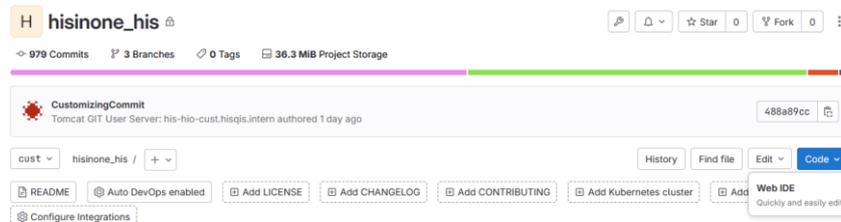


# Anleitung Gitlab

## Nutzung Gitlab WEB IDE

Wir empfehlen für die Bearbeitung von Dateien den integrierten Web IDE. Dieser kann innerhalb des Projektes über "Edit" --> "Web IDE" aufgerufen werden



## Vergleich von Dateien

Für den Vergleich von Dateien können zwei Dateien markiert werden: Linksklick auf Datei 1 --> STRG gedrückt halten --> Linksklick auf Datei 2 --> Rechtsklick --> Compare Selected



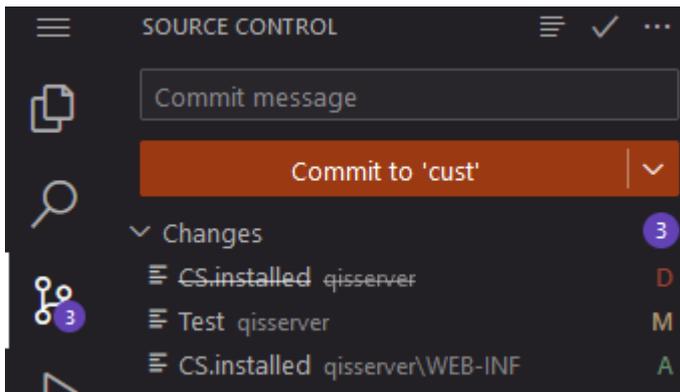
Die Dateien öffnen sich nebeneinander und die Differenzen werden entsprechend ausgewiesen

## Bearbeitung und Anlegen von Dateien und Ordnern

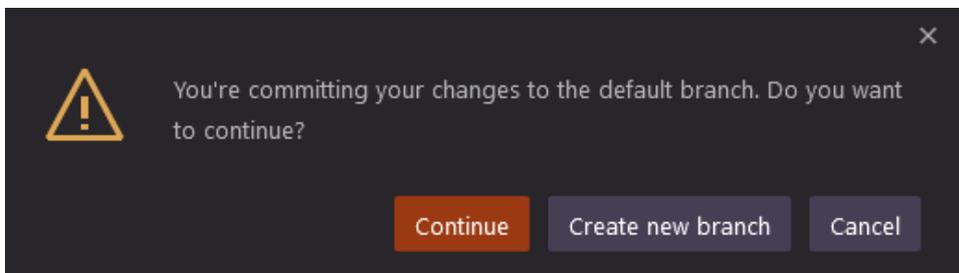
Bei Klick auf eine Datei, öffnet sich diese automatisch im Editor und kann dort bearbeitet werden. Neue Dateien oder Ordner können per Rechtsklick angelegt werden. Beim Anlegen von Ordnern ist zu berücksichtigen, dass ein leerer Ordner nicht ins Git gepusht werden kann. Soll ein Ordner persistent angelegt werden, muss hier mindestens eine Datei enthalten sein. Dies kann über eine leere Datei mit dem Namen ".gitkeep" realisiert werden. Das Hochladen von Dateien erfolgt über Rechtsklick --> Upload. Ein Upload von mehreren Dateien gleichzeitig ist darüber ebenso möglich (STRG gedrückt halten und Dateien auswählen). Die Löschung von Dateien erfolgt über Rechtsklick --> delete permanently. Das Verschieben von Dateien kann per Drag & Drop erfolgen. Dabei ist zu beachten, dass ein Verschieben immer bedeutet, dass die Datei im Quellordner gelöscht und im Zielordner neu angelegt wird.

## Pushen von Änderungen

Nachdem die Änderungen im Editor durchgeführt wurden, müssen diese über Source Control noch gepusht werden. Dafür links im Menü vom Editor in Source Control wechseln. Sie sehen nun eine Übersicht der Änderungen:



Anhand der Bezeichnung können Sie sehen, ob die Datei gelöscht (D), geändert (M) oder neu angelegt wurde (A). Nach Angabe einer Commit message, können Sie die Änderung über Commit to 'branch' pushen. Hier ist immer der Branch vorbelegt der vor dem Start des Web IDE im Projekt ausgewählt wurde. Sofern Sie Änderungen im CUST-Branch vornehmen, muss folgende Frage mit Continue bestätigt werden:



## Säulentransfer

Standardmäßig haben Sie nur Schreibrechte auf Customizing-Systeme. Für das Übertragen auf die Test- und Produktivsäule stellen Sie bitte ein HISZILLA-Ticket. Gerne kann auch ein Sammelticket geöffnet werden, welches immer wieder geöffnet und vom HIS-SaaS-Support nach Bearbeitung geschlossen wird. Bei HIS-SaaS wird das Konzept von Merge Requests innerhalb von Git **nicht** genutzt. Aus dem Ticket muss hervorgehen, welche Dateien übertragen werden sollen (gibt es keine Angabe zum Zielsystem, wird nur auf TEST übertragen). Falls Dateien direkt auf TEST und PROD übertragen werden sollen, muss das explizit vermerkt werden. Folgendes kann beauftragt werden:

- Transfer von Einzeldateien unter namentlicher Nennung (inkl. Voller Pfadangabe)
- Transfer der gesamten Konfigurationsdateien
- Übertragung von Commits unter Angabe der Commit-Nr.

Update 3 files ...  
0 Dennis Trautmann authored 1 hour ago

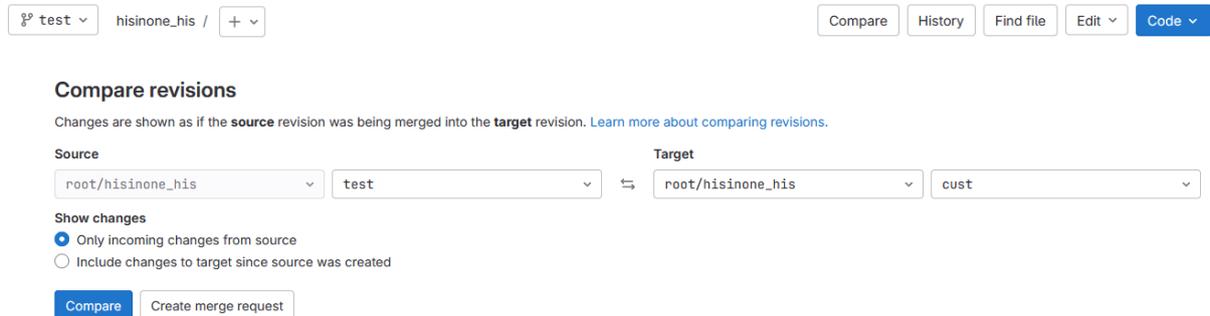
b6a09854

Copy commit SHA

## Tipps und Tricks

1. Sie möchten wissen, welche Dateien sich zwischen den Branches unterscheiden?

Wählen Sie dafür im Projekt einen Branch aus (**nicht** cust, da dies der Default-Branch ist) und wählen dann compare. Im Anschluss den Branch auswählen, der verglichen werden soll.



The screenshot shows the GitLab 'Compare revisions' interface. At the top, there are navigation elements: a search icon, a dropdown menu with 'test', the repository name 'hisinone\_his', and a '+' icon. To the right are buttons for 'Compare', 'History', 'Find file', 'Edit', and 'Code'. Below this is the 'Compare revisions' section, which includes a sub-header and a link to 'Learn more about comparing revisions'. The 'Source' section has a dropdown menu with 'root/hisinone\_his' and a dropdown menu with 'test'. The 'Target' section has a dropdown menu with 'root/hisinone\_his' and a dropdown menu with 'cust'. Below the source and target sections is the 'Show changes' section, which has two radio buttons: 'Only incoming changes from source' (selected) and 'Include changes to target since source was created'. At the bottom of the 'Show changes' section are two buttons: 'Compare' and 'Create merge request'.

Der Vergleich liefert zunächst die Commits on Source. Diese können Sie überspringen, da bei HIS-SaaS keine Merge Requests genutzt werden. Danach folgt die Auflistung aller Dateien, die sich unterscheiden.



The screenshot shows the GitLab diff view. At the top, there is a status bar: 'Showing 18 changed files' with a dropdown arrow, followed by 'with 3418 additions and 24 deletions'. To the right are buttons for 'Expand all', 'Hide whitespace changes', 'Inline', and 'Side-by-side'. Below this is a list of files. The first file is '.gitignore' with a 'View file @ 1c7d4b2a' button. Below the file name is the text 'This diff is collapsed. Click to expand it.' The second file is 'README' with a 'View file @ 1c7d4b2a' button. Below the file name is the text 'This diff is collapsed. Click to expand it.'

## Eclipse - Repository über https einbinden

das Einbinden vom Gitlab-Repository ist bei HIS-SaaS von außen nur über https möglich. Hierfür benötigen Sie einen Personal Access Token. Gehen Sie wie folgt vor:

- Klicken Sie links oben auf Ihr Profil und wählen "Edit Profile"
- wählen Sie links "Access Tokens"
- Erstellen Sie einen neuen Token über "add new token"
- vergeben Sie einen Token name zB eclipse
- setzen Sie ein Ablaufdatum
- als scopes können Sie read\_repository und write\_repository auswählen
- mit create personal access token können Sie den Token erzeugen
- Wichtig: Speichern Sie diesen Token sicher! Nach Abschluss kann man sich den Key nicht noch einmal anschauen.

Nun können Sie über die Oberfläche im Projekt über Code--> Clone with HTTPS sich die benötigten Infos in den Cache holen.

**Add Task Repository...**

**GitHub Issues**

Success!

Server:

Label:   Disconnected

User ID:

Token:   Save Token

Use Access Token Authentication

▶ Additional Settings

▶ Proxy Server Configuration

Dann z.B. in Eclipse importieren. Es sollte alles vorausgefüllt sein. Das ggf. vorausgefüllte Passwort tauschen Sie gegen den Personal Access Token, dann sollte keine Fehlermeldung mehr kommen. Wenn Sie den Haken bei "Store in Secure Store" setzen, ist der Personal Access Token zukünftig auch als Passwort gespeichert.

#### Hinweis zu Urheber- und Nutzungsrechten

Dieses Dokument ist urheberrechtlich geschützt. Es darf ausschließlich innerhalb der juristischen Person, der es seitens der HIS Hochschul-Informationssystem eG zur Verfügung gestellt wurde, zu Informations-, Qualifizierungs- und Fortbildungszwecken verwendet und zu diesem Zweck in unveränderter und vollständiger Form vervielfältigt und innerhalb der juristischen Person weitergegeben werden.

Gleiches gilt für juristische Personen, die in einem Verbund zusammengeschlossen sind und deren koordinierende juristische Person sich zugleich mit der HIS eG in einem entsprechenden Rahmenvertragsverhältnis befindet. In diesem Fall darf das Dokument innerhalb dieses Zusammenschlusses unter Wahrung des Vorgenannten entsprechend weitergegeben werden.

Alle weiteren Rechte verbleiben bei der HIS eG.

Insbesondere ist jede – auch auszugsweise – Weitergabe an Adressaten außerhalb des o. g. Empfängerkreises, dem dieses Dokument seitens der HIS eG zur Verfügung gestellt wurde und jede Veröffentlichung außerhalb des o. g. Empfängerkreises, insbesondere im Internet, ohne explizite Zustimmung der HIS eG ausdrücklich untersagt.