Machine Learning Applications (MLA)



Instruction for Infrastructure Overview of Gitlab





General Information



What is Gitlab?

- Gitlab offers a GUI for the widely known code versioning software Git
- You can work together on code projects in Gitlab at the same time
- Code changes are versioned by commits inside of branches and can be merged together

How do I access the Gitlab?

Open the URL: https://mla.fsr.maschinenbau.tu-darmstadt.de



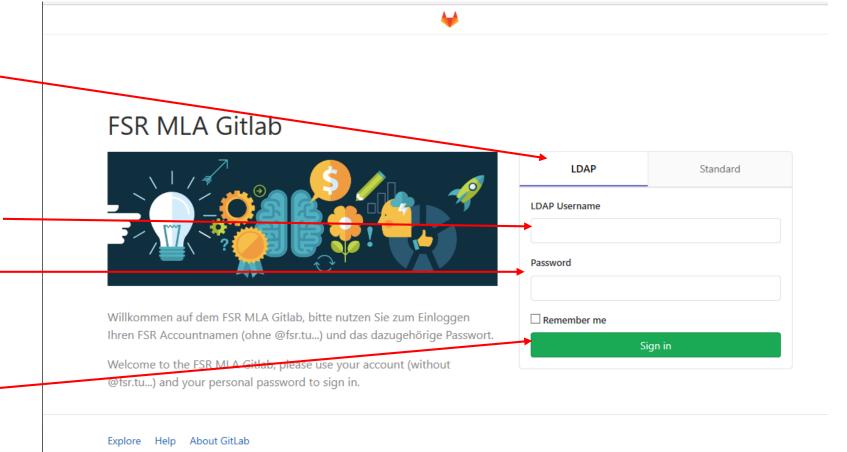
Login Page



Make sure LDAP is selected

Enter your usernameand password ———

Click Sign in





Create a Gitlab Group for Your Group (1)



- A Gitlab group is used to collect different projects together
- Each person of your group has to login to Gitlab once to be added to a group → you can not add a person that has never loged-in
- One person of your group has to create the group
- You should name the group according to your group
- → See next slide

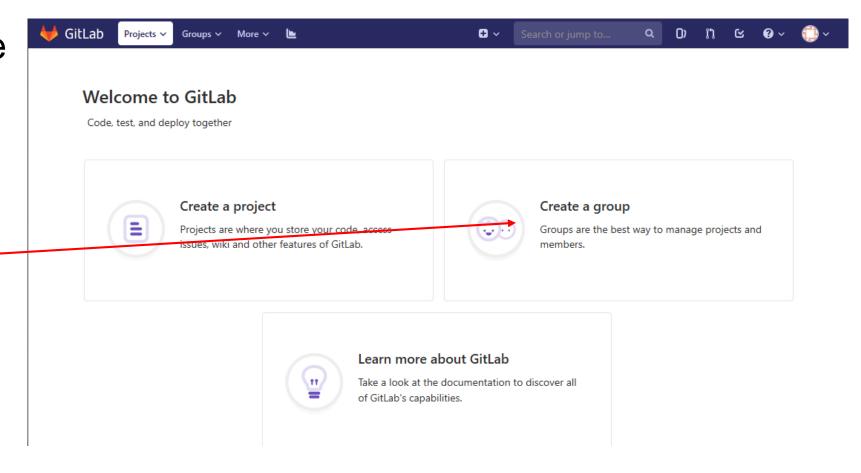


Create a Gitlab Group for Your Group (2)



 This is the start page after your first login

Click on Create a group





Create a Gitlab Group for Your Group (3)



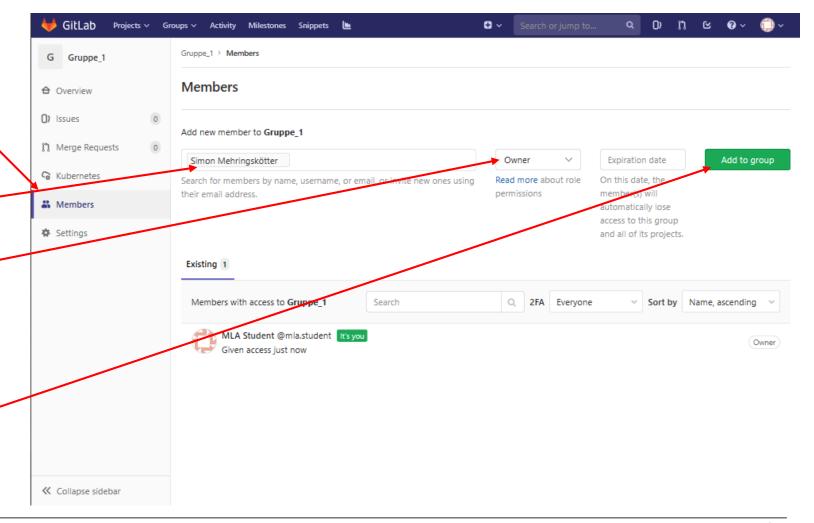
🤟 GitLab Projects 🗸 Groups 🗸 Activity Milestones Snippets 🖿 Enter your group name (e.g. New group Gruppe_1) Group name w you to manage and collaborate across multiple proje Gruppe_1 Members of a group have access to all of its projects. Group URL Do **not** modify this field! Groups can also be nested by creating https://mla.fsr.maschinenbau.tu-darmstadt.de/ gruppe 1 Projects that belong to a group are prefixed Group description (optional) with the group namespace. Existing projects This is the group for Gruppe 1 to collect all the code projects developed may be moved into a group. Enter a description of your during the MLA exam group Group avatar Choose file... | No file chosen The maximum file size allowed is 200KB. Upload an Avatar if you want Visibility level Who will be able to see this group? View the documentation The group and its projects can only be viewed by members. Leave visibility level at *Private*. The group and any internal projects can be viewed by any logged in user. → Click Create group Create group Cancel



Create a Gitlab Group for Your Group (4)



- Click on Members
- Type each name of your group members and select them
- Select Owner in the drop down
- → Click Add to group





Create a Gitlab Project in Your Group (1)



- Click on Groups and selectYour Groups
- Click on your group name (e.g. Gruppe_1)

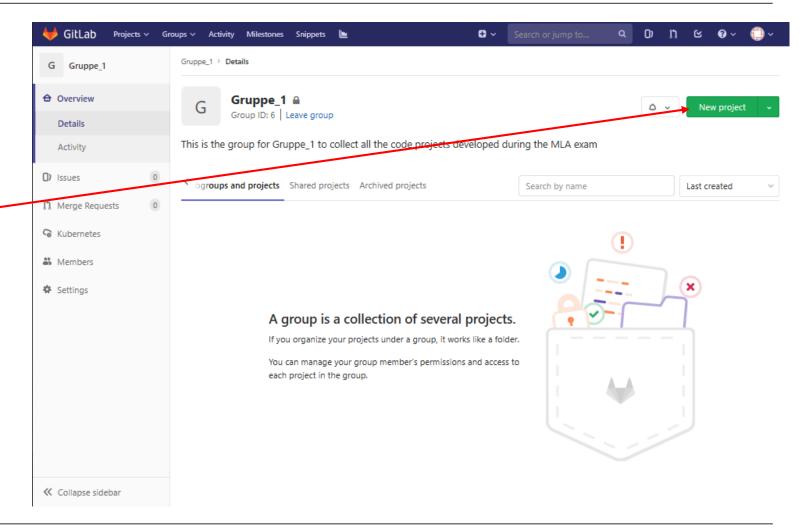




Create a Gitlab Project in Your Group (2)



- This page is the overview of your group and the corresponding projects
- Click on New project
- If your are not in the group overview, you will not create a project in your group and none of your group members will have access to the project!

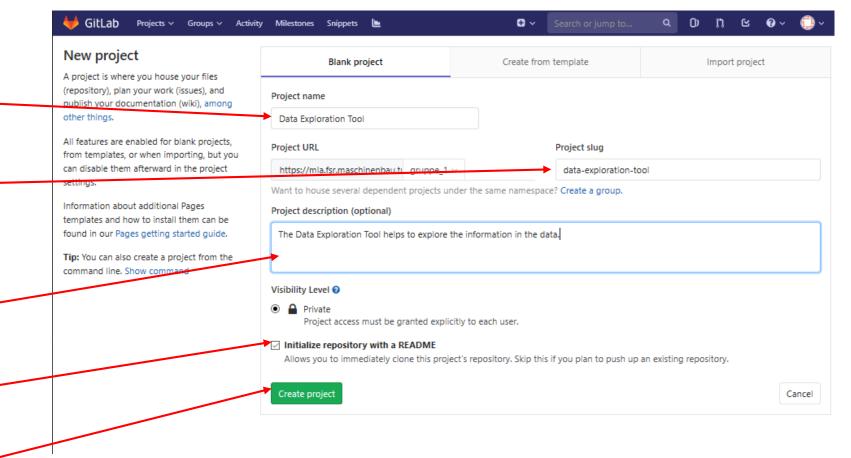




Create a Gitlab Project in Your Group (2)



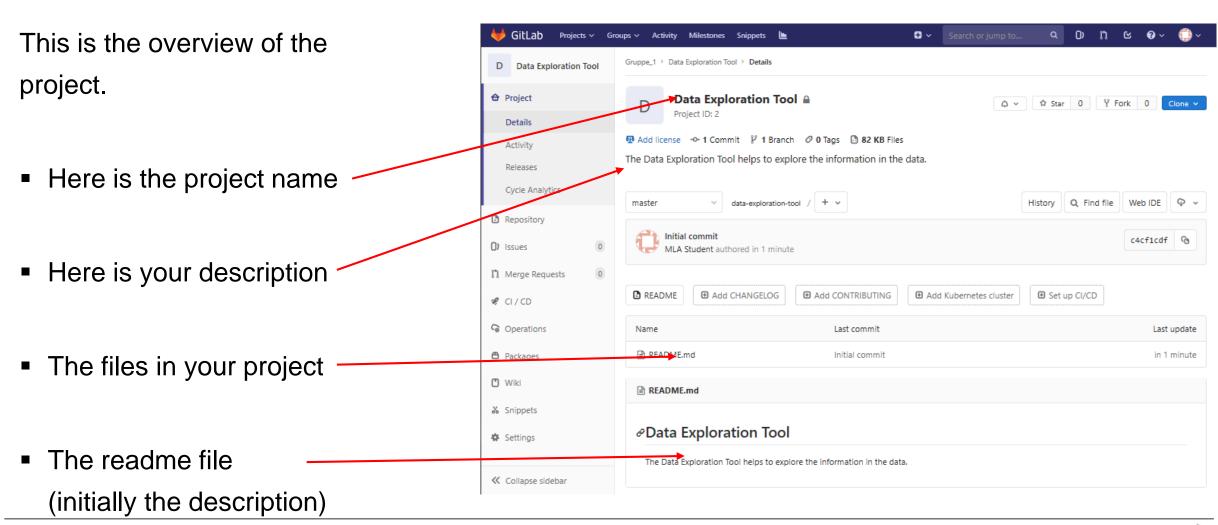
- Enter a project name, i.e. a name for the software you will develop in the project
- Do <u>not</u> modify this field!
- Enter a description that helps to understand what the software is good for
- Check this box
- → Click Create project





Create a Gitlab Project in Your Group (3)

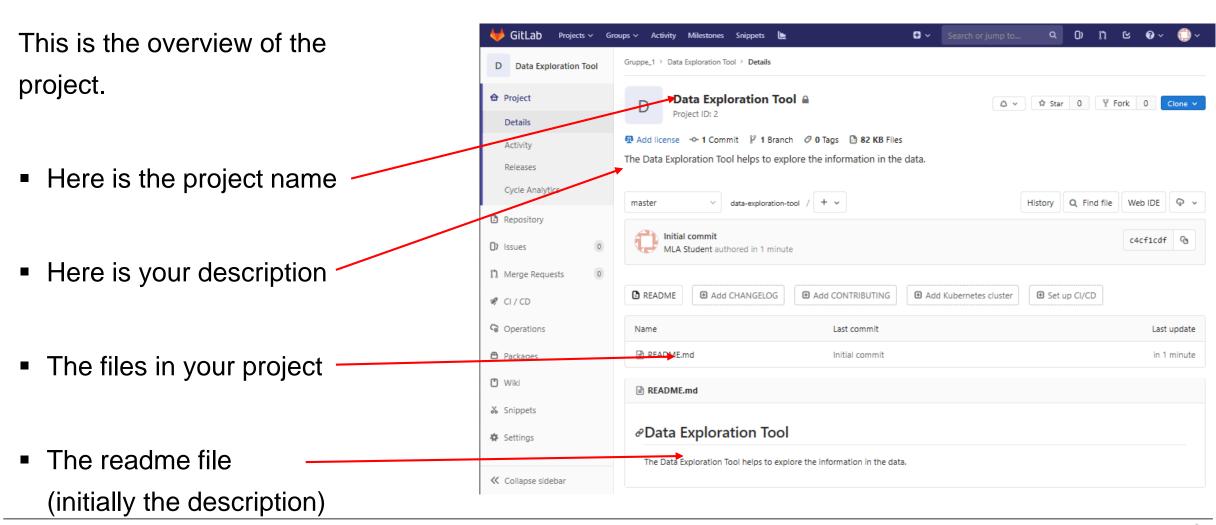






Create a Gitlab Project in Your Group (4)







Some Remarks on How to Use Gitlab



- We can not provide a full tutorial for Gitlab here
 Read through the internet there are lots of websites for further details
 (e.g. https://docs.gitlab.com/ee/gitlab-basics/)
- Always create a .gitignore file as a first step in your Gitlab project
 - gitignore lists file types that should be ignored because they do not really belong to code (e.g. temporary files)
 - https://www.gitignore.io/ is a great website to create a .gitignore file for the programming languages that are used in the project
- Always work with different branches you should have develop and master branches (a way to initially set up the branches is shown some slides later)
- Use the *Issue* function to document on issues or anything else in your code and assign members to work on the issues – this can really help to coordinate your work



Setup Git Software to Work on Your Computer (1)



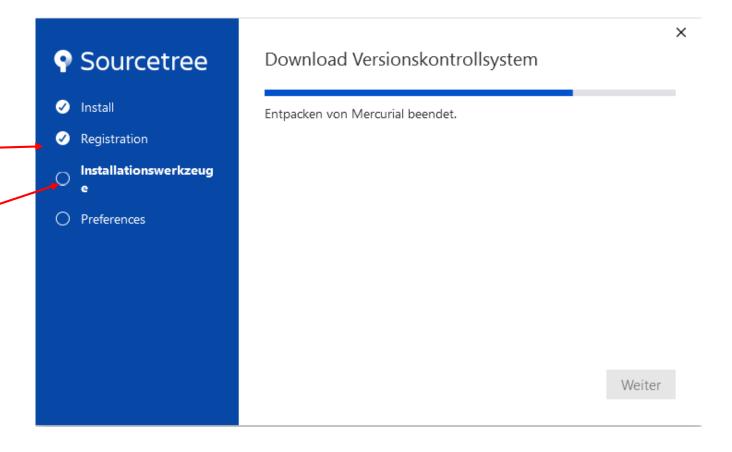
- Typically you don't want to upload the files via the browser
- Sourcetree is only an exemplary software that is shown here you can use any other software
 - Sourcetree is for free however, you need to create an Atlassian account
 - https://www.sourcetreeapp.com/
- The following slides show how to set up source tree using an SSH key to access your projects



Setup Git Software to Work on Your Computer (2)



- Download and install Sourcetree from the website
- Follow the steps of the wizard
 - Registration
 - You have to click Create one for free if you don't have an account
 - Click on Bitbucket and Login with your account
 - Installationswerkzeuge
 - Select Git and Mercurial
- After installation Sourcetree asks for a SSH-Key → Click on No





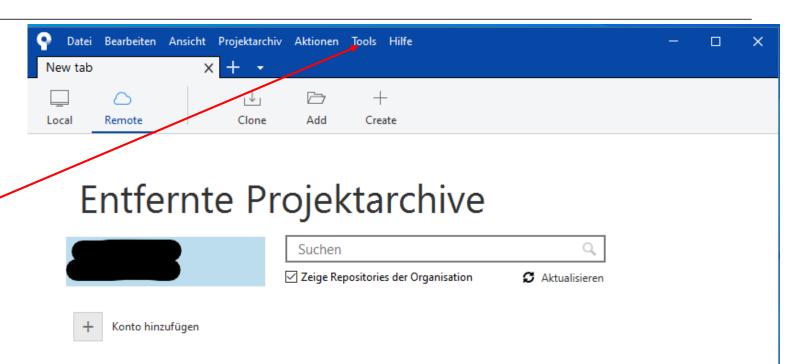
Setup Git Software to Work on Your Computer (3)

Accounts bearbeiten...



 This is the main window of Sourcetree

- First you should create a SSH-Key
- Click on Tools and then on Import or Create SSH-Key / SSH-Schlüssel importieren oder erstellen

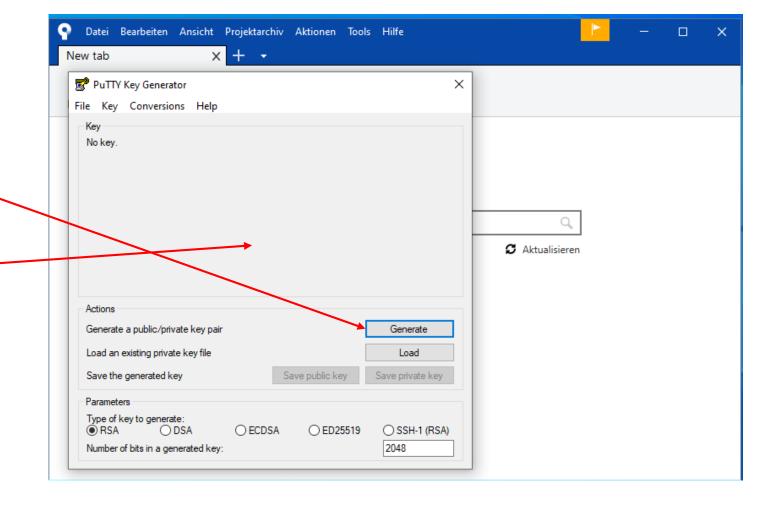




Setup Git Software to Work on Your Computer (4)



- The *PuTTY Key Generator* opens
- Click on Generate
- Move your mouse in the blank area until the bar is filled

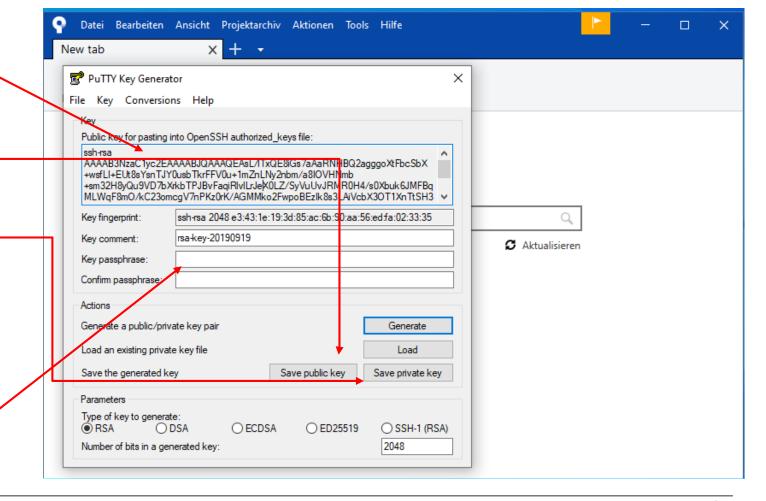




Setup Git Software to Work on Your Computer (5)



- Here is your Key
 Select everything and copy it to the clipboard
 we need it on the next slide
- Click on Save public key and save the file somewhere on your computer (usually you will not need this again)
- Click on Save private key and store the key on your computer (we will need this)
 - The software will ask you if you really don't want to use a password – you can confirm it
 - It's more secure to use a password because anyone with your key file could access your Gitlab projects. Define the password here
 - If you have password you need to enter it each time you start Sourcetree





Setup Git Software to Work on Your Computer (6)



Go to Gitlab webpage and click here <mark>
→ GitLab Projects ∨ Groups ∨ Activity Milestones Snippets </mark>
▶ a u n 🗷 🕡 🗸 choose Settings in the menu that opens User Settings User Settings > SSH Keys Add an SSH key SSH Keys Profile To add an SSH key you need to generate one or use an existing key. SSH keys allow you to establish a secure Click on SSH Keys connection between your computer and GitLab. Paste your public SSH key, which is usually contained in the file '~/.ssh/id ed25519.pub' or *** Applications '~/.ssh/id_rsa.pub' and begins with 'ssh-ed25519' or 'ssh-rsa'. Don't use your private SSH key. □ Chat Typically starts with "ssh-ed25519 ..." or "ssh-rsa ..." Access Tokens ☑ Emai A Notifications SSH Keys Paste the key from the clipboard into Title P GPG Keys this field e.g. My MacBook key 2 Preferences Name your individual key via a title Add key Click on Add key



Setup Git Software to Work on Your Computer (7)



Datei Bearbeiten Ansicht Projektarchiv Aktionen Tools Hilfe Go back to Sourcetree and click on *Tools* Optionen and select *Preferences / Optionen* Allgemein Updates Mercurial Eigene Aktionen Authentifizierung Sourcetree erlauben, Änderungen an der globalen Git- und Mercurial-Konfiguration durchzuführen Use this version of SourceTree for URI association Choose the private key file that you saved two In Zukunft "Lesezeichen erstellen" anbieten Standard Benutzer Informationen slides before Allständiger Name: Simon Mehringskötter E-Mail Adresse: | mehringskoetter@fsr.tu-darmstadt.de SSH-Verbindungskonfiguration SSH Schlüssel: Click on OK -SSH-Client: PuTTY / Plink V (Nur Git, Mercurial nutzt auf Windows immer Plink) ☑ Starte den SSH Agent automatisch wenn Sourcetree gestartet wird Repo Settings Projekt Ordner: Sprache: Automatisch (Erfordert Neustart) Helfen Sie dabei Sourcetree zu übers Standard Textcodierung | utf-8 🔻 ✓ Behalte Sicherungen bei gefährlichen Operationen Automatisch aktualisieren, wenn sich Dateien ändern Aktualisieren wenn die Anwendung nicht den Fokus hat Remotes auf Aktualisierungen überprüfen alle 10 Minuten Projektarchiv-Tabs beim Starten wiederherstellen ☐ Vollständige Konsole-Ausgaben immer anzeigen Close Sourcetree



Setup Git Software to Work on Your Computer (7)



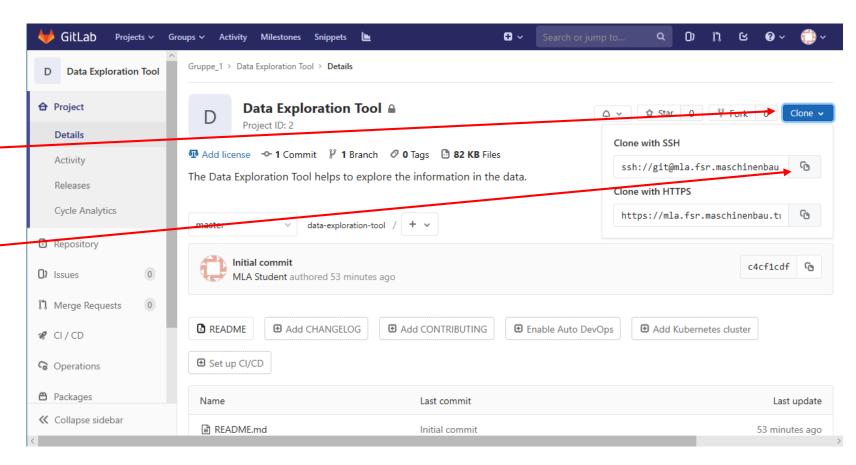
Datei Bearbeiten Ansicht Projektarchiv Aktionen Tools Hilfe Go back to Sourcetree and click on *Tools* Optionen and select *Preferences / Optionen* Allgemein Updates Mercurial Eigene Aktionen Authentifizierung Sourcetree erlauben, Änderungen an der globalen Git- und Mercurial-Konfiguration durchzuführen Use this version of SourceTree for URI association Choose the private key file that you saved two In Zukunft "Lesezeichen erstellen" anbieten Standard Benutzer Informationen slides before Allständiger Name: Simon Mehringskötter E-Mail Adresse: | mehringskoetter@fsr.tu-darmstadt.de SSH-Verbindungskonfiguration SSH Schlüssel: Click on OK -SSH-Client: PuTTY / Plink (Nur Git, Mercurial nutzt auf Windows immer Plink) ☑ Starte den SSH Agent automatisch wenn Sourcetree gestartet wird Repo Settings Projekt Ordner: Sprache: Automatisch (Erfordert Neustart) Helfen Sie dabei Sourcetree zu übers Standard Textcodierung | utf-8 🔻 Behalte Sicherungen bei gefährlichen Operationen Automatisch aktualisieren, wenn sich Dateien ändern Close Sourcetree Aktualisieren wenn die Anwendung nicht den Fokus hat Remotes auf Aktualisierungen überprüfen alle 10 Minuten Close the PuTTY-Agent in the task bar ✓ Projektarchiv-Tabs beim Starten wiederherstellen ☐ Vollständige Konsole-Ausgaben immer anzeigen right click on it and exit



Setup Git Software to Work on Your Computer (8)



- Go to the website of Gitlab and open the project you want to have on your computer
- Click on Clone
- Click the copy symbol below Clone with SSH





Setup Git Software to Work on Your Computer (9)



Datei Bearbeiten Ansicht Projektarchiv Aktionen Tools Hilfe Open Sourcetree New tab Click on Clone Local Remote Add Create Enter the copied URL Clone leave the field by clicking somewhere Cloning is even easier if you set up a remote account else schinenbau.tu-darmstadt.de:11024/gruppe_1/data-exploration-tool.git Durchsucher Projektarchiv Typ: 📵 Überprüfe Quelle... 🦳 C:\Users\pcnutzer\Documents\data-exploration-tool Durchsucher If this window appears click on Ja (will only appear first time you want to clone a project data-exploration-tool from that server) Local Fol Soll der Server-Key akzeptiert werden? [Root] Der Schlüssel des Servers 'mla.fsr.maschinenbau.tu-darmstadt.de:11024' If another window appears click *OK* wurde noch nicht zwischengespeichert - deshalb schlug der Prozess fehl. Der Server-Schlüssel lautet: > Erweiterte Optionen (will only appear first time you want to clone a project Möchten Sie den Server-Schlüssel akzeptieren und den Vorgang from that server) wiederholen? Klone Close Sourcetree and open Sourcetree again in Nein



case the window appeared

Setup Git Software to Work on Your Computer (10)



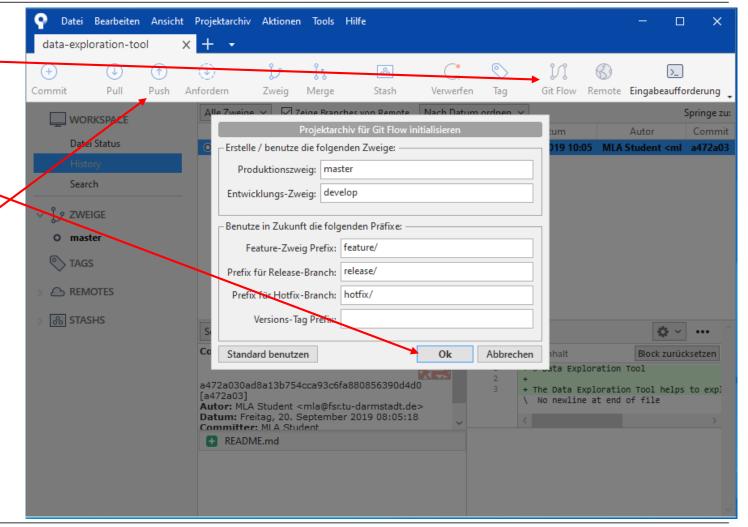
Datei Bearbeiten Ansicht Projektarchiv Aktionen Tools Hilfe Open Sourcetree Click on Clone Remote Create Enter the copied URL Clone leave the field by clicking somewhere Cloning is even easier if you set up a remote account else schinenbau.tu-darmstadt.de:11024/gruppe_1/data-exploration-tool.git Durchsucher Projektarchiv Tope W Dies ist ein Git Projektarchiv C:\Users\pcnutzer\Documents\data-exploration-tool It should state that this is a Git project Durchsucher data-exploration-tool You can adjust the local path where Local Folder: [Root] the git project should be stored if you want > Erweiterte Optionen Klone Click Klone



Setup Git Software to Work on Your Computer (11)



- You can create a Git flow in order to have a develop and master branch
- Leave everything as it is and click Ok
- Push the develop branch to the server (if it is not already there)
- Now you can use the Git flow to create releases and features that are tagged automatically





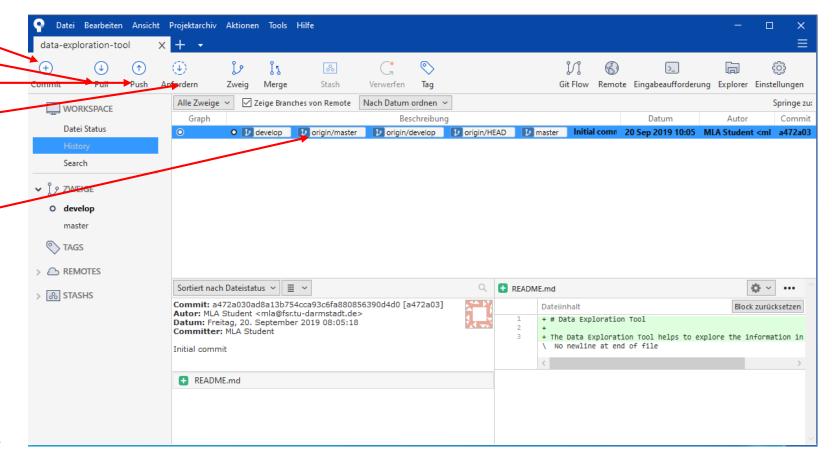
Setup Git Software to Work on Your Computer (12)



- Commit creates a snapshots
- Pull downloads a branch from server
- Push uploads a branch to server
- Anfordern checks branch status at server
- origin/XXXX are the branches on the server the ones without origin are the local branches
- ! Be careful before you push data to or pull data from the server – you can delete your local changes or the changes on the server made by other persons

Read: https://www.atlassian.com/git/tutorials/comparing-

workflows





Contact / Questions / Problems



Simon Mehringskötter, M. Sc.

Institute of Flight Systems and Automatic Control

L1|01 - 573

E-Mail: mehringskoetter@fsr.tu-darmstadt.de

Tel: +49 6151 16-21067







