsmanagement (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
(use "git push" to publish your local commits)
nothing to commit, working tree clean

Domenic@DomenicPC MINGW64 /g/Arbeit/HS Hannover/Semester 6/Software Engineering
/Vortrag Versionsmanagement (master)
$ git push -u githubRepo master
Counting objects: 14, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (12/12), done.
Writing objects: 100% (14/14), 75.44 KiB | 0 bytes/s, done.
Total 14 (delta 3), reused 0 (delta 0)
remote: Resolving deltas: 100% (3/3), done.
To https://github.com/dome0157/tutorials.git
 * [new branch]      master -> master
Branch master set up to track remote branch master from githubRepo.

Domenic@DomenicPC MINGW64 /g/Arbeit/HS Hannover/Semester 6/Software Engineering
/Vortrag Versionsmanagement (master)
$ |
```

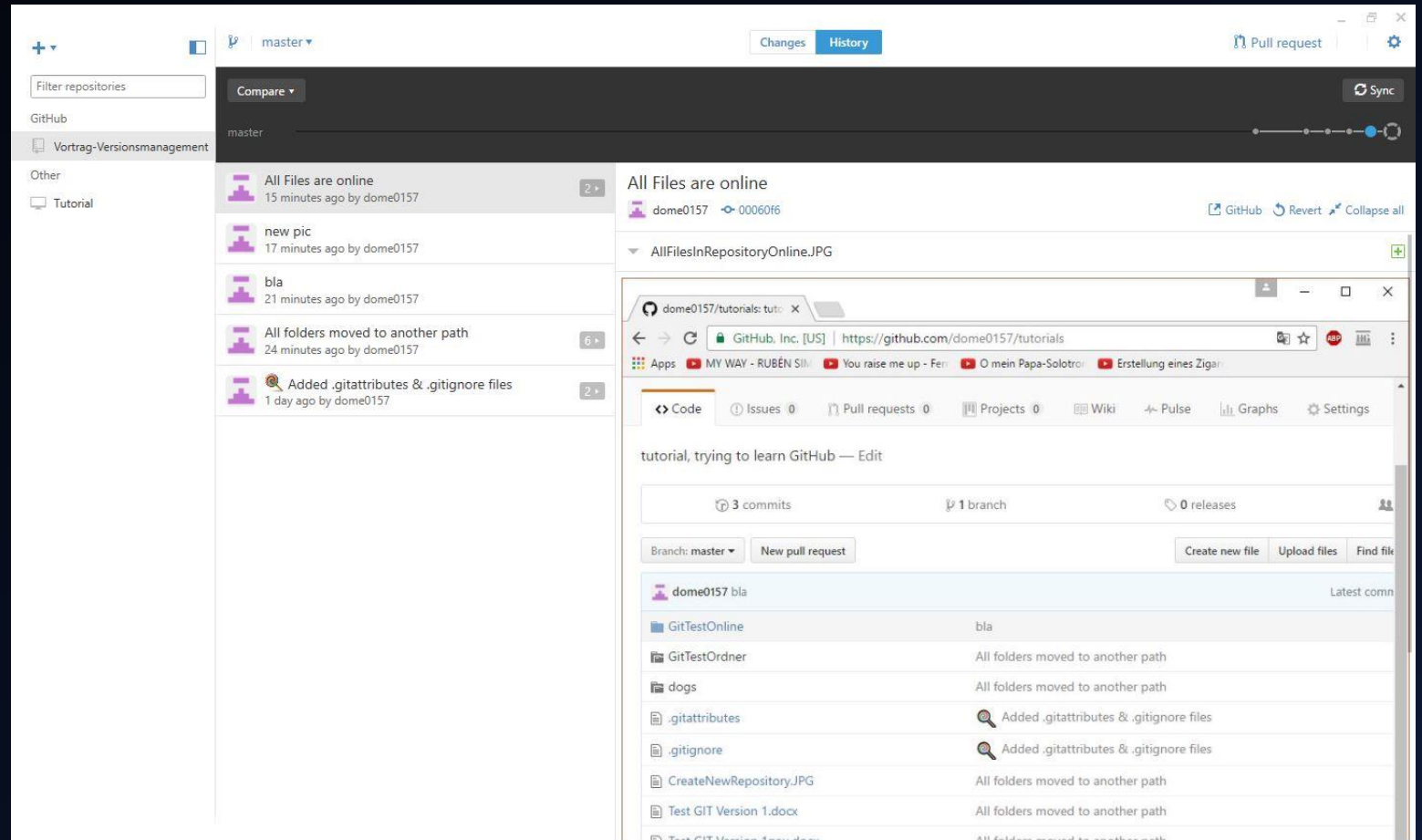
↑  
GIT - Konsole

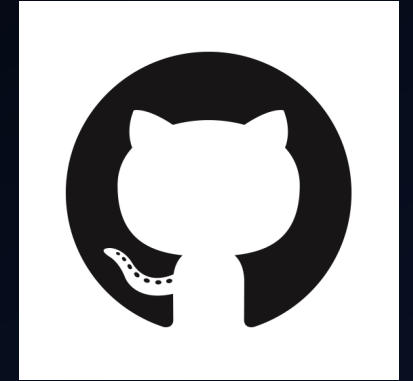
← GitHub - Online



# Kurzeinführung GIT

GitHub - Desktop





# GitHub - Beispiel

ALLGEMEINE MÖGLICHKEITEN

BEISPIEL MIT ANDROID-STUDIO



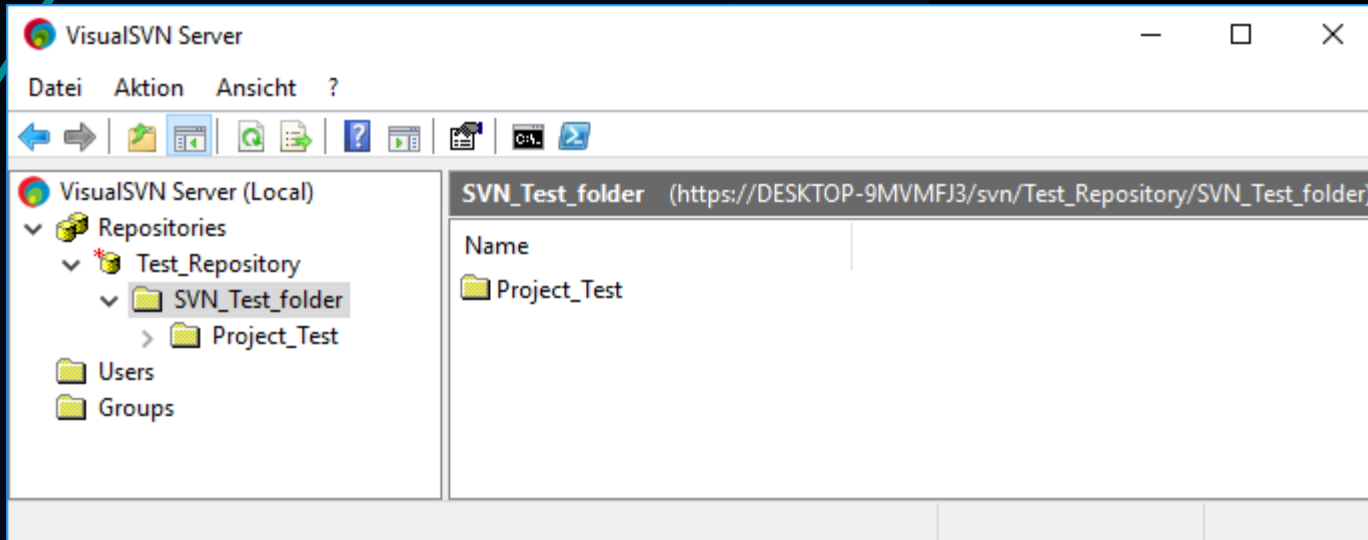
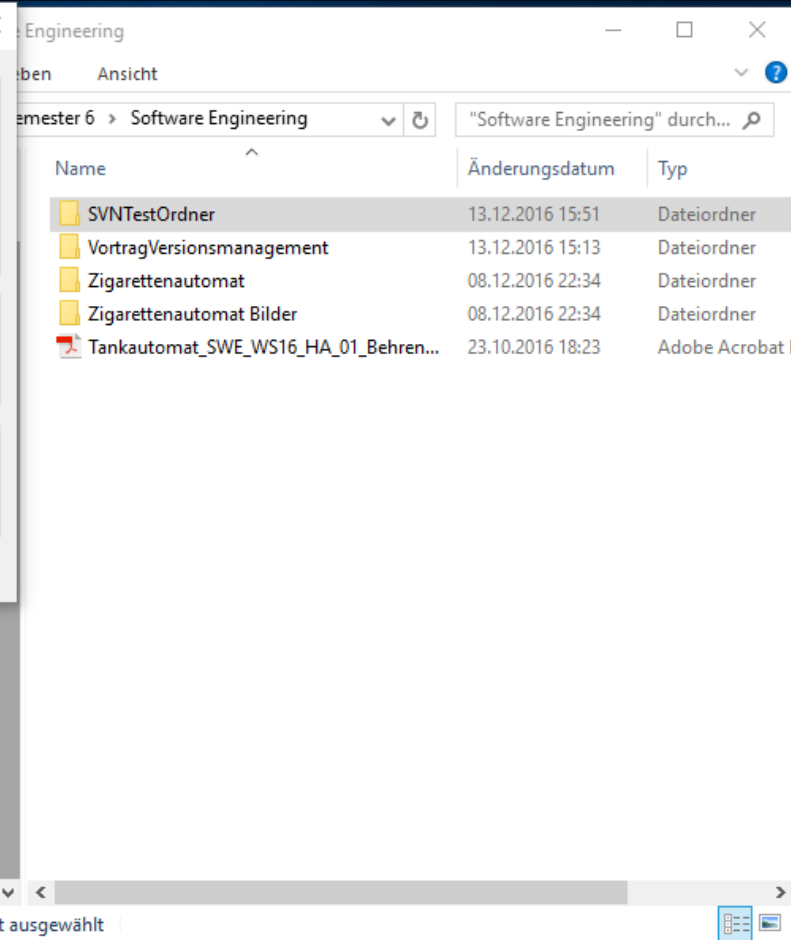
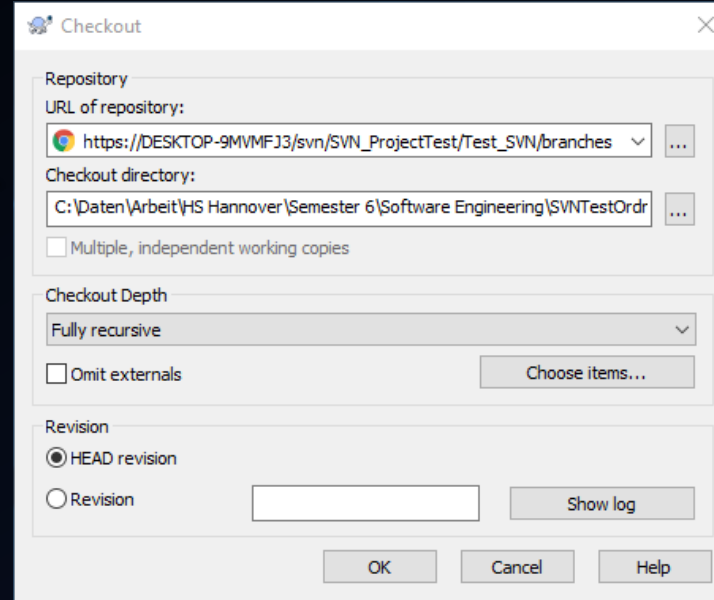
# Kurzeinführung SVN

# Kurzeinführung SVN

- SVN: Apache Subversion
- Online per URL
- Per Verzeichnisse auf Computer
- Client-Server-Struktur
  - Aufbau eines Servers mit VisualSVN Server
  - URL-Erzeugung z.B. [https://DESKTOP-9MVMFJ3/svn/Test\\_Repository](https://DESKTOP-9MVMFJ3/svn/Test_Repository)

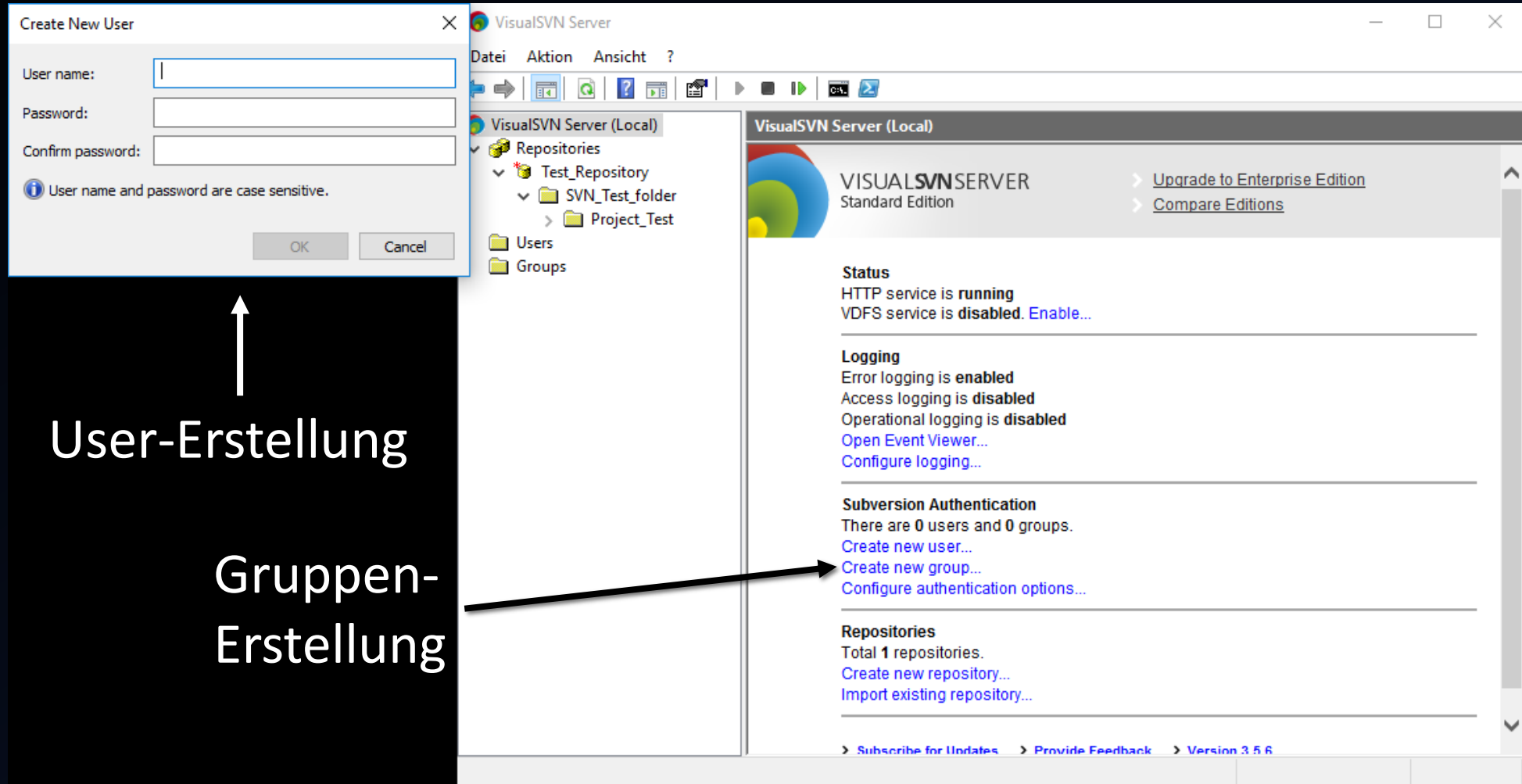
# Kurzeinführung SVN

Checkout-Box



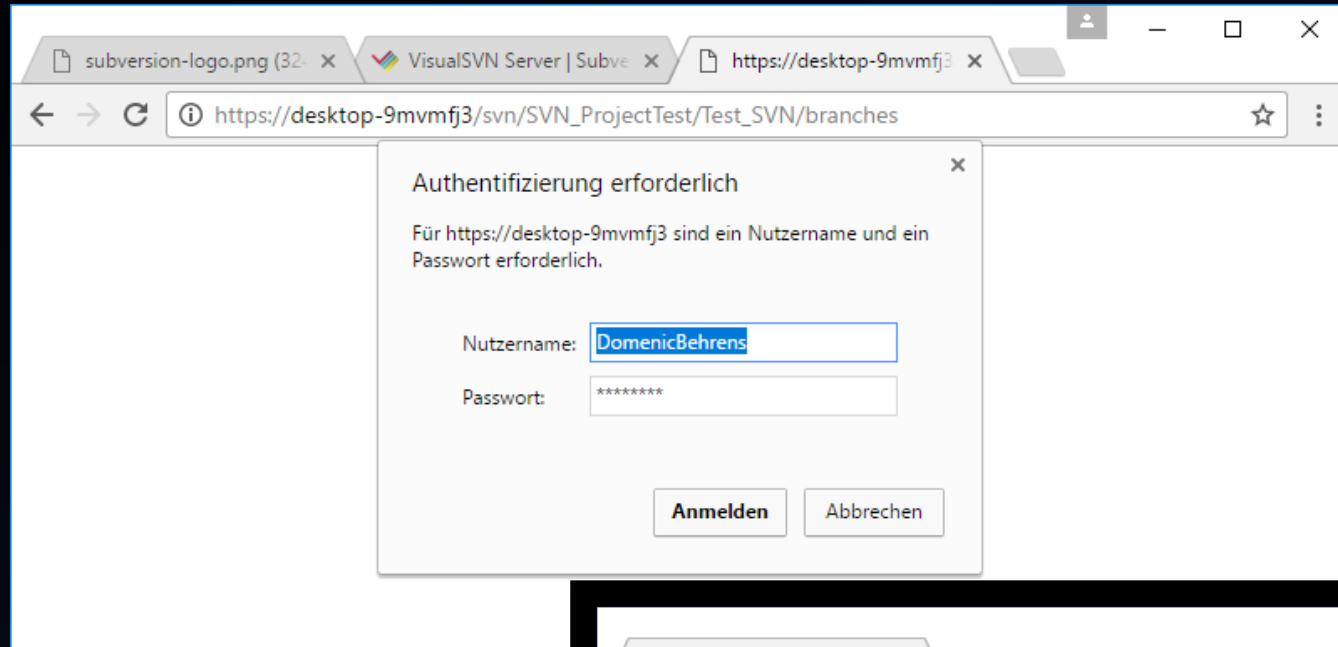
← VisualSVN – Server

# Kurzeinführung SVN



- Login per URL und user-profile

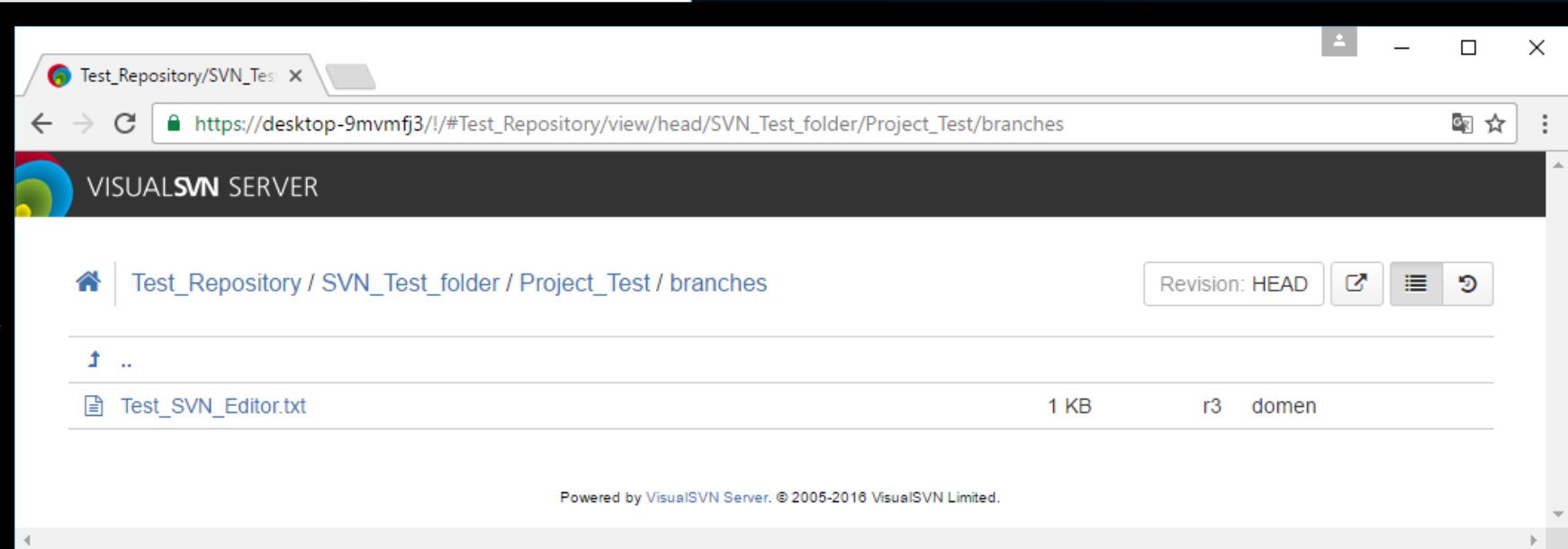
# Kurzeinführung SVN



Login per URL und user-profile



Datei in Repository  
-abgerufen über Login



# Unterschiede der Tools

## GITHUB

- Online, Kommandobox, Desktop Anwendung
- Hauptsächlich Online-Server
- User einladen in GitHub-Online
- Bedienung: Recht Intuitiv und einfach

## SVN

- Per Verzeichnisstruktur
- Hauptsächlich Server auf eigenem PC
- Per URL-Erzeugung
- User Erstellung in VisualSVN Server
- Bedienung: nicht ganz so intuitiv, eher komplex





FAZIT

WELCHES TOOL IST BESSER



# Quellen

- <https://subversion.apache.org/>
- <https://git-scm.com/>
- <https://help.github.com/>

# Vielen Dank für Ihre Aufmerksamkeit

DOMENIC BEHRENS (1295429)

PATRICK SAWADSKI (1296017)

HS HANNOVER – FAKULTÄT 1