

Versionierung mit GitLab & Subversion

Versionsverwaltung von Dokumenten und Dateien

1. Übersicht GitLab & Subversion

Aktuell werden an der Universität Hamburg zwei Dienste zur Versionsverwaltung von Dateien angeboten: [GitLab & Subversion](#).

GitLab kann über den Browser oder über einen installierten Client bedient werden und ist für Inhalte aus Forschung und Lehre unter <https://gitlab.rrz.uni-hamburg.de/> erreichbar; außerhalb von Forschung und Lehre unter <https://gitlab-ce.rrz.uni-hamburg.de/>. **Anmelden** können sich Personen mit ihrer B-Kennung & Passwort, externe Personen melden sich über den Reiter „Standard“ mit einer lokalen GitLab-Kennung an.

Zu beachten sind die [Nutzungsbedingungen](#).

GitLab ist vielseitig nutzbar: Es beinhaltet die Möglichkeit der Versionsverwaltung, hat darüber hinaus aber auch Funktionalitäten wie ein integriertes Wiki oder Issue-Tracking.

Subversion kann über den Browser angesprochen werden, aber die eigentliche Nutzung erfolgt über einen installierten Client. Der Fokus von Subversion liegt in der Versionierung von Dateien und Dokumenten.

Voraussetzung für die Nutzung ist eine gültige B-Kennung, mithilfe derer auch die erstmalige Einrichtung eines Subversion-Repositories [beantragt werden kann](#).

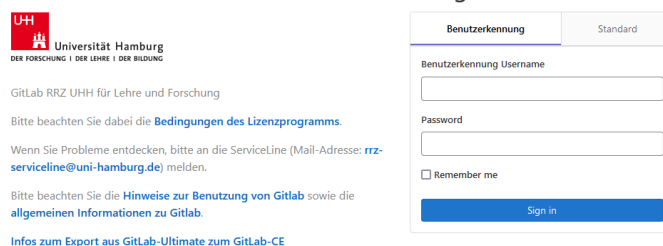
Zu beachten sind die [Nutzungsbedingungen](#).

2. GitLab

Im Folgenden soll die Nutzung von GitLab für Inhalte aus Forschung und Lehre dargestellt werden.

- Nach dem Aufrufen der Webseite loggen Sie sich mit Ihrer B-Kennung & Passwort an.

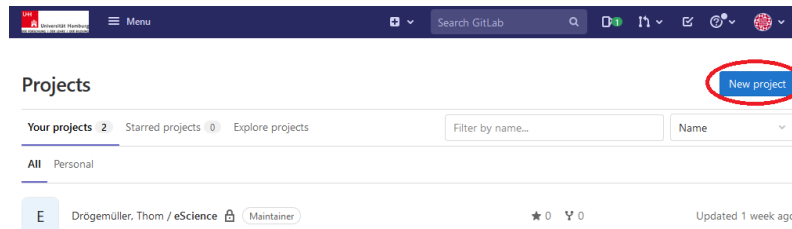
GitLab RRZ UHH für Lehre und Forschung



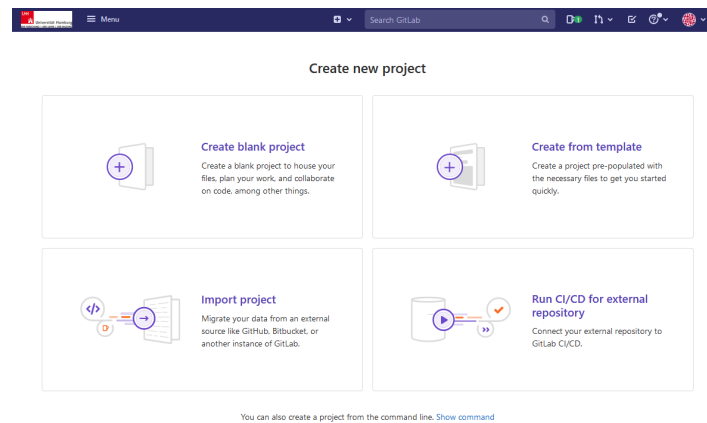
- Standardmäßig landen Sie auf der „Projects“-Seite. Wenn Sie bereits aktive Projekte haben, werden diese dargestellt. Sie können auch ein neues Projekt erstellen. Die ersten Schritte sind [vom Rechenzentrum beschrieben](#).

2.1 Einrichtung eines neuen Projekts

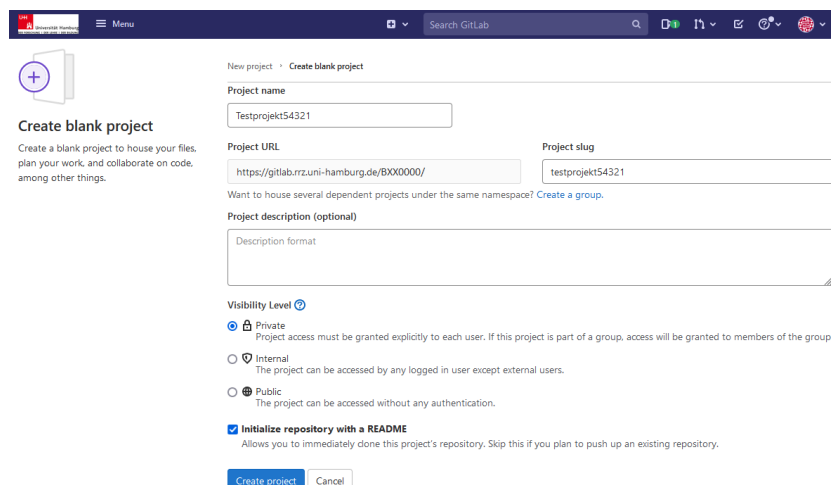
- Zum Erstellen eines neuen GitLab „Projects“ klicken Sie oben rechts auf „New Project“.



- Sie können unter anderem auswählen, ob Sie Ihr neues Projekt mit vorgefertigten Informationen oder als leeres Projekt erstellen wollen. Hier soll die Nutzung mit einem leeren Projekt aufgezeigt werden. Dazu klicken Sie auf „Create blank project“.



- Im nächsten Schritt können Sie Ihr Projekt benennen und beschreiben sowie die Sichtbarkeit einstellen. Zum Erstellen drücken Sie den Button „Create project“.
 - Der Projektname, den Sie auswählen, wird automatisch auch zum „Project slug“, Sie können den slug aber anpassen. Durch diese Einstellungen entsteht der Link zum Projekt, der „slug“ entspricht dabei dem letzten Teil der URL (z.B. <https://gitlab.rrz.uni-hamburg.de/BXX0000/testprojekt54321>).
 - Damit Ihr Projekt nur für diejenigen Personen sichtbar ist, die Sie ausdrücklich eingeladen haben, müssen Sie „Private“ auswählen.
 - Sie können zudem entscheiden, ob Sie das Repository des Projekts mit einer Readme-Datei ausstatten möchten, diese können Sie aber auch später hinzufügen.



- Sie landen nun auf der Übersichtsseite für Ihr soeben erstelltes Projekt. Auf der linken Seite befinden sich eine Reihe an Links.

The screenshot shows the GitLab interface for a new project named 'Testprojekt54321'. On the left, there is a sidebar with navigation links: Project information, Repository, Issues, Merge requests, Requirements, CI/CD, Security & Compliance, Deployments, Monitor, Infrastructure, Packages & Registries, Analytics, Wiki, Snippets, and Settings. The main content area displays the project name, ID (2823), and statistics (1 Commit, 1 Branch, 0 Tags, 143 KB Files, 143 KB Storage). Below this, there are buttons for 'History', 'Find file', 'Web IDE', and 'Clone'. An 'Initial commit' is shown, authored 8 minutes ago. A table lists the files in the repository:

Name	Last commit	Last update
→ README.md	Initial commit	8 minutes ago

The content of the README.md file is displayed below the table: 'Testprojekt54321'.

2.2 Nutzung von GitLab

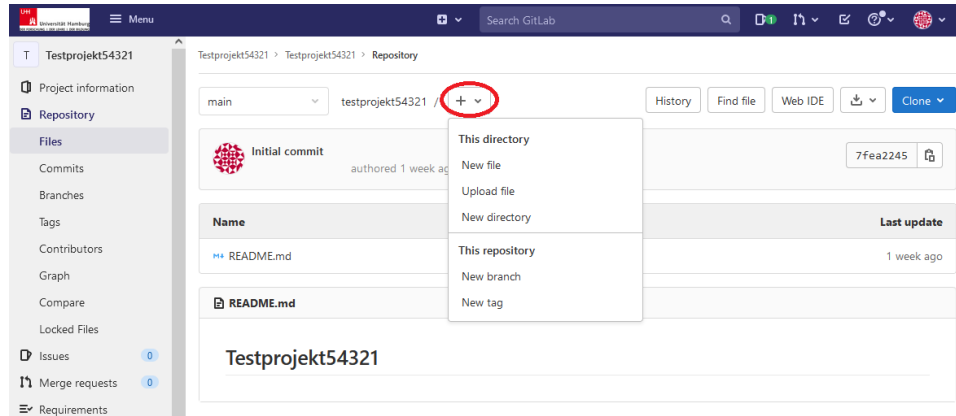
- Die Möglichkeiten innerhalb von GitLab übersteigen den Bedarf einer simplen Versionierung. An dieser Stelle wird neben den Informationen zur Versionsverwaltung sowie einiger Grundeinstellungen noch kurz auf die „Issues“ und das Wiki eingegangen. Für alle weiteren Möglichkeiten können Sie bspw. auf [die offizielle GitLab Dokumentation](#) zurückgreifen (Englisch).
- Relevante Links:
 - „Project information“ und insbesondere der Unterpunkt „Members“: Hier können Sie dem Projekt andere Personen hinzufügen.
 - Sie können Personen via Namen oder B-Kennung finden, Ihnen eine Rolle zuweisen und sie über den „Invite“-Button hinzufügen.

The screenshot shows the 'Project members' page in GitLab. The page title is 'Project members' and it includes the text: 'You can invite a new member to Testprojekt54321 or invite another group.' There are two tabs: 'Invite member' (selected) and 'Invite group'. The 'Invite member' form contains the following fields:

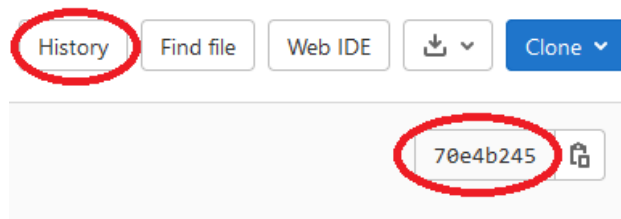
- GitLab member or Email address:** A text input field.
- Select a role:** A dropdown menu with 'Guest' selected. A link 'Learn more about roles.' is provided below.
- Access expiration date:** A date input field.

At the bottom of the form are 'Invite' and 'Import' buttons. Below the form, there is a 'Members' section with a count of 1 member and a search bar labeled 'Filter members'.

- „Repository“:
Unter dem standardmäßig ausgewählten Unterpunkt „Files“ finden Sie eine Übersicht aller dem Projekt hinzugefügten Dateien, zudem können Sie neue Dateien hinzufügen.
Dies ist der wichtigste Ort zwecks der Versionskontrolle. Sie können hochgeladene Dateien auswählen, herunterladen, durch andere Dateien ersetzen, für weitere Benutzer sperren und mehr.



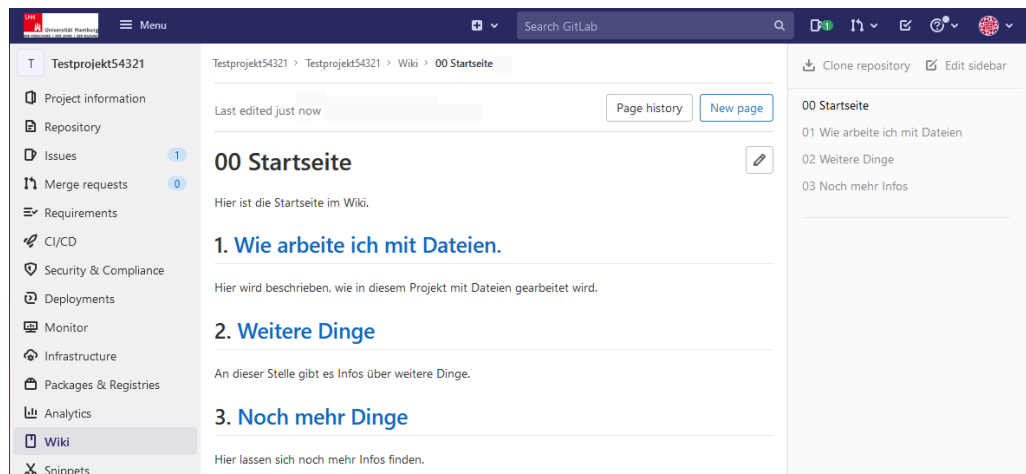
- Wenn Sie auf das Plus-Symbol klicken, können Sie hier Dateien hochladen sowie Ordner erstellen.
- Zudem erlaubt GitLab es Ihnen, „branches“ (Abzweigungen) innerhalb des Projekts zu erstellen. Dies erzeugt ein Abbild des aktuellen Projekts, an dem Änderungen vorgenommen werden können, ohne das Hauptprojekt zu beeinflussen. Zu einem späteren Zeitpunkt kann man einen so erstellten branch über ein „merge request“ zusammenführen. Die Änderungen werden dann im Hauptprojekt übernommen.
- **Besonders hervorzuheben ist dahingehend der „History“-Button.** Über diesen lässt sich die Versionsgeschichte einsehen. Jedes Hinzufügen, Bearbeiten oder Entfernen wird als „commit“ interpretiert, mit einer individuellen Buchstaben- und Zahlenkombination versehen und kann auf diese Weise eindeutig identifiziert werden. So lässt sich eine frühere Dateiversion auch später wieder auffinden.



Weitere Infos zu Unterpunkten innerhalb von „Repository“:

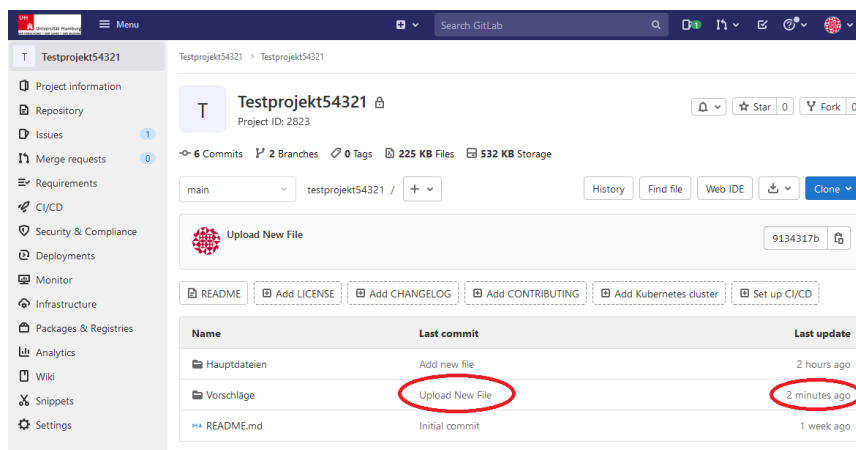
- Commits = Hier werden alle Änderungen in zeitlicher Reihenfolge angezeigt.
- Branches = Hier werden alle vorhandenen Abzweigungen angezeigt
- Locked Files = Hier werden alle Dateien angezeigt, die aktuell für die Bearbeitung gesperrt sind.
- „Issues“:
Hier können Sie Aufgaben verwalten und einzelnen Personen innerhalb des Projekts zuweisen, um einen Überblick zu behalten (ähnlich zu Trello).
 - Standardmäßig sind die „Boards“ *Open* und *Closed* angelegt.

- An der Universität Hamburg wird alternativ für diesen Zweck der Dienst „[Wekan](#)“ angeboten.
- „Wiki“:
Hier können Sie Informationen im Wiki-Stil hinterlegen – wie zum Beispiel projektbezogene Instruktionen zu bestimmten Aufgabenbereichen. Nach der Erstellung einer ersten Seite wird diese (und alle weiteren später erstellten Seiten) im rechten Bereich dargestellt. Genaue Informationen zur Befüllung eines Wikis [lassen sich hier nachlesen](#) (Englisch).

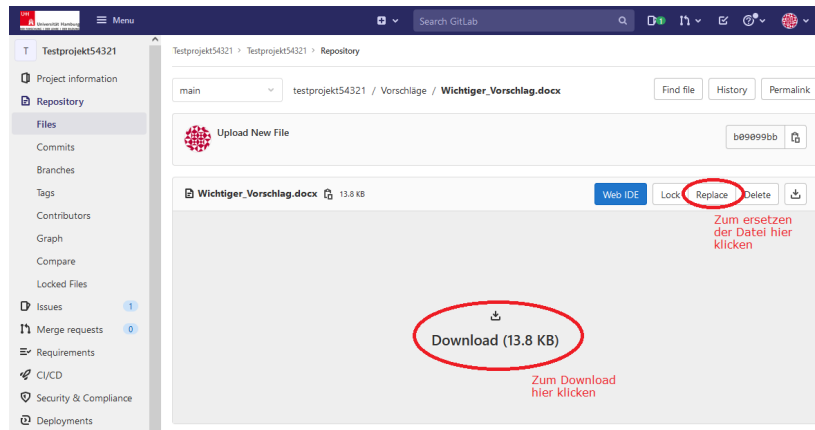


2.3 Ein Beispielszenario

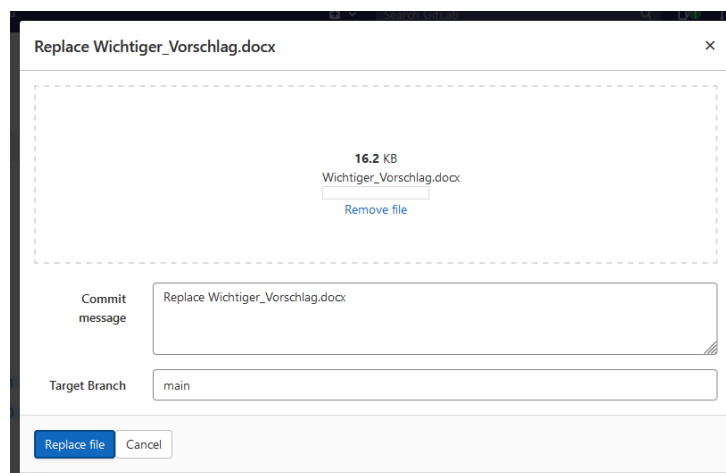
In dem Beispielszenario hat eine Person dem Projekt eine neue Datei hinzugefügt. Nach dem Einloggen ist zu erkennen, dass in dem Ordner „Vorschläge“ vor kurzer Zeit eine Änderung vorgenommen wurde.



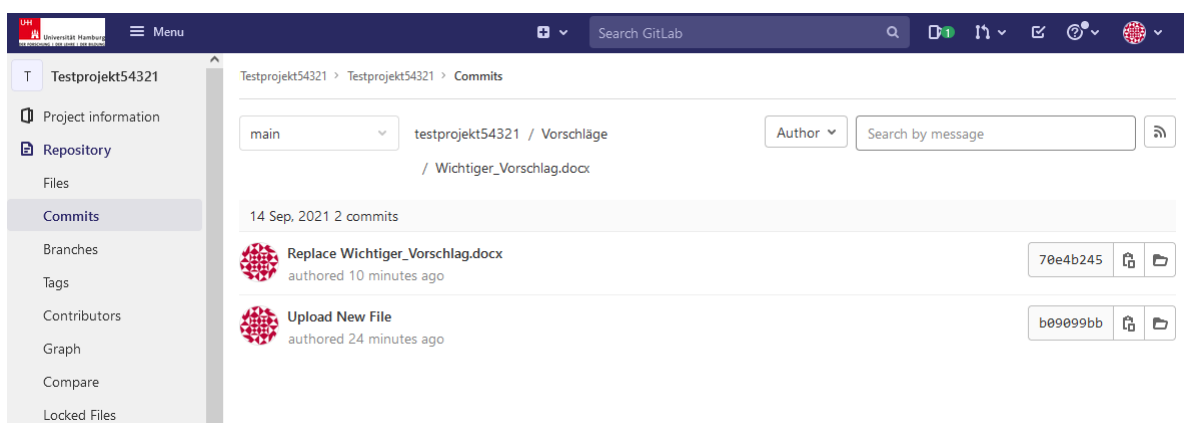
Sie können nun den Ordner öffnen und die entsprechende(n) Datei(en) herunterladen. Nach dem Download bearbeiten Sie die Datei und möchten anschließend das aktualisierte Dokument wieder hochladen.



Dazu können Sie auf den Button „Replace“ klicken. In dem sich öffnenden Fenster fügen Sie das bearbeitete Dokument hoch und klicken „Replace File“.



Betrachten Sie nun die „History“ der einzelnen Dateien, können Sie sowohl den ursprünglichen Upload sowie die vorgenommene Änderung nachvollziehen.



3. Subversion

Das [RRZ beschreibt](#), dass das [beantragte](#) Subversion-Repository nach Einrichtung über folgenden Server erreicht werden kann:

<https://svn.rrz.uni-hamburg.de/repositoryname>

Das bedeutet einerseits, dass sich über den Link der letzte Stand des Repositories über einen Browser aufrufen lässt (Zugang mit dem Repository zugehöriger B-Kennung). Andererseits wird der Link benötigt, um in einem der Clients Zugriff auf das Repository zu erhalten.

3.1 Verwaltung der Repository-Informationen

Über eine bereitgestellte [Webschnittstelle](#) können aus dem Universitätsnetz (also vor Ort oder via VPN) mit der dazugehörigen B-Kennung die Informationen zum Repository angepasst sowie weitere Personen hinzugefügt werden.

- Zum Hinzufügen zusätzlicher Kennungen geben Sie unter der Option „Zugriffsdaten zu folgendem Repository ändern“ den Namen Ihres Repositories an.



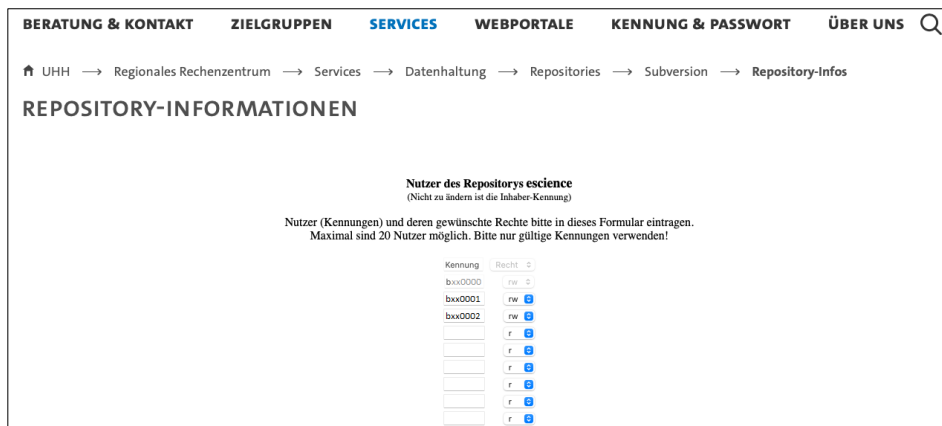
UHH → Regionales Rechenzentrum → Services → Datenhaltung → Repositories → Subversion → Repository-Infos

REPOSITORY-INFORMATIONEN

Mit den folgenden Formularen können Sie Informationen zu Ihren Subversions-Repositories abfragen, die Gültigkeitsdauer verlängern und die Zugriffsrechte ändern (nur aus dem Uni-Netz). Es wird jeweils nach Ihrer Benutzerkennung (b.....) und dem dazugehörigen Paßwort gefragt.

Liste aller Subversion-Repositories zur Kennung erstellen: (mit Gültigkeitsdatum, ggf. mit Verlängerungsmöglichkeit)	ausführen
Informationen zum angegebenen Subversion-Repository erstellen:	ausführen
Zugriffsdaten zu folgendem Repository ändern:	ausführen

- Anschließend können Sie dem Repository bis zu 20 Kennungen hinzufügen.



BERATUNG & KONTAKT ZIELGRUPPEN SERVICES WEBPORTALE KENNUNG & PASSWORT ÜBER UNS

UHH → Regionales Rechenzentrum → Services → Datenhaltung → Repositories → Subversion → Repository-Infos

REPOSITORY-INFORMATIONEN

Nutzer des Repositories escience
(Nicht zu ändern ist die Inhaber-Kennung)

Nutzer (Kennungen) und deren gewünschte Rechte bitte in dieses Formular eintragen.
Maximal sind 20 Nutzer möglich. Bitte nur gültige Kennungen verwenden!

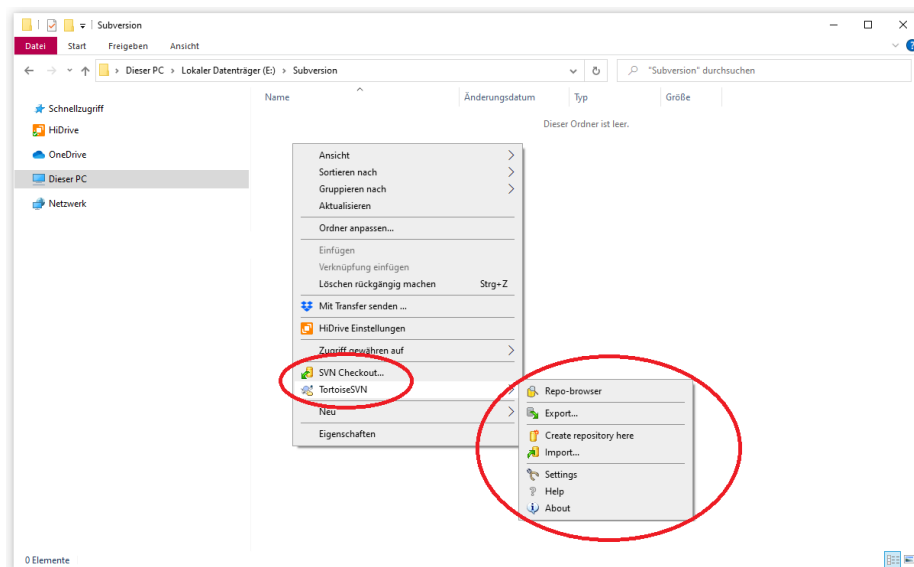
Kennung	Recht
bxx0000	rw
bxx0001	rw
bxx0002	rw
	r
	r
	r
	r
	r
	r

3.2 Graphische Clients für Subversion

Das [RRZ verlinkt](#) zur Subversion-Website und beschreibt kurz die Nutzung mit der Kommandozeile. Für die Nutzung einer graphischen Oberfläche ist der Download eines Clients notwendig. Dabei gibt es unterschiedliche Angebote für [MacOS](#), [Windows](#) und [Linux](#). Einige der Clients benötigen eine Registrierung, andere sind direkt nutzbar. Beispielhaft soll hier die Nutzung mit dem Windows-Client [TortoiseSVN](#) dargestellt werden. Dieser Client fügt dem Kontextmenü in Windows zusätzliche Optionen hinzu.

3.2.1 Download & Installation von TortoiseSVN

- Auf der [TortoiseSVN-Website](#) wird die aktuellste Version für ein 64-Bit System heruntergeladen. (Optional lassen sich hier auch Sprachpakete herunterladen.)
- Zur Installation des Clients folgen Sie den Installationsanweisungen.
- Nach der erfolgreichen Installation sind bei einem Rechtsklick (z.B. im Explorer) zusätzliche Kontextoptionen dem Menü hinzugefügt:
 - SVN Checkout
 - TortoiseSVN
 - Repo-Browser
 - Export
 - Create repository here
 - Import
 - Settings
 - Help
 - About

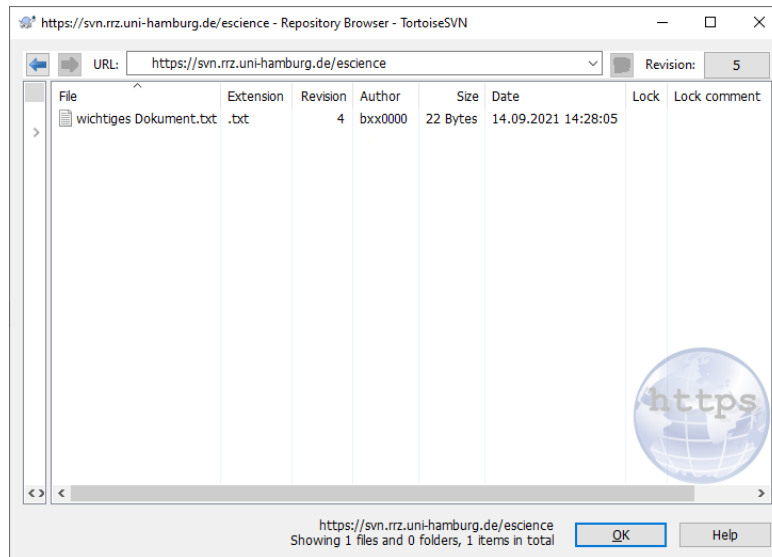


3.2.2 Nutzung von TortoiseSVN

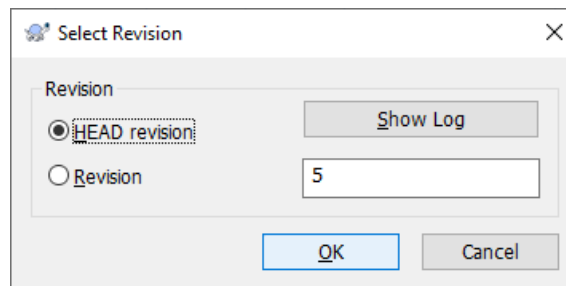
Es werden zwei Wege zur Verwaltung eines Repositories mit TortoiseSVN dargestellt.

Der erste Weg zur Nutzung von TortoiseSVN ist: Rufen Sie an einem beliebigen Ort mit einem Rechtsklick das Kontextmenü auf und wählen Sie dann TortoiseSVN > **Repo-browser** aus.

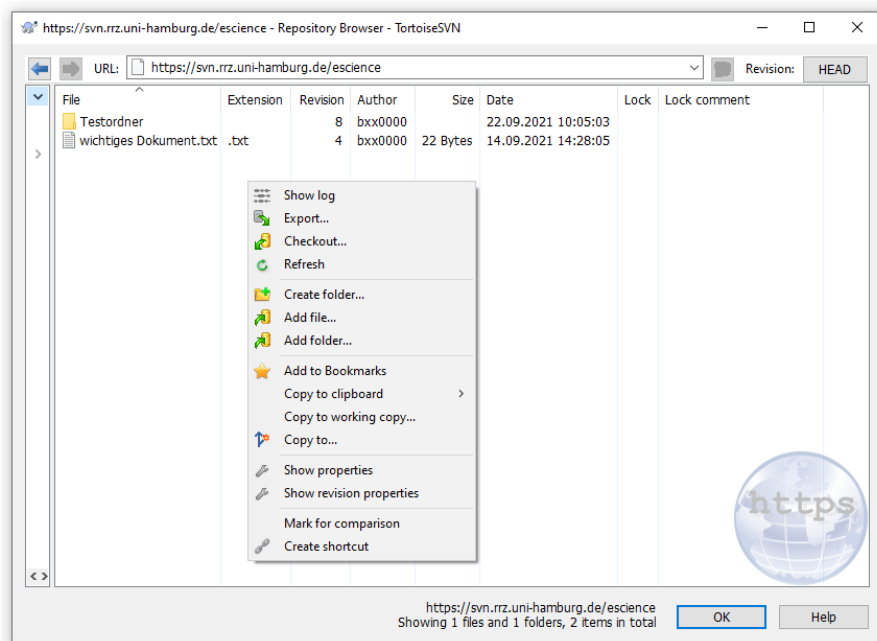
- Es öffnet sich ein Fenster, das nach einer URL fragt. Hier geben Sie die URL des Repositories an (<https://svn.rz.uni-hamburg.de/repositoryname>)
- Anschließend öffnet sich ein Fenster, in dem die Anmeldeinformationen abgefragt werden. Nach Eingabe der B-Kennung + Passwort bestätigen Sie mit OK.
- Es öffnet sich der „**Repository Browser**“, der Zugriff auf alle Dateien im Repository ermöglicht.



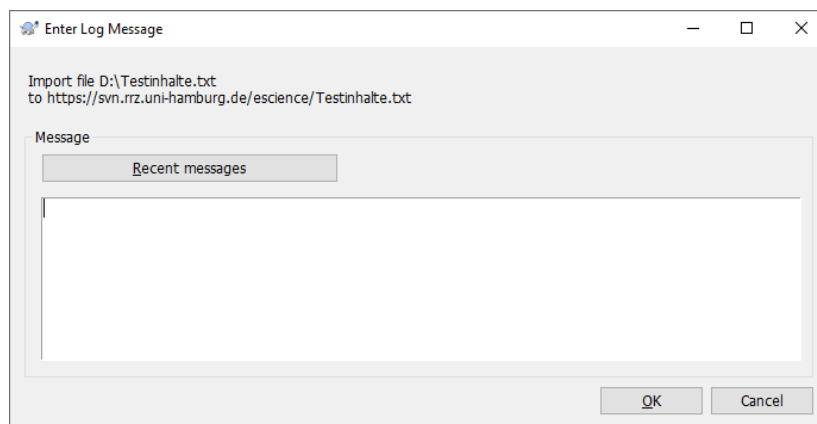
- Oben rechts wird die aktuelle „Revision“ angezeigt und ausgewählt, die neueste Version hat den Titel „HEAD“, ältere Revisionen lassen sich über eine numerische Angabe auswählen. Im Beispielbild ist die Revision 5 ausgewählt.



- Was bedeutet das? In aller Kürze: Es lassen sich alle vorherigen Versionen der Dateien anzeigen. Wenn Sie beispielsweise eine Datei löschen, können Sie durch Aufrufen von einer vorherigen Revisionsnummer weiterhin auf den letzten Stand der Datei zugreifen.
- Um innerhalb des Repository Browsers dem Repository Dateien hinzuzufügen, können Sie, sofern die HEAD Revision ausgewählt ist, entweder mit drag&drop Dokumente in dem Fenster ablegen oder durch einen Rechtsklick ein Kontextmenü aufrufen.
 - Innerhalb des Kontextmenüs haben Sie mehrere Optionen: Zum Befüllen können Sie mit "Create folder" einen neuen Ordner erstellen, mit „Add file“ eine einzelne Datei auswählen oder mit „Add folder“ einen gesamten Ordner samt Inhalt dem Repository hinzufügen.

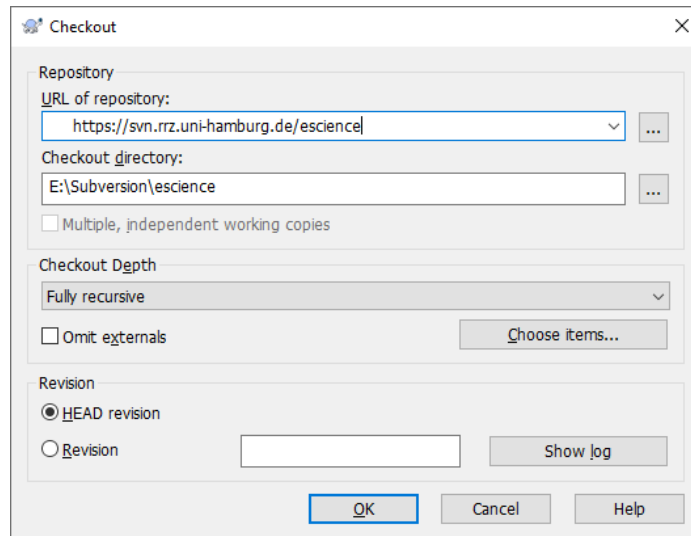


- Immer wenn Sie dem Repository neue Dateien hinzufügen, wird die Revisionsversion um 1 erhöht und sie werden aufgefordert, die Änderung zu kommentieren.
 - Tipp: Eine strukturierte Kommentar-Kultur vereinfacht späteres Nachvollziehen vorgenommener Änderungen. Sie können Dateien auch unkommentiert verändern.

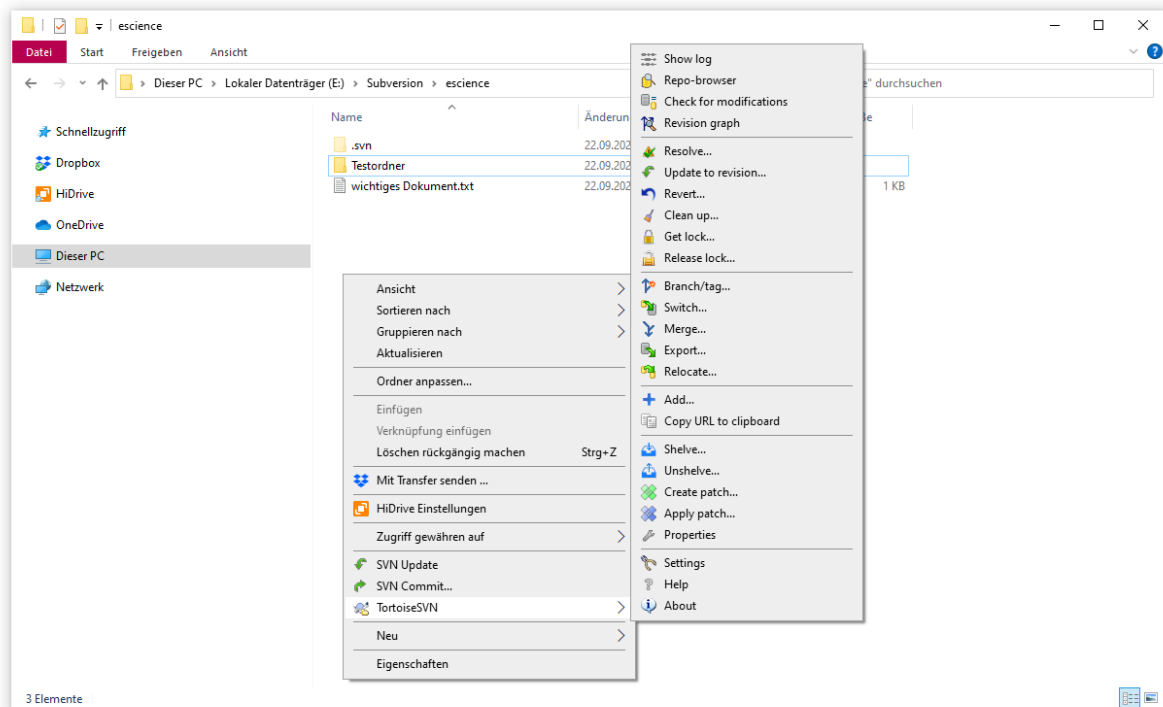


Der zweite Weg zur Nutzung von TortoiseSVN ist: Rufen Sie in einem leeren Ordner mit einem Rechtsklick das Kontextmenü auf und wählen Sie dann **SVNCheckout** aus, um eine lokale Kopie des Repositories auf dem PC zu erstellen.

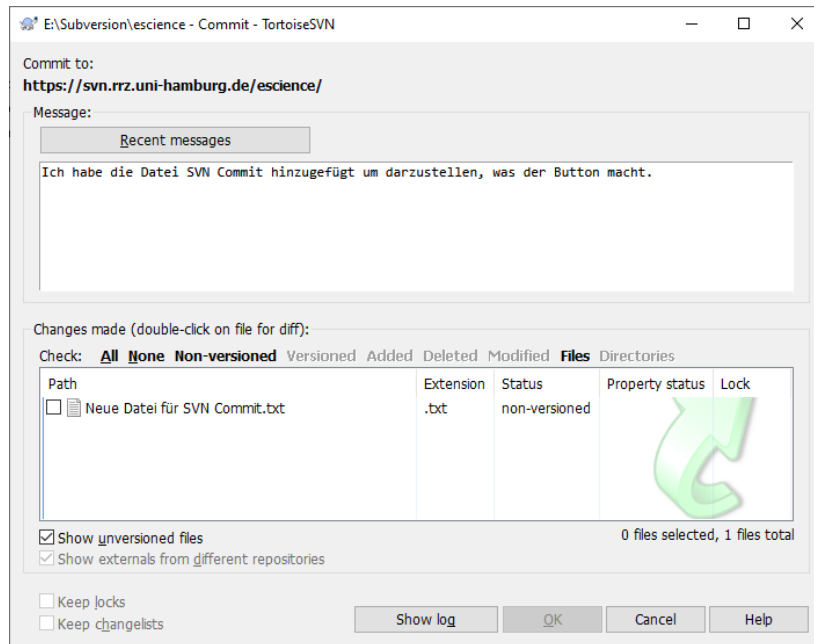
- Es öffnet sich ein Fenster mit dem Titel „Checkout“, das nach einer URL fragt. Hier geben Sie die URL des Repositories an (<https://svn.rz.uni-hamburg.de/repositoryname>). Zudem können Sie über die Option „Checkout directory“ einen anderen Zielordner auswählen. Aber wenn Sie sich in einem leeren Ordner befinden, können Sie die Option so belassen. Bei der Option „Revision“ können Sie auswählen, ob eine ältere Fassung oder die aktuelle HEAD Fassung des Repositories dargestellt werden soll. Bestätigen Sie mit „OK“.



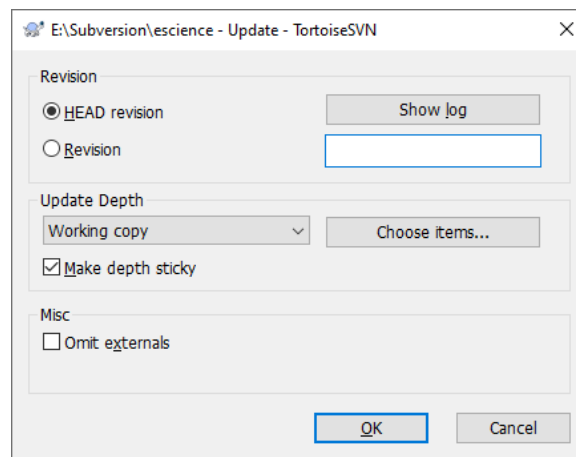
- Es öffnet sich ein kleines Fenster, das darstellt, welche Dateien heruntergeladen werden. Am angegebenen Ort befindet sich nun eine lokale Version aller Dateien und Ordner, die im Repository abgelegt waren. Zudem ist das Kontextmenü innerhalb des Ordners um weitere Funktionen ergänzt, von denen einige im Folgenden aufgezeigt werden.



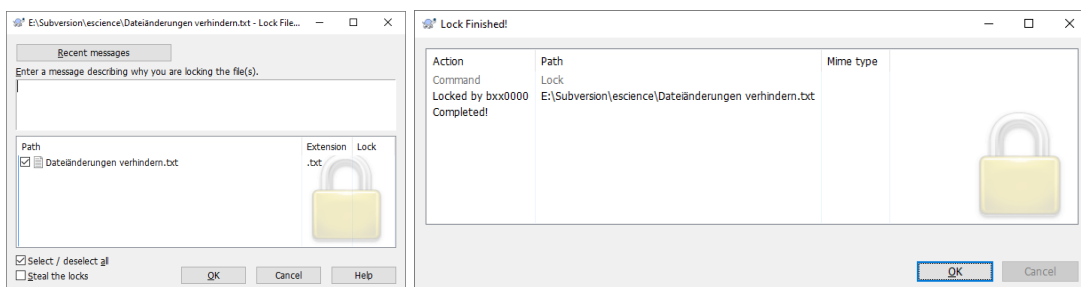
- Über „SVN Update“ lässt sich die lokale Version des Repositories updaten. Wenn in der Zwischenzeit beispielsweise andere Personen etwas geupdated haben, wäre man durch ein Klicken auf die Option wieder auf dem neuesten Stand.
- „SVN Commit“ ermöglicht, den aktuellen lokalen Stand des Repositories dem Server hinzuzufügen. Wenn Dateien hinzugefügt, geändert oder gelöscht wurden, kann man diese in einem sich öffnenden Fenster auswählen und (un)kommentiert hochladen.



- Die Kontextoption TortoiseSVN > „**Update to revision**“ ermöglicht, den Inhalt des lokal abgelegten Repositories durch eine andere Version zu ersetzen. Dazu kann man durch eine numerische Angabe die gewünschte Revision oder mit HEAD die aktuelle Version auswählen.



- Die Kontextoption TortoiseSVN > „**Get lock**“ ermöglicht Dateien zu sperren, sodass nur die Person, die die Datei „abgeschlossen“ hat, Änderungen an der Datei vornehmen kann, bis die Sperrung wieder aufgehoben wird.





- Allerdings ist ein Failsafe eingebaut, sodass andere Personen durch einen Klick auf „Steal the locks“ im Fall der Fälle die Dateien dennoch verändern können.

Für weitere Details können Informationen auf der TortoiseSVN-Webseite [auf Deutsch](#) oder [Englisch](#) nachgelesen werden.