Machine Learning Applications (MLA)

TECHNISCHE UNIVERSITÄT DARMSTADT

Instruction for Infrastructure

Overview of Gitlab

General Information

TECHNISCHE UNIVERSITÄT DARMSTADT

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

TECHNISCHE UNIVERSITÄT DARMSTADT

- Make sure LDAP is selected
- Enter your username and password
- Click *Sign in*

Create a Gitlab Group for Your Group (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

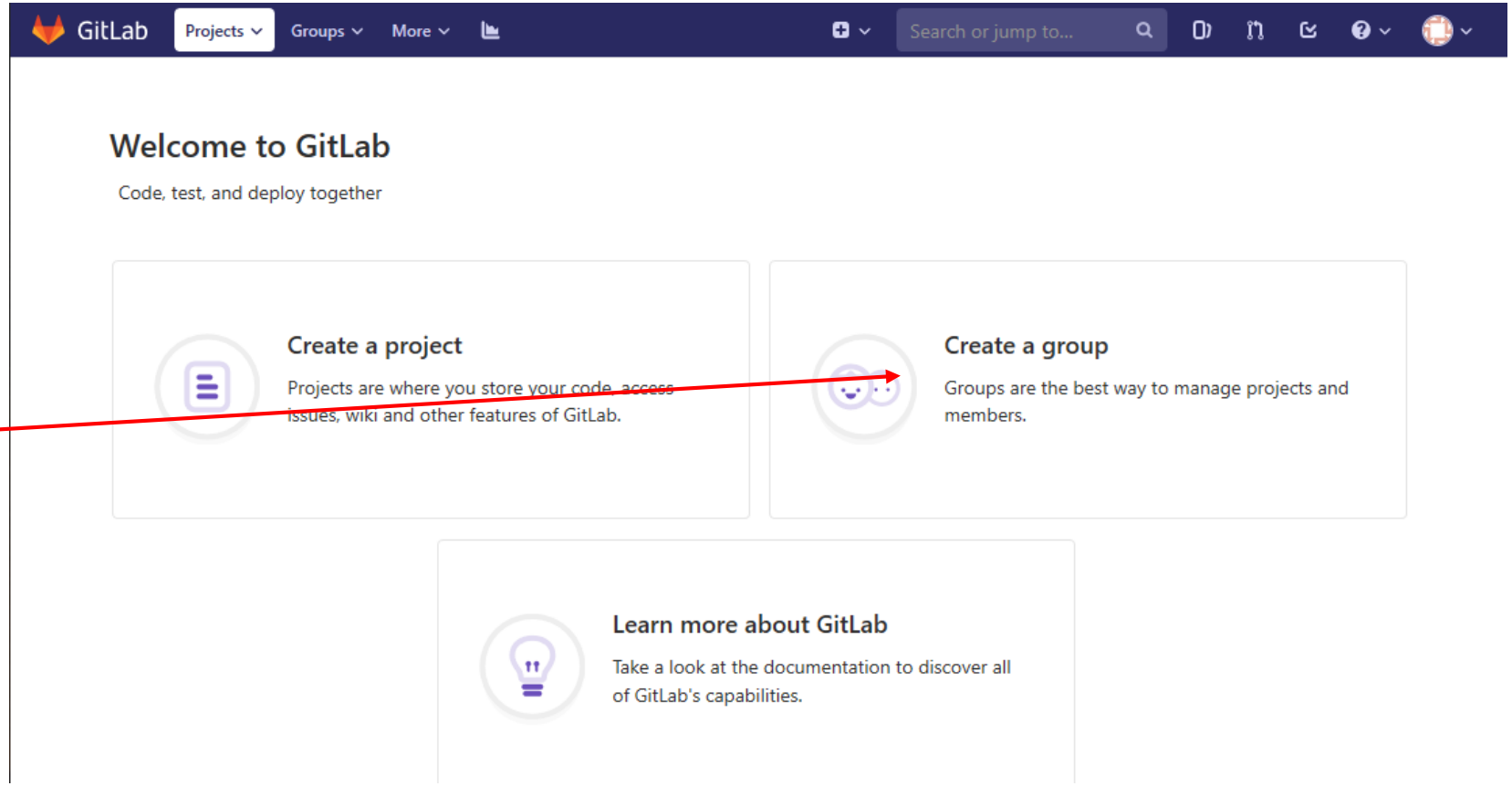
- A Gitlab group is used to collect different projects together
- Each person of your group has to login to Gitlab once to be added to a group → you can not add a person that has never loged-in
- One person of your group has to create the group
- You should name the group according to your group

→ See next slide

Create a Gitlab Group for Your Group (2)

TECHNISCHE UNIVERSITÄT DARMSTADT

- This is the start page after your first login
- Click on *Create a group*

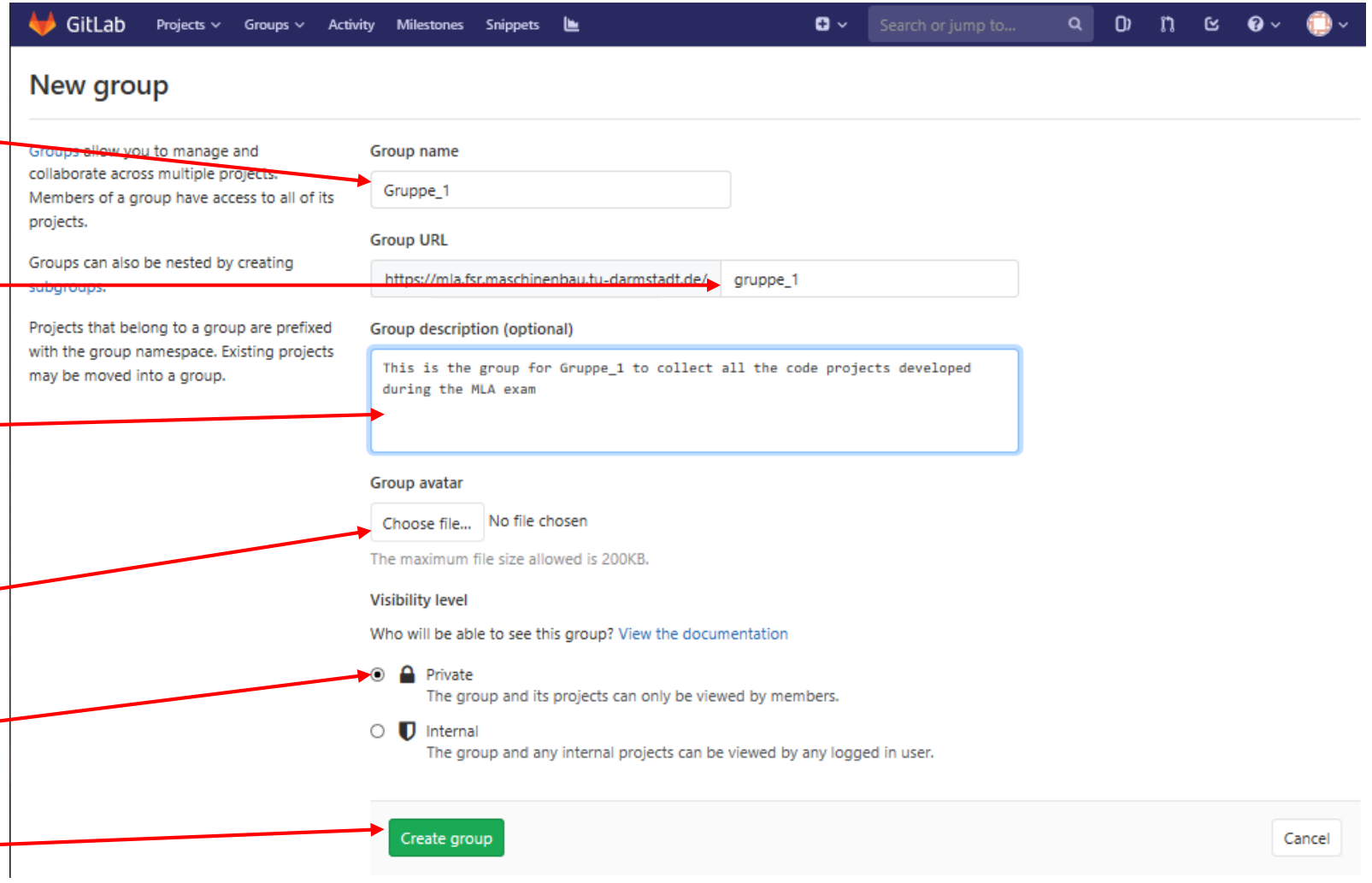


Create a Gitlab Group for Your Group (3)

TECHNISCHE UNIVERSITÄT DARMSTADT

- Enter your group name (e.g. Gruppe_1)
- Do **not** modify this field!
- Enter a description of your group
- Upload an Avatar if you want
- Leave visibility level at *Private*

→ Click *Create group*

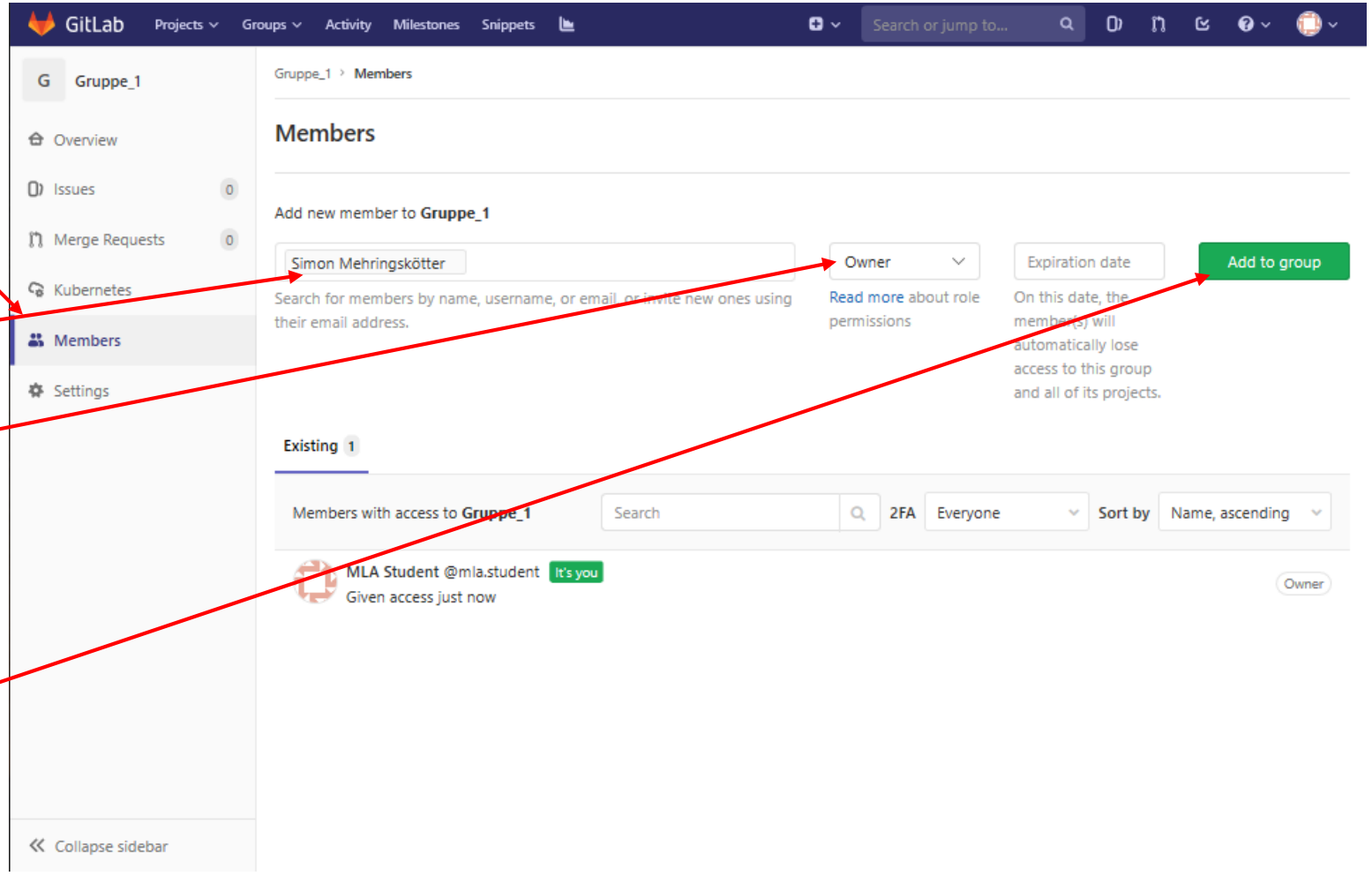


Create a Gitlab Group for Your Group (4)

TECHNISCHE UNIVERSITÄT DARMSTADT

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

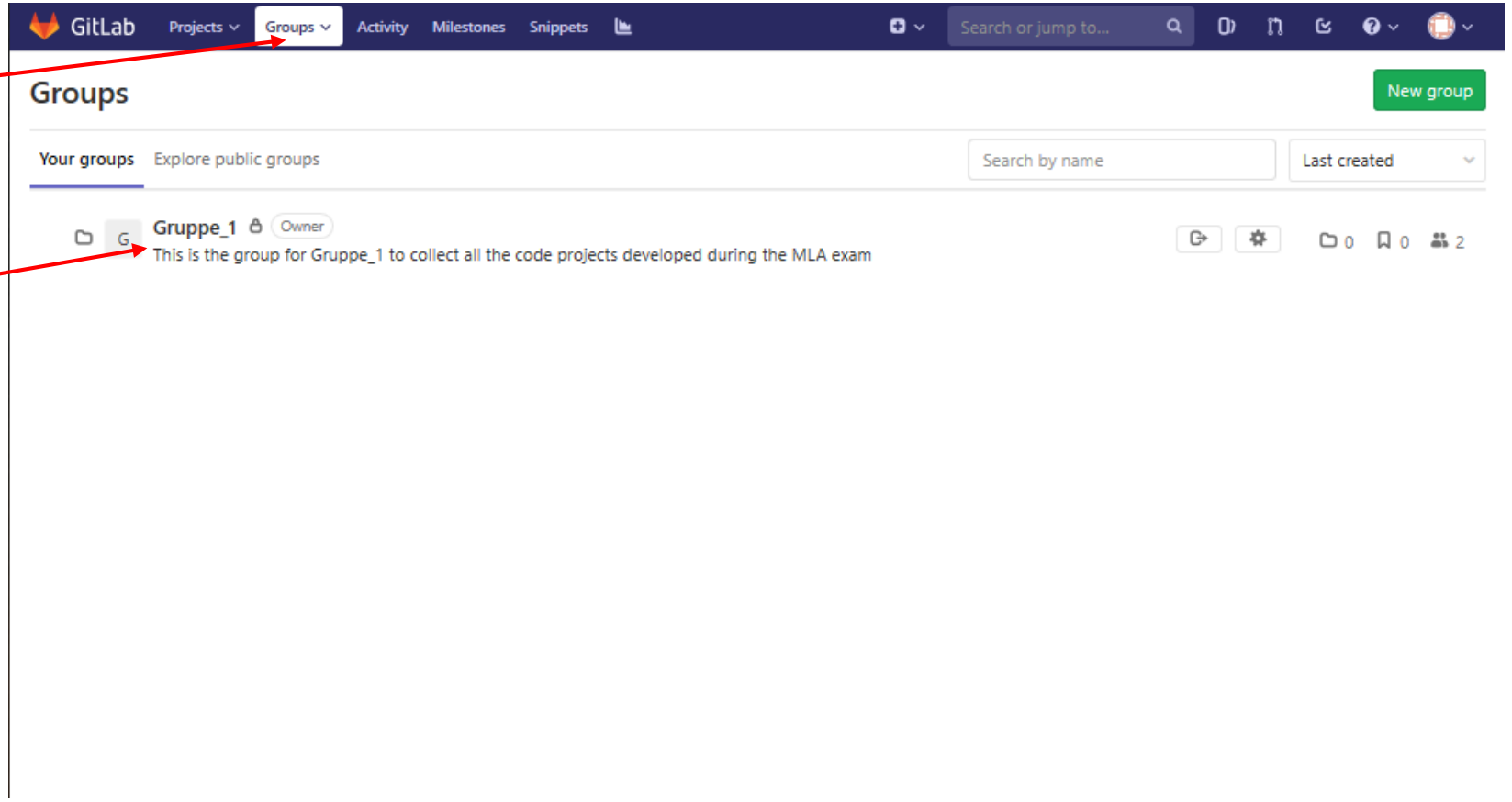
→ Click *Add to group*



Create a Gitlab Project in Your Group (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

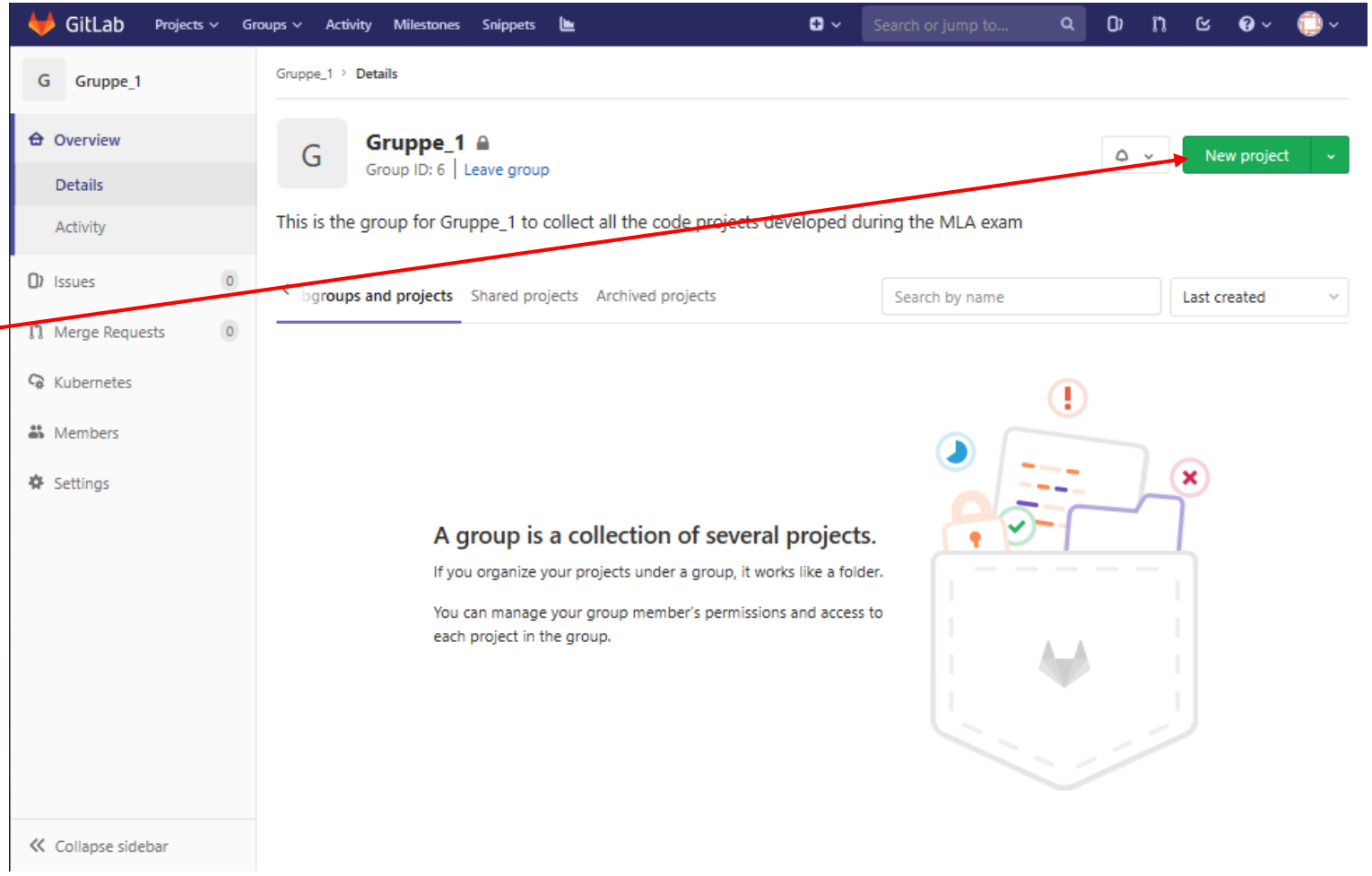
- Click on *Groups* and select *Your Groups*
- Click on your group name (e.g. Gruppe_1)



Create a Gitlab Project in Your Group (2)

TECHNISCHE UNIVERSITÄT DARMSTADT

- 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!

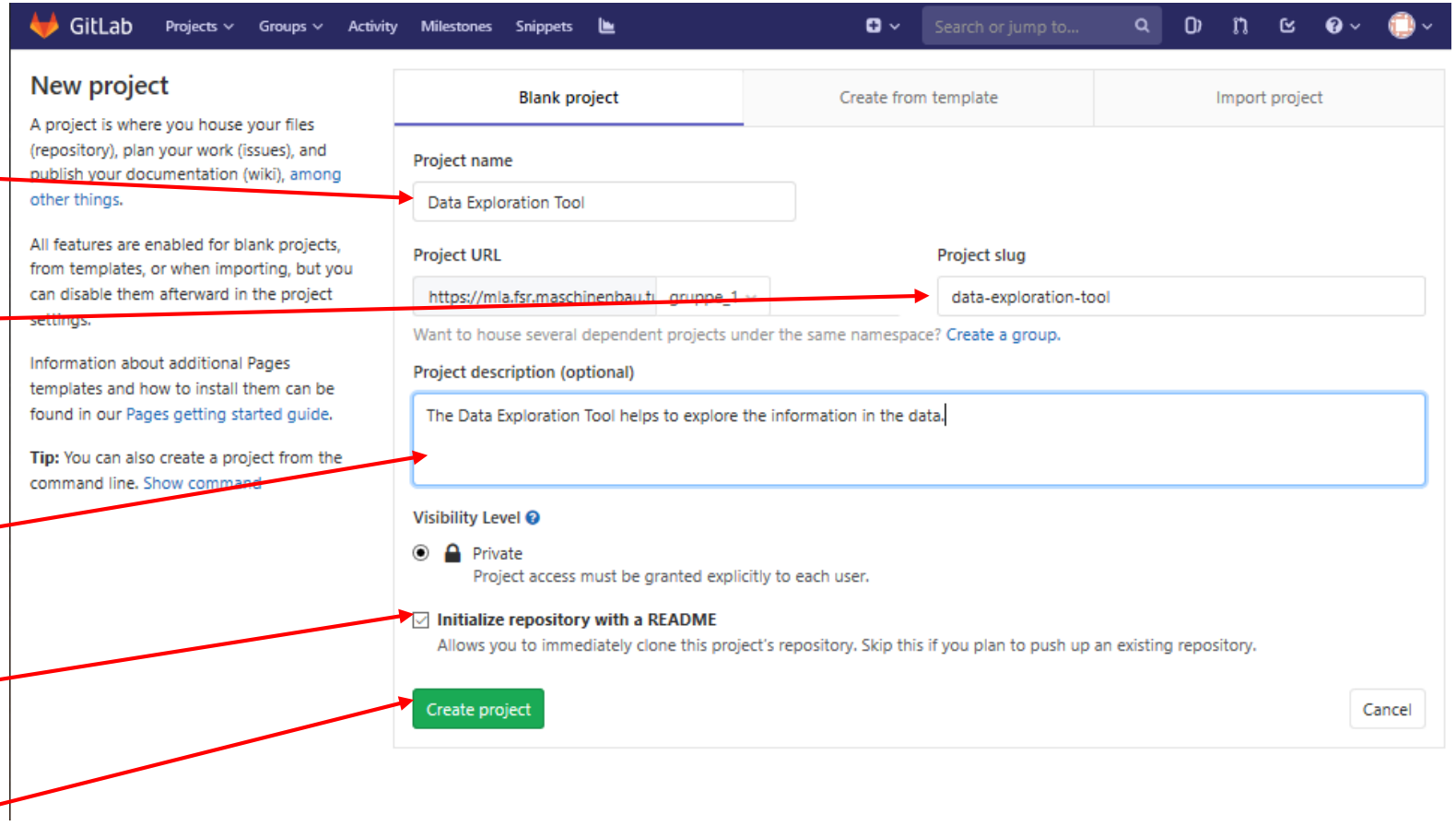


Create a Gitlab Project in Your Group (2)

TECHNISCHE UNIVERSITÄT DARMSTADT

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

→ Click *Create project*

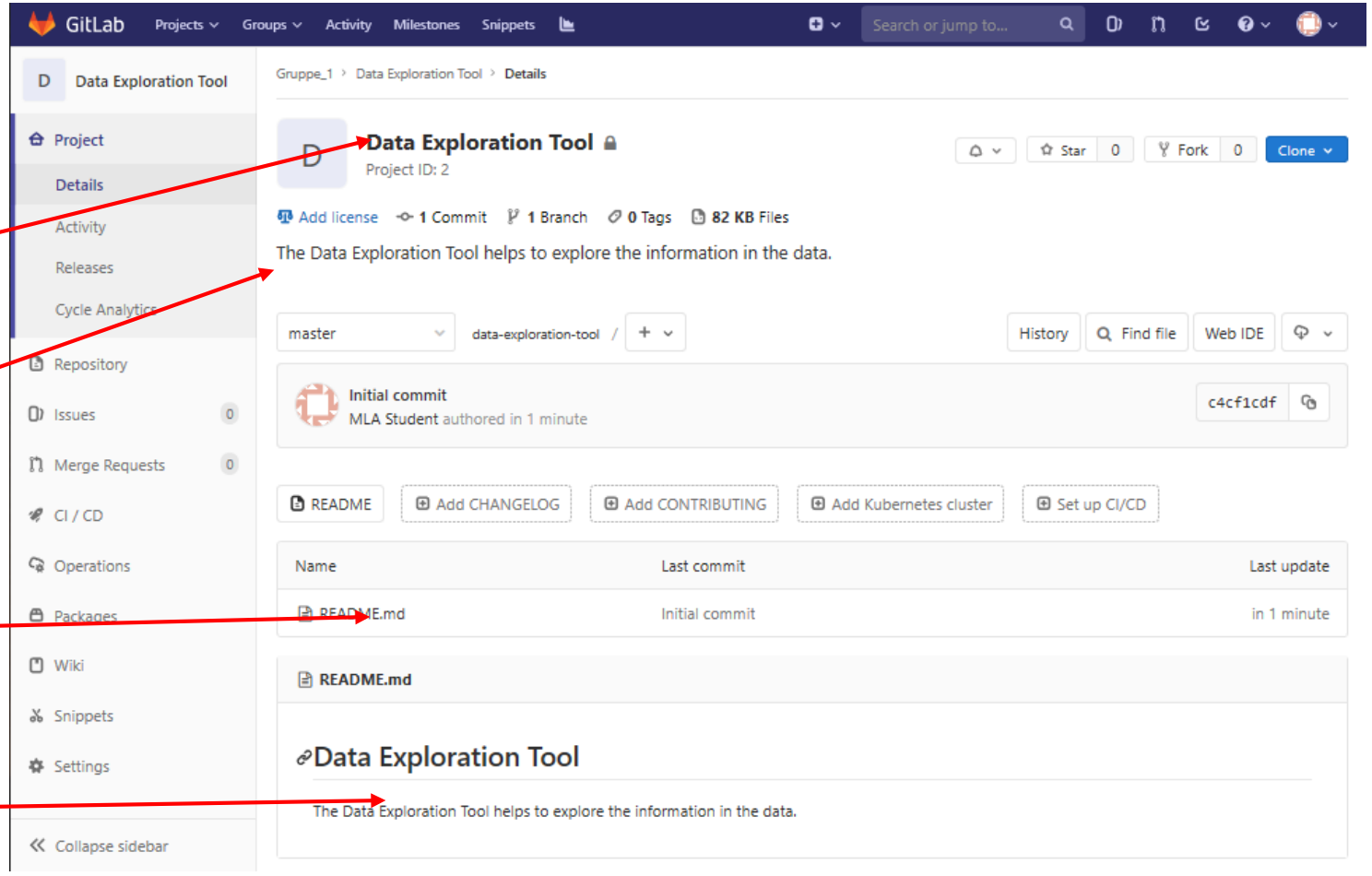


Create a Gitlab Project in Your Group (3)

TECHNISCHE UNIVERSITÄT DARMSTADT

This is the overview of the project.

- Here is the project name
- Here is your description
- The files in your project
- The readme file (initially the description)

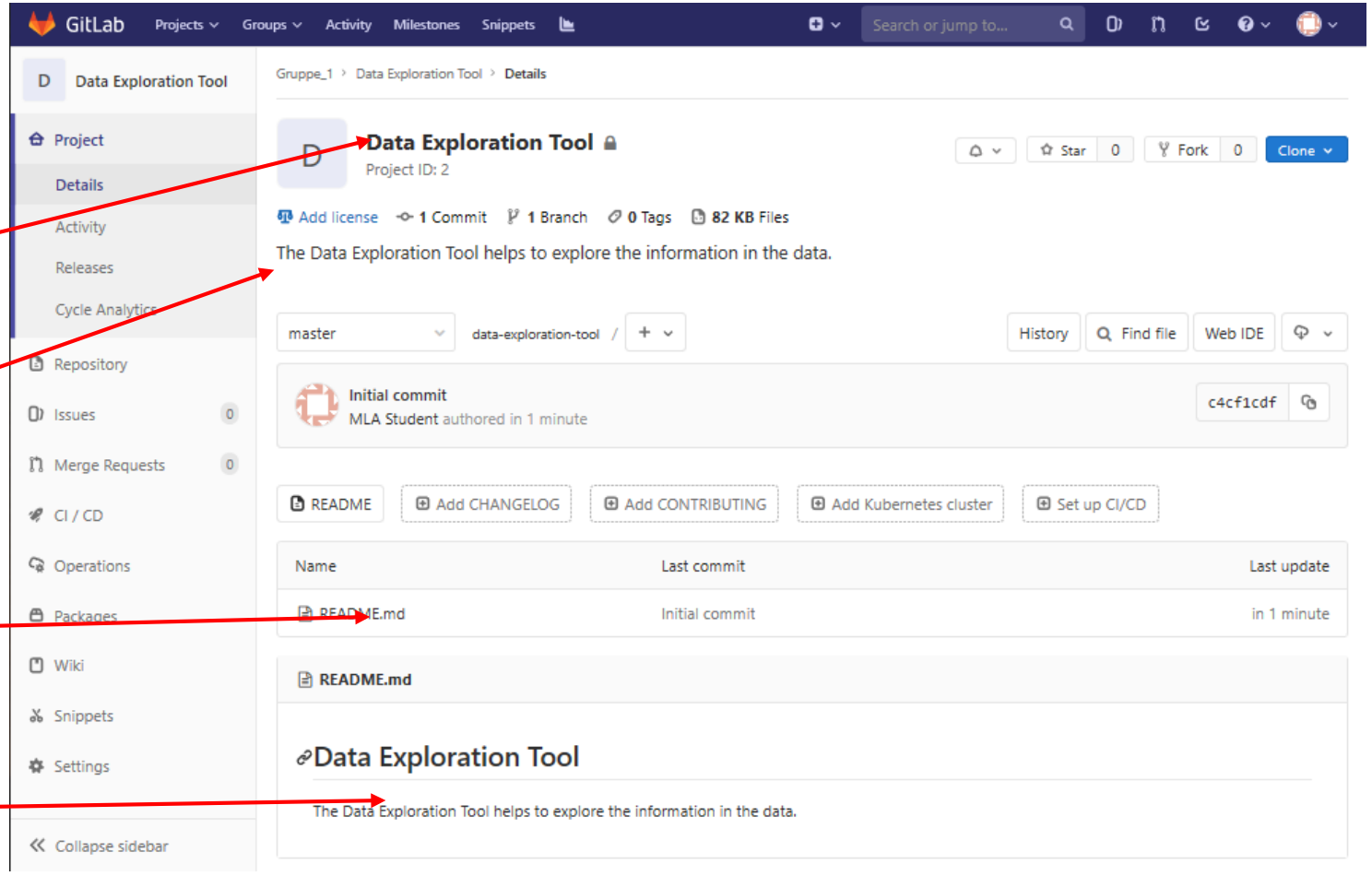


Create a Gitlab Project in Your Group (4)

TECHNISCHE UNIVERSITÄT DARMSTADT

This is the overview of the project.

- Here is the project name
- Here is your description
- The files in your project
- The readme file (initially the description)



Some Remarks on How to Use Gitlab

TECHNISCHE UNIVERSITÄT DARMSTADT

- 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

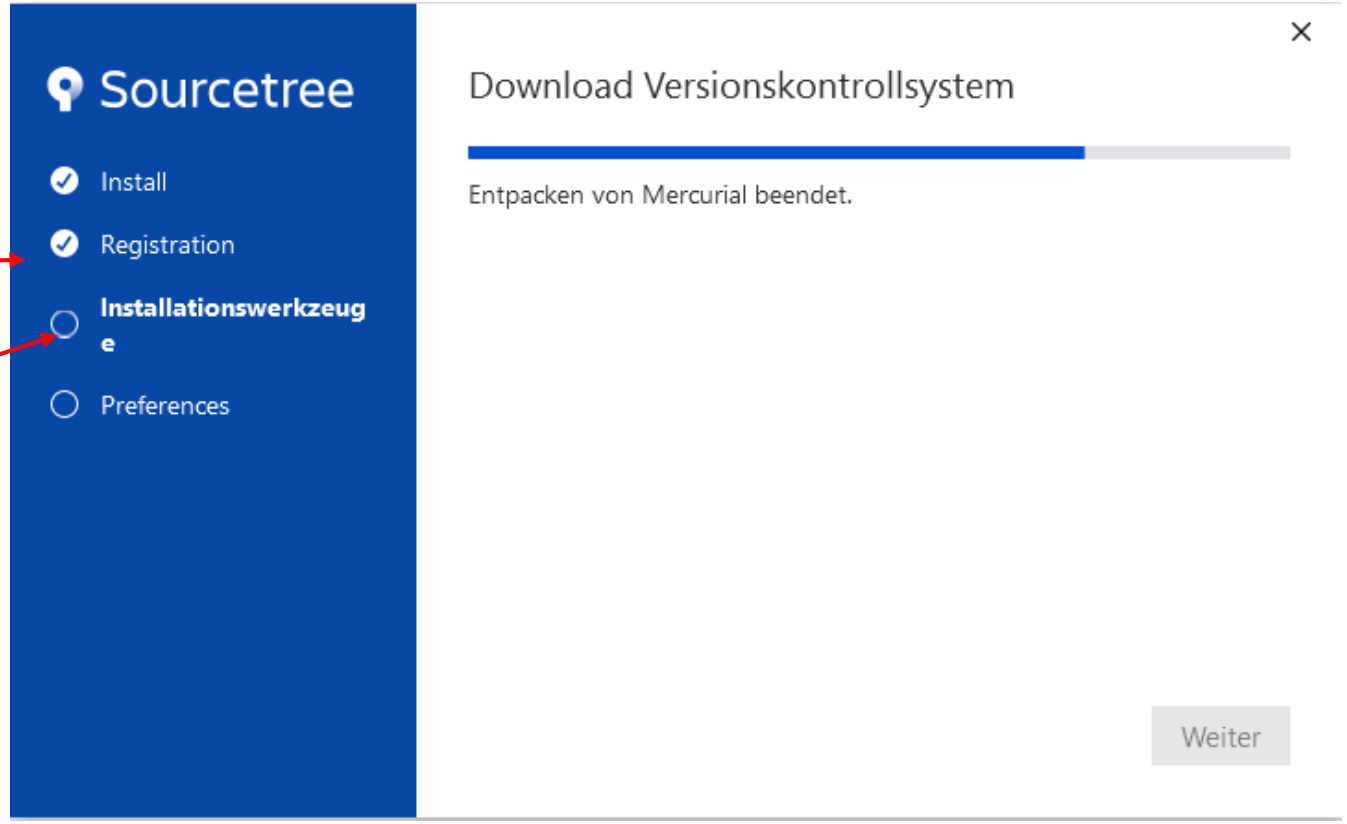
Setup Git Software to Work on Your Computer (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

- 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)

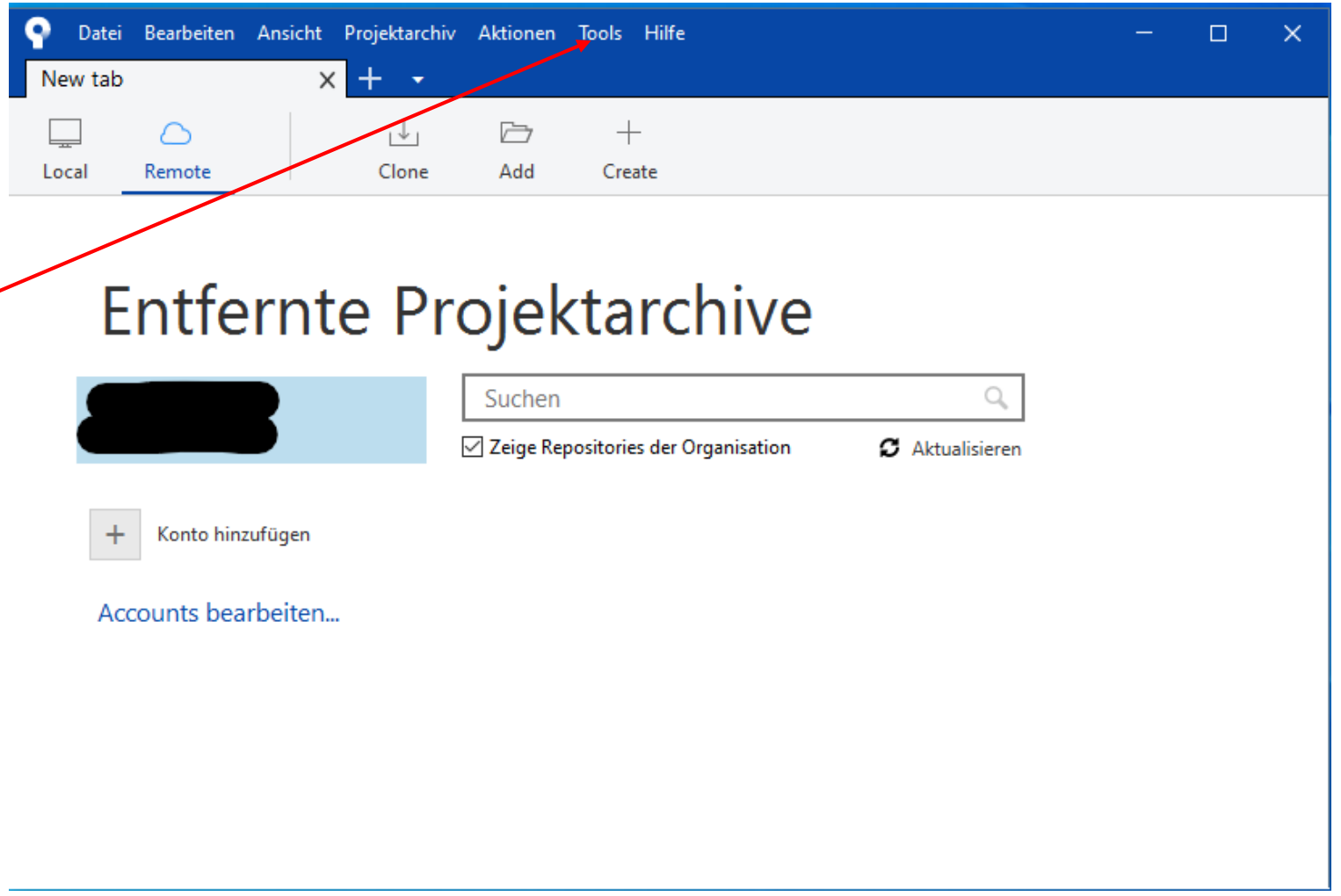
- Download and install Sourcetree from the website
- Follow the steps of the wizard
  - Registration
    - You have to click *Create one for free* if you don‘t have an account
    - Click on *Bitbucket* and Login with your account
  - Installationswerkzeuge
    - Select Git and Mercurial
- After installation Sourcetree asks for a SSH-Key → Click on *No*



Setup Git Software to Work on Your Computer (3)

TECHNISCHE UNIVERSITÄT DARMSTADT

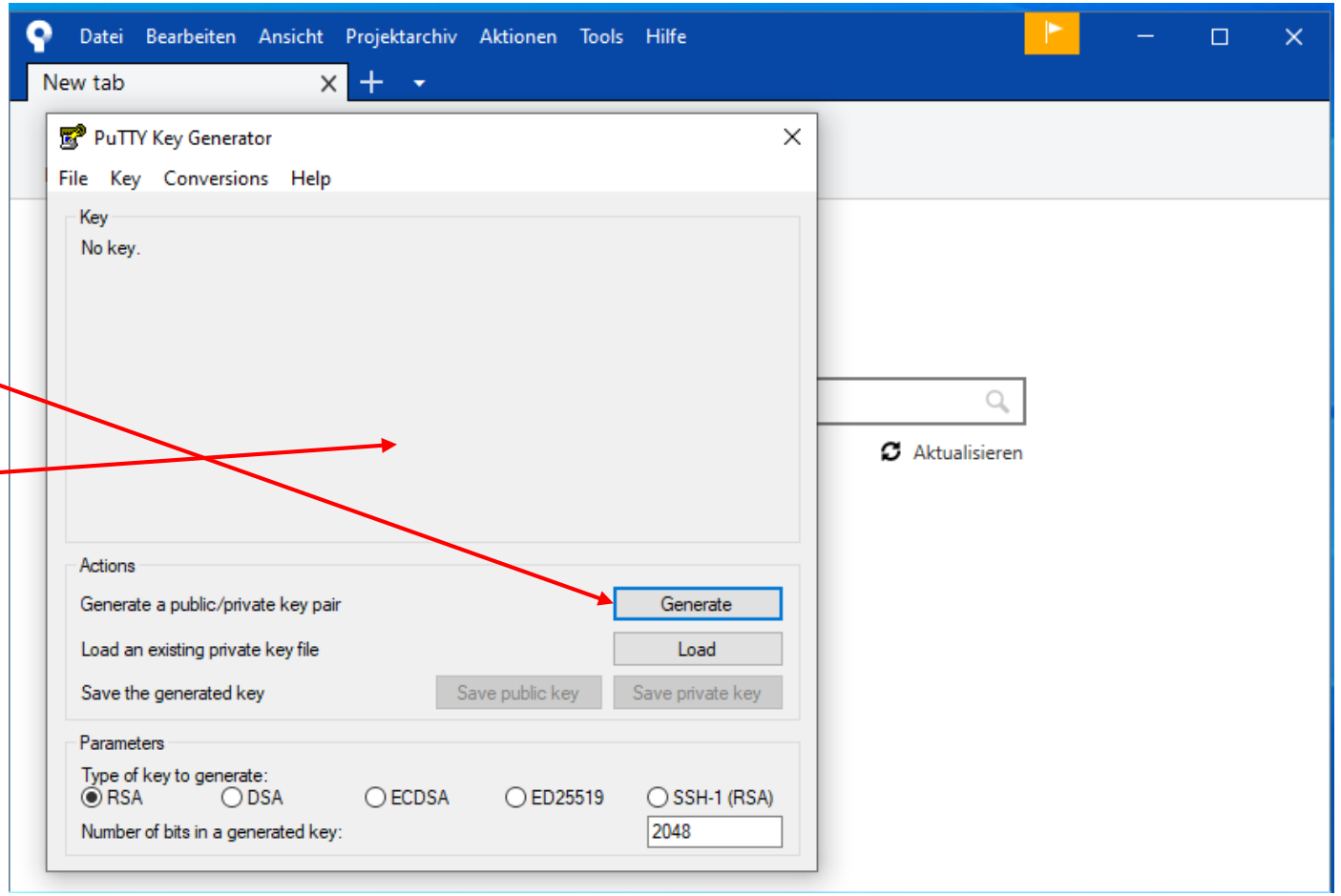
- This is the main window of Sourcetree
- First you should create a SSH-Key
- Click on *Tools* and then on *Import or Create SSH-Key / SSH-Schlüssel importieren oder erstellen*



Setup Git Software to Work on Your Computer (4)

TECHNISCHE UNIVERSITÄT DARMSTADT

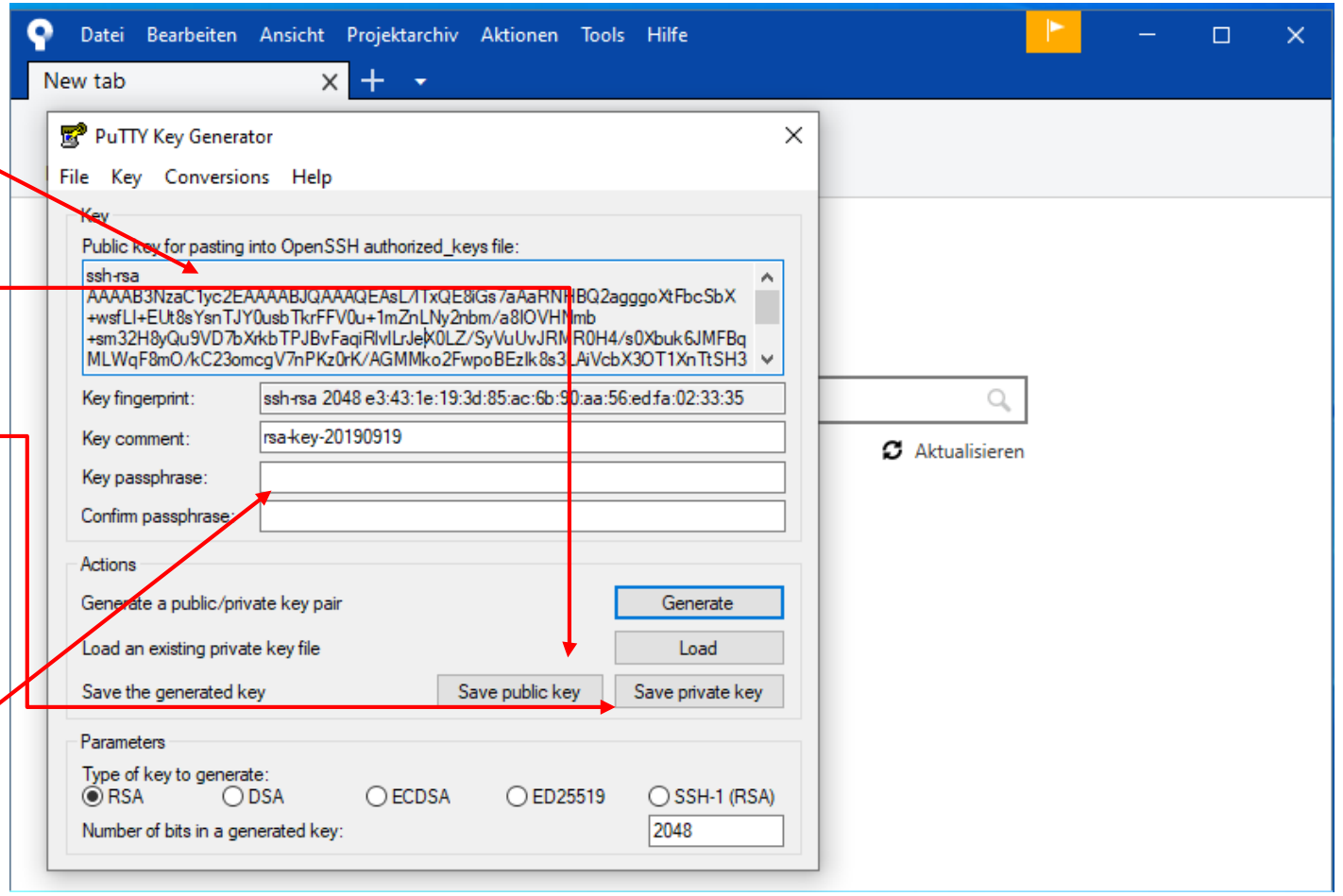
- The *PuTTY Key Generator* opens
- Click on *Generate*
- Move your mouse in the blank area until the bar is filled



Setup Git Software to Work on Your Computer (5)

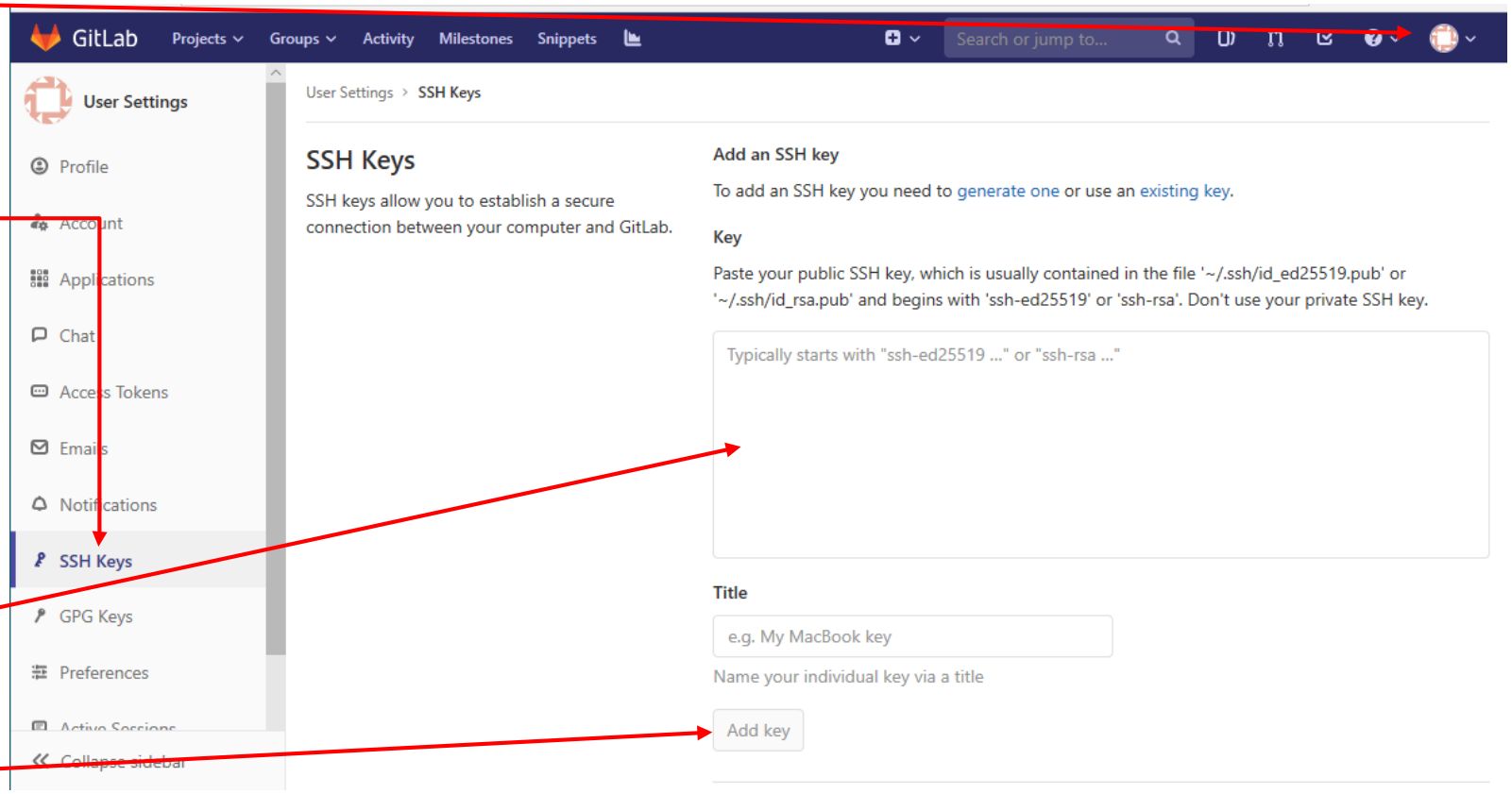
TECHNISCHE UNIVERSITÄT DARMSTADT

- 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)

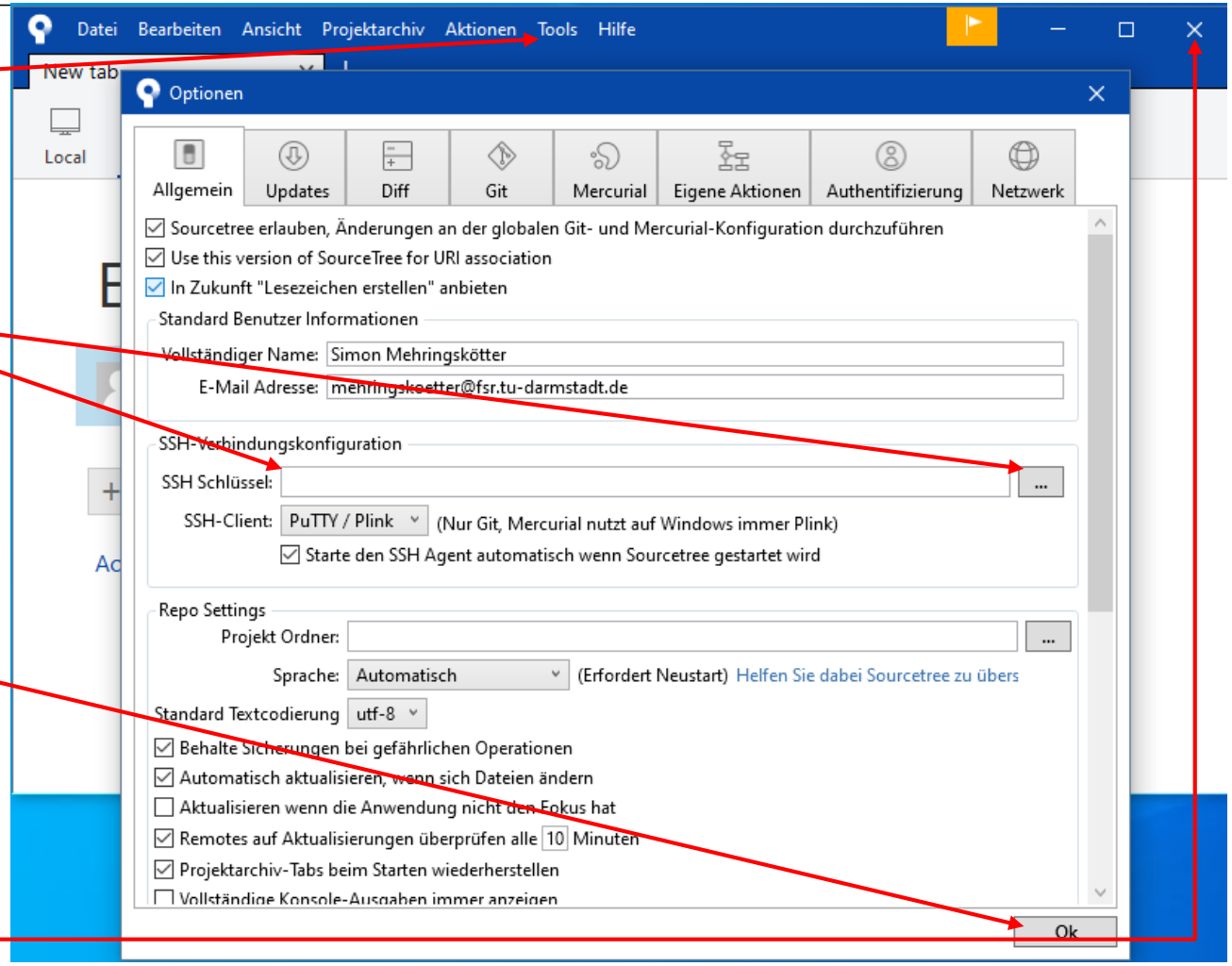
- Go to Gitlab webpage and click here choose *Settings* in the menu that opens
- Click on *SSH Keys*
- Paste the key from the clipboard into this field
- Click on *Add key*



Setup Git Software to Work on Your Computer (7)

TECHNISCHE UNIVERSITÄT DARMSTADT

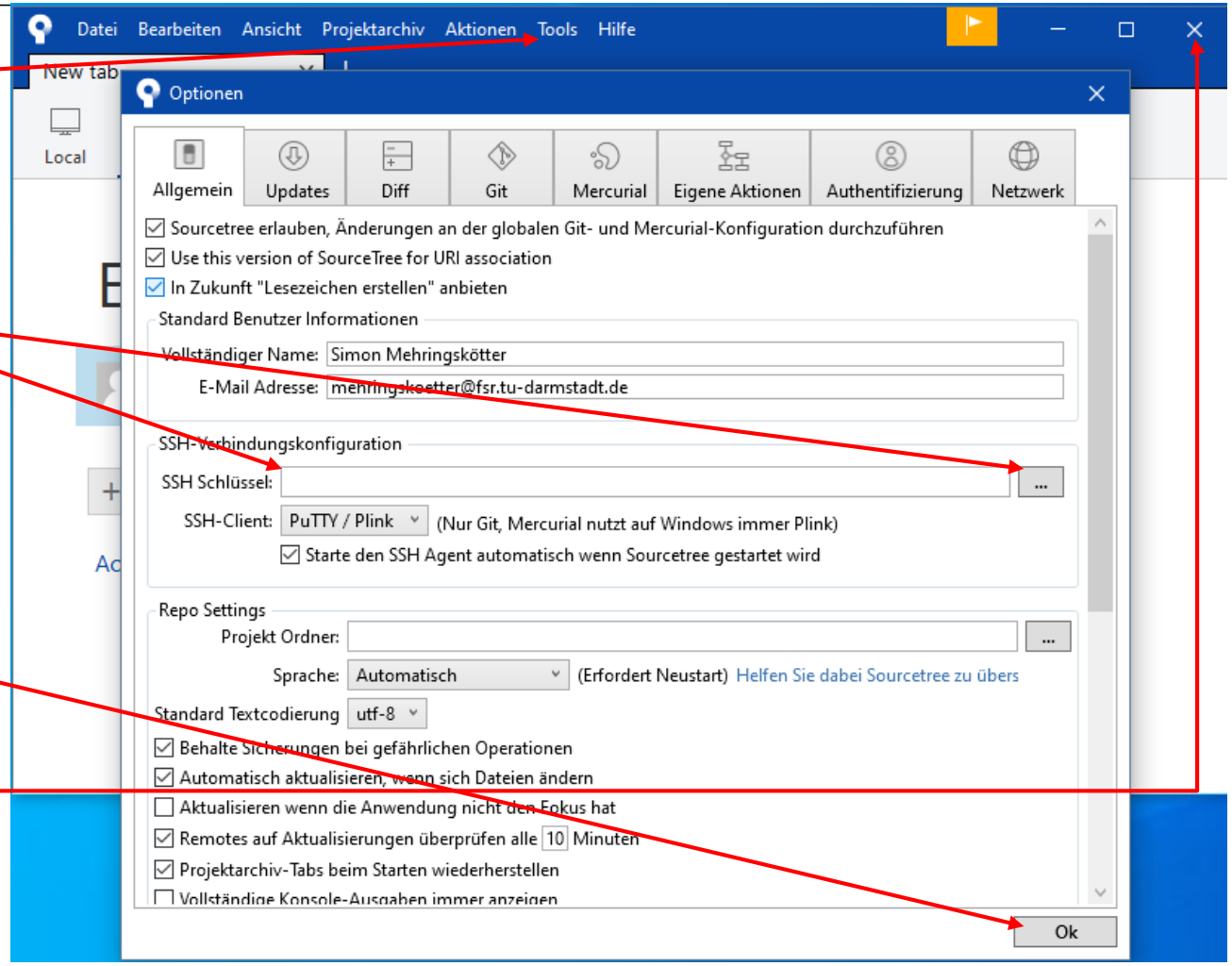
- Go back to Sourcetree and click on *Tools* and select *Preferences / Optionen*
- Choose the private key file that you saved two slides before
- Click on *OK*
- Close Sourcetree



Setup Git Software to Work on Your Computer (7)

TECHNISCHE UNIVERSITÄT DARMSTADT

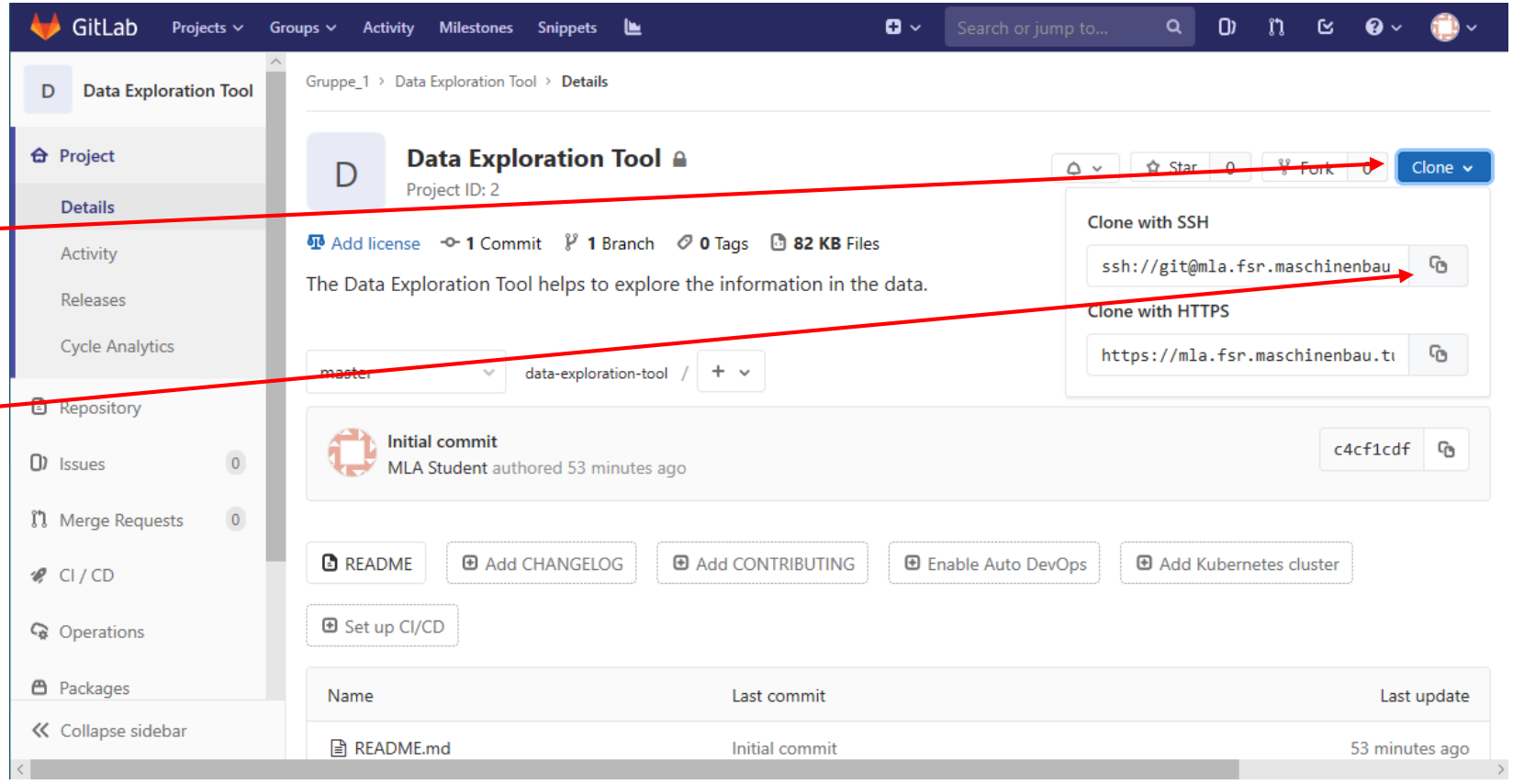
- Go back to Sourcetree and click on *Tools* and select *Preferences / Optionen*
- Choose the private key file that you saved two slides before
- Click on *OK*
- Close Sourcetree
- Close the PuTTY-Agent in the task bar right click on it and exit



Setup Git Software to Work on Your Computer (8)

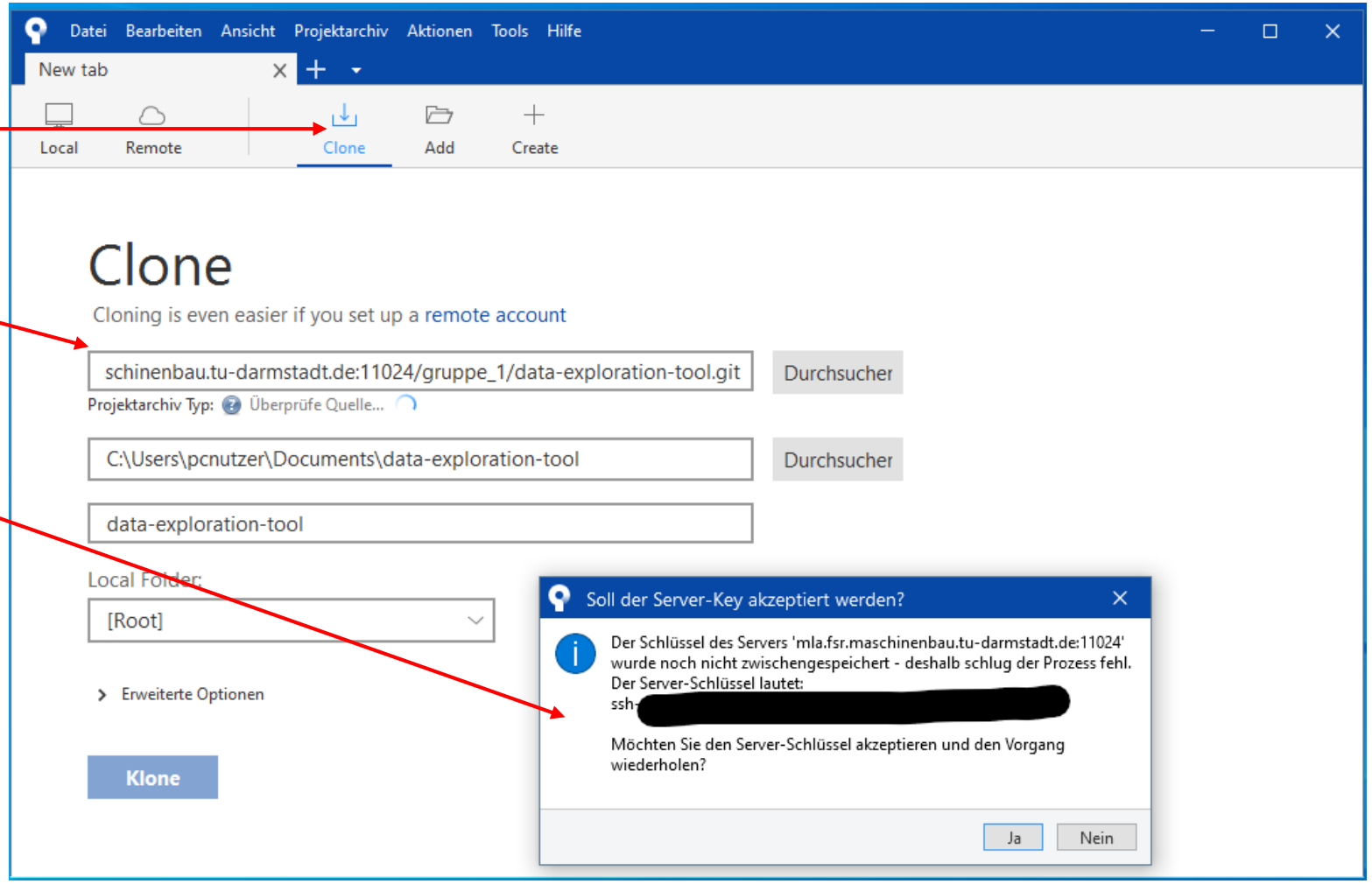
TECHNISCHE UNIVERSITÄT DARMSTADT

- 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)

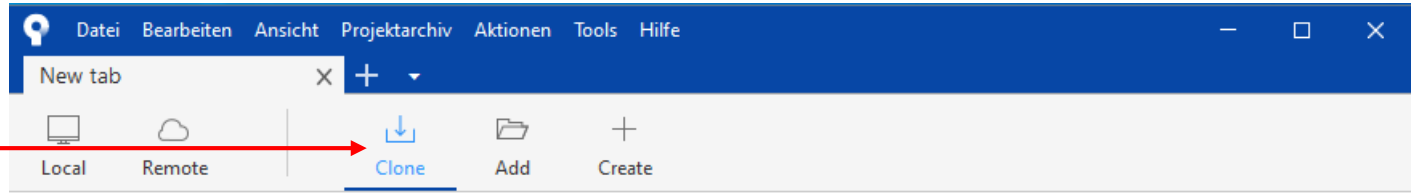
- Open Sourcetree
- Click on *Clone*
- Enter the copied URL
  leave the field by clicking somewhere else
- If this window appears click on *Ja*
  (will only appear first time you want to clone a project from that server)
- If another window appears click *OK*
  (will only appear first time you want to clone a project from that server)
- **Close** Sourcetree and **open** Sourcetree again in case the window appeared



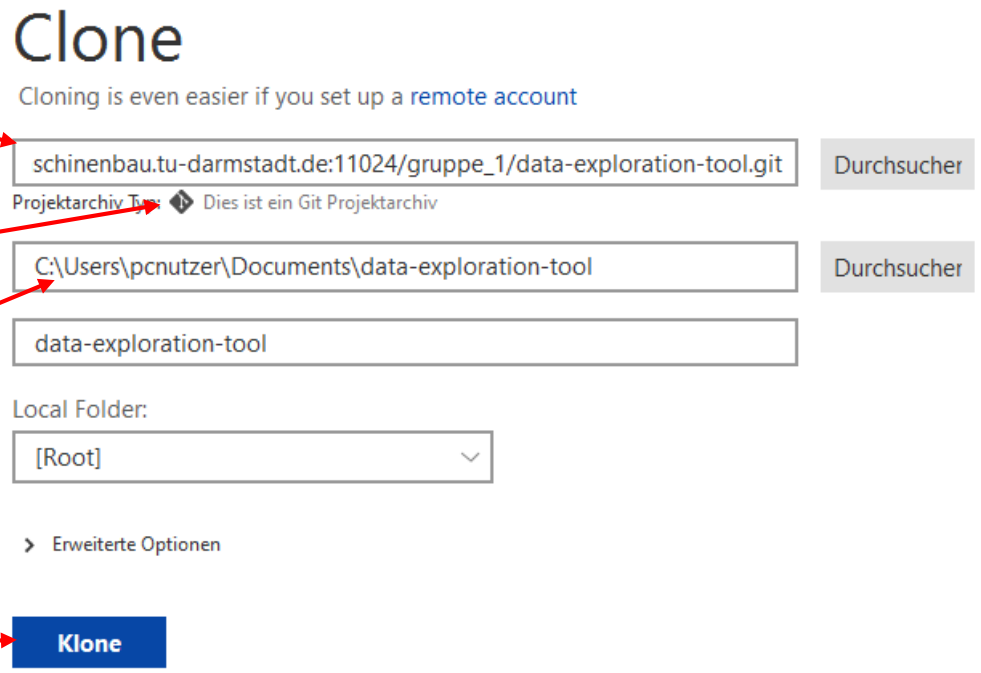
Setup Git Software to Work on Your Computer (10)

TECHNISCHE UNIVERSITÄT DARMSTADT

- Open Sourcetree
- Click on *Clone*



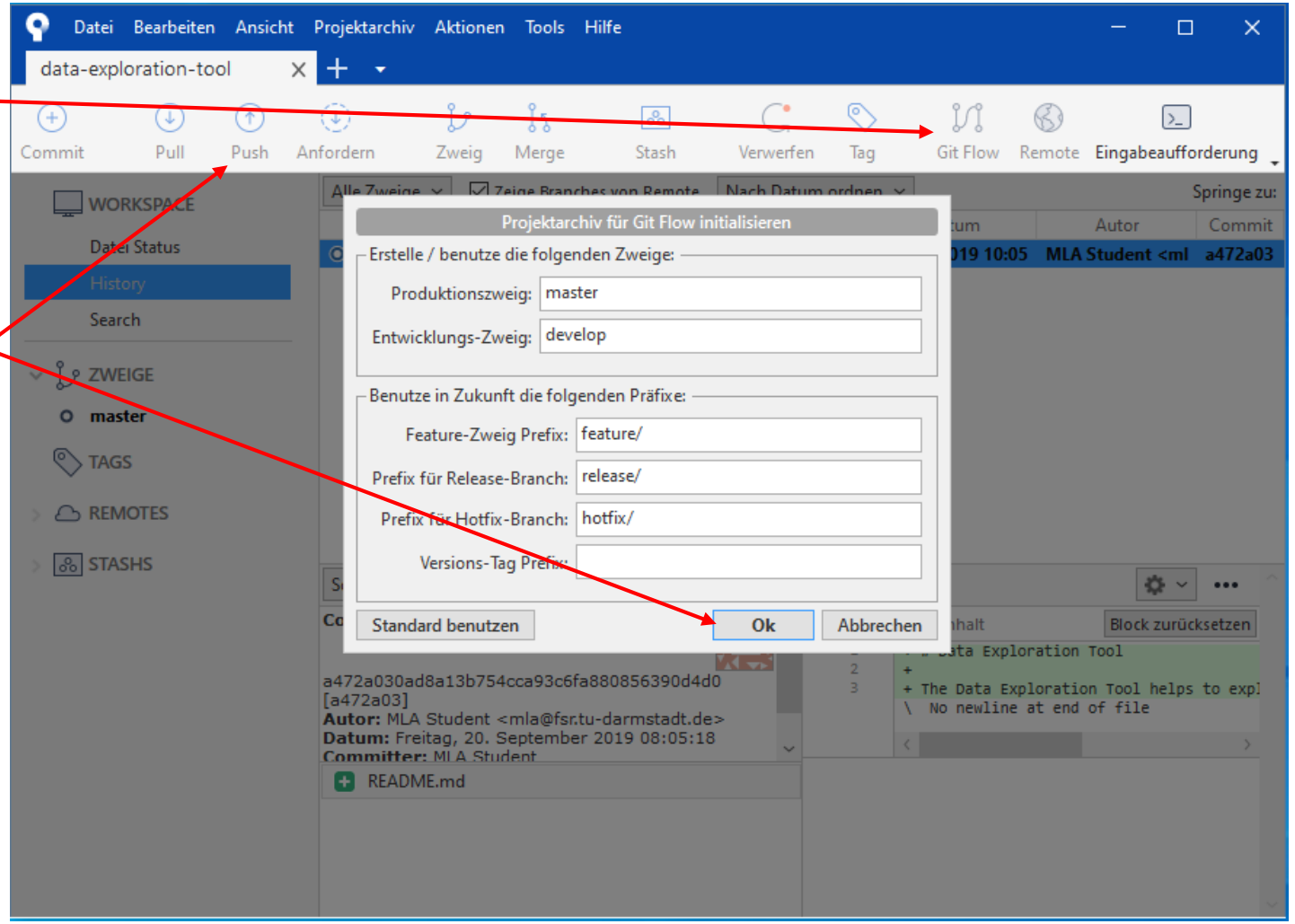
- Enter the copied URL
  leave the field by clicking somewhere else
- It should state that this is a Git project
- You can adjust the local path where the git project should be stored if you want



- 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)

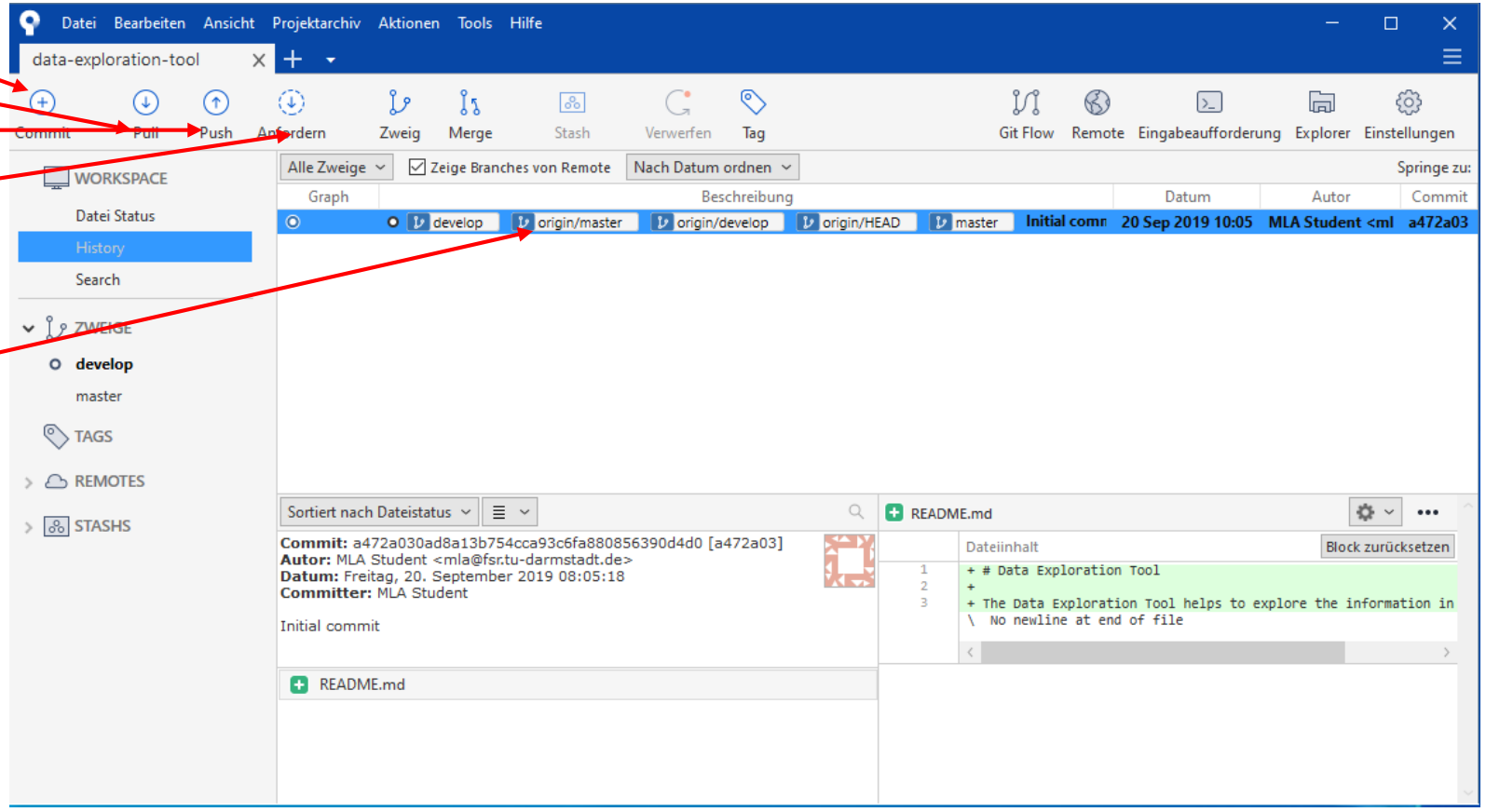
TECHNISCHE UNIVERSITÄT DARMSTADT

- *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


THANK U