


# When Software Enters the Stage

**Roel Aaij**

**November 23rd, 2022**

Nik|hef



# Version Control

- It's very easy to forget what you did even a few days ago
- When working on multiple projects, it's worse
- When writing your thesis, you'll need to remake plots
  
- You want to keep track of the history of what you're doing
  - Data
  - Code
  - Documentation
  
- The answer is: version control
- A very popular tool: git
- <https://swcarpentry.github.io/git-novice/>

# GitLab

- We now have a repository somewhere on stoomboot, let's make that more useful
- go to <https://gitlab.nikhef.nl> and hit “Federated login”



## GitLab Community Edition

Username or email

Password

Remember me

[Forgot your password?](#)

Sign in

Don't have an account yet? [Register now](#)

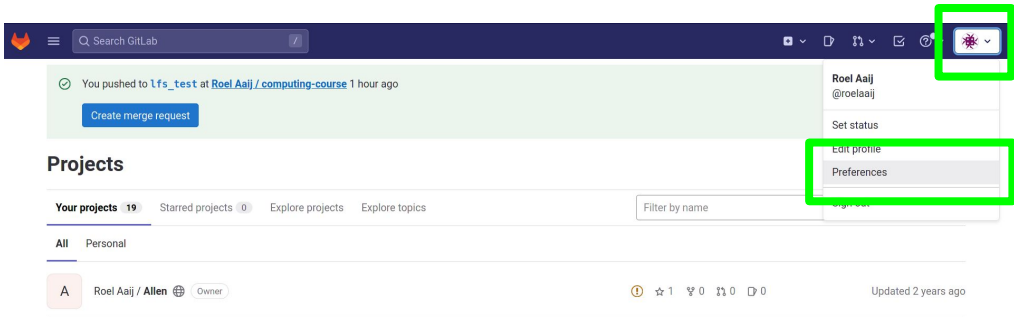
Sign in with

Federated login

Remember me

# GitLab SSH Keys

- Intermezzo: add your SSH key to gitlab
- <https://www.nikhef.nl/pdp/computing-course/work/ssh.html>
- `cat ~/.ssh/id_rsa.pub`
- Click on your profile and select “Preferences”



The screenshot shows the GitLab web interface. At the top, there is a dark blue navigation bar with the GitLab logo, a search bar, and several utility icons. A green box highlights the user profile icon in the top right corner. Below the navigation bar, a notification banner indicates a successful push to the 'lfs\_test' branch. Underneath, there is a 'Projects' section with filters for 'Your projects' (19), 'Starded projects' (0), and 'Explore projects'. A 'Filter by name' input field is also visible. The user's profile information is shown, including the name 'Roel Aaij' and the handle '@roelaaij'. A green box highlights the profile dropdown menu, which contains options like 'Set status', 'Edit profile', and 'Preferences'. The 'Preferences' option is highlighted in grey.

# GitLab SSH Keys

- Select “SSH Keys” from the menu on the left
- Copy paste your public key into the box and add it with a title

The screenshot shows the GitLab user settings interface. On the left sidebar, the 'SSH Keys' option is highlighted with a green box. The main content area is titled 'SSH Keys' and includes a description: 'SSH keys allow you to establish a secure connection between your computer and GitLab.' Below this is a section for 'SSH Fingerprints' with a link to the 'current instance configuration'. A large text input field for the key is highlighted with a green box. Below the key field, there are fields for 'Title' (with an example 'Example: MacBook key') and 'Expiration date' (set to '2023-11-23'). A green box highlights the 'Add key' button at the bottom.

Search GitLab

User Settings

- Profile
- Account
- Applications
- Chat
- Access Tokens
- Emails
- Password
- Notifications
- SSH Keys**
- GPG Keys
- Preferences
- Active Sessions
- Authentication log

SSH keys allow you to establish a secure connection between your computer and GitLab.

**SSH Fingerprints**

SSH fingerprints verify that the client is connecting to the correct host. Check the [current instance configuration](#).

Add an SSH key for secure access to GitLab. [Learn more](#).

**Key**

begins with 'ssh-rsa', 'ssh-dss', 'ecdsa-sha2-nistp256', 'ecdsa-sha2-nistp384', 'ecdsa-sha2-nistp512', 'ssh-ed25519', 'sk-ecdsa-sha2-nistp256@openssh.com', or 'sk-ssh-ed25519@openssh.com'.

**Title**

Example: MacBook key

Key titles are publicly visible.

**Expiration date**

2023-11-23

Optional but recommended. If set, key becomes invalid on the specified date.

Add key

# New GitLab project

## Projects

[New project](#)[Your projects](#) 19[Starred projects](#) 0[Explore projects](#)[Explore topics](#)[Name](#)[All](#)[Personal](#)

A

Roel Aaij / **Allen**

Owner



1



0



0



0

Updated 2 years ago

# New GitLab project

New project > **Create blank project**



## Project name

## Project URL

## Project slug

Want to organize several dependent projects under the same namespace? [Create a group](#).

## Visibility Level ?

-  Private  
Project access must be granted explicitly to each user. If this project is part of a group, access is granted to members of the group.
-  Internal  
The project can be accessed by any logged in user except external users.

## Project Configuration

- Initialize repository with a README  
Allows you to immediately clone this project's repository. Skip this if you plan to push up an existing repository.
- Enable Static Application Security Testing (SAST)  
Analyze your source code for known security vulnerabilities. [Learn more](#).

# New GitLab project

- Copy the commands, but instead of https, we need ssh

 Project 'Test Project' was successfully created. 

To make it easy for you to get started with GitLab, here's a list of recommended next steps.

Already a pro? Just edit this README.md and make it your own. Want to make it easy? [Use the template at the bottom!](#)

## Add your files

- [Create](#) or [upload](#) files
- [Add files using the command line](#) or push an existing Git repository with the following command:

```
cd existing_repo
git remote add origin https://gitlab.nikhef.nl/roelaaij/test-project.git
git branch -M main
git push -uf origin main
```





# New GitLab project

- Copy the ssh URL

The screenshot shows the GitLab interface for a new project named "Test Project". The project ID is 307. It has 1 commit, 1 branch, 0 tags, and 61 KB of project storage. The main branch is selected. A "Clone" button is highlighted with a green box. A dropdown menu is open, showing options to clone with SSH, HTTPS, or open in an IDE. The SSH URL is `git@gitlab.nikhef.nl:roelaaij/t` and the HTTPS URL is `https://gitlab.nikhef.nl/roelaa`. The SSH URL is also highlighted with a green box.

**Test Project** Project ID: 307   Star 0 Fork 0

1 Commit 1 Branch 0 Tags 61 KB Project Storage

main test-project / +

**Use ssh to clone**  
Roel Aaij authored in 25 seconds

README Add LICENSE Add CHANGELOG Add CONTRIBUTING End

Set up CI/CD Configure Integrations

Find file Web IDE **Clone**

**Clone with SSH**  
git@gitlab.nikhef.nl:roelaaij/t

**Clone with HTTPS**  
https://gitlab.nikhef.nl/roelaa

**Open in your IDE**

## What about data?

- You should keep track of your data too
- Git is great for ASCII encoded files, not so much for binary
- Do it in the same repository, with symlinks
- Create folders for various types of files
  - Work in progress
  - Plot created from this data
  - Data used for thesis or publication
  - etc.
- Use the symlinks in your code and notebooks
- Commit them as part of the rest

## What about software?

- Create a conda environment  
<https://www.nikhef.nl/pdp/computing-course/software/where-to-get.html>
- Create conda spec file, and store it in your repository:  
`$> conda list --explicit > spec-file.txt`
- Create environment from spec file:  
`$> conda create --name myenv --file spec-file.txt`
- This relies on the availability of packages in conda
- For experiment software: include the version(s) of the release
- For software you develop:
  - Create a repository for your software
  - Create a release
  - Upload it to zenodo
  - cite the DOI
- For those who want more: [build containers](#)

# Zenodo

- Zenodo is a general purpose repository for scientific documents and data
- Hosted by CERN
- For each submission, a digital object identifier (DOI) is created, which allow citation of the submission
- Let's go and upload something:
  - `$> tar jcf my_repo.tar.bz2 my_repo`
  - <https://sandbox.zenodo.org/>
  - sign up (easiest with ORCID)
  - log in
  - upload

[Upload](#)[Communities](#) roelaaij@nikhef.nl

# arXiv

- arXiv is a preprint server
- The majority of papers in our fields are put on arXiv
- No peer-review, also many “interesting” papers
- A DOI is generated for each submission
- arXiv publications can be cited
- Don't forget to update the citation once a journal publication is available
- Many categories available:
  - <https://arxiv.org>
  - <https://arxiv.org/archive/hep-ex>
  - <https://arxiv.org/archive/hep-ph>
  - <https://arxiv.org/archive/hep-th>
  - <https://arxiv.org/archive/cs.LG>

# Journal

- In addition to what you create, you may also want to keep track of what you do
- A research journal
- You can do it on old-fashioned paper
- Or use a digital one and store it using git (with a remote on gitlab.nikhef.nl)
  
- Emacs + org-mode:  
<https://orgmode.org/worg/org-tutorials/org4beginners.html>
- Markdown in your editor of choice
- LaTeX
- Jupyter notebooks:  
<https://nextjournal.com/schmudde/how-to-version-control-jupyter>