FAIR Bioinfo 2022

Introduction to Git, GitHub/Gitlab

P. Marin, Mateo Hiriart, P. Ruiz, N. Goué
aubi@uca.fr

Université Clermont Auvergne, AuBi, Mésocentre

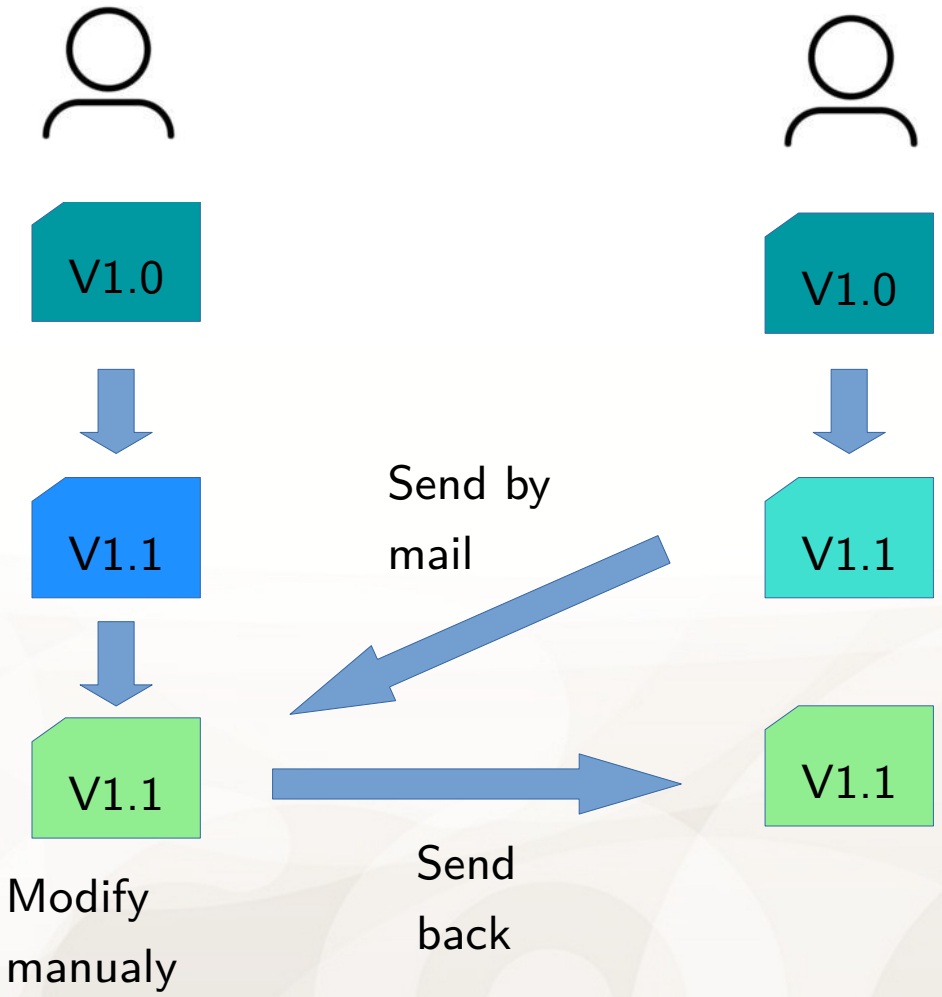
du 28 novembre au 02 décembre 2022

Version control system

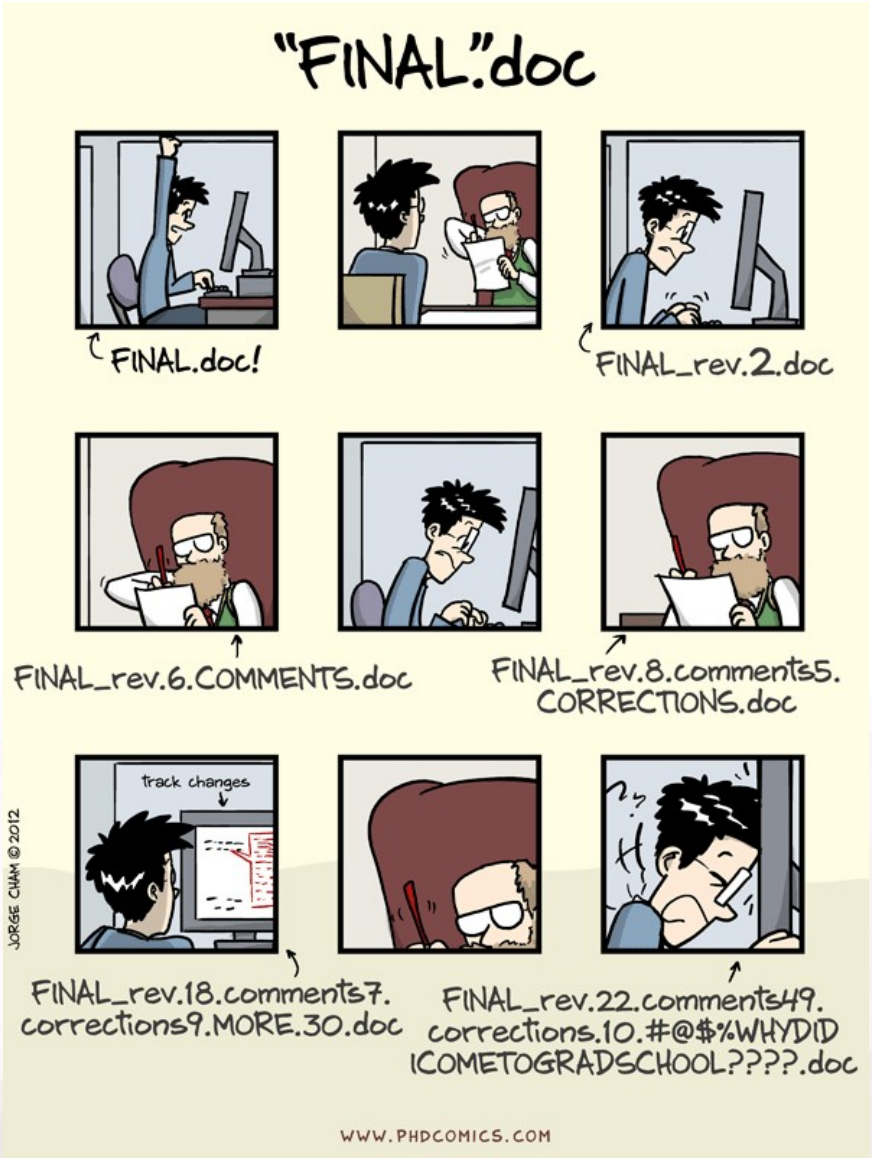
Online hosting service

for software base on git

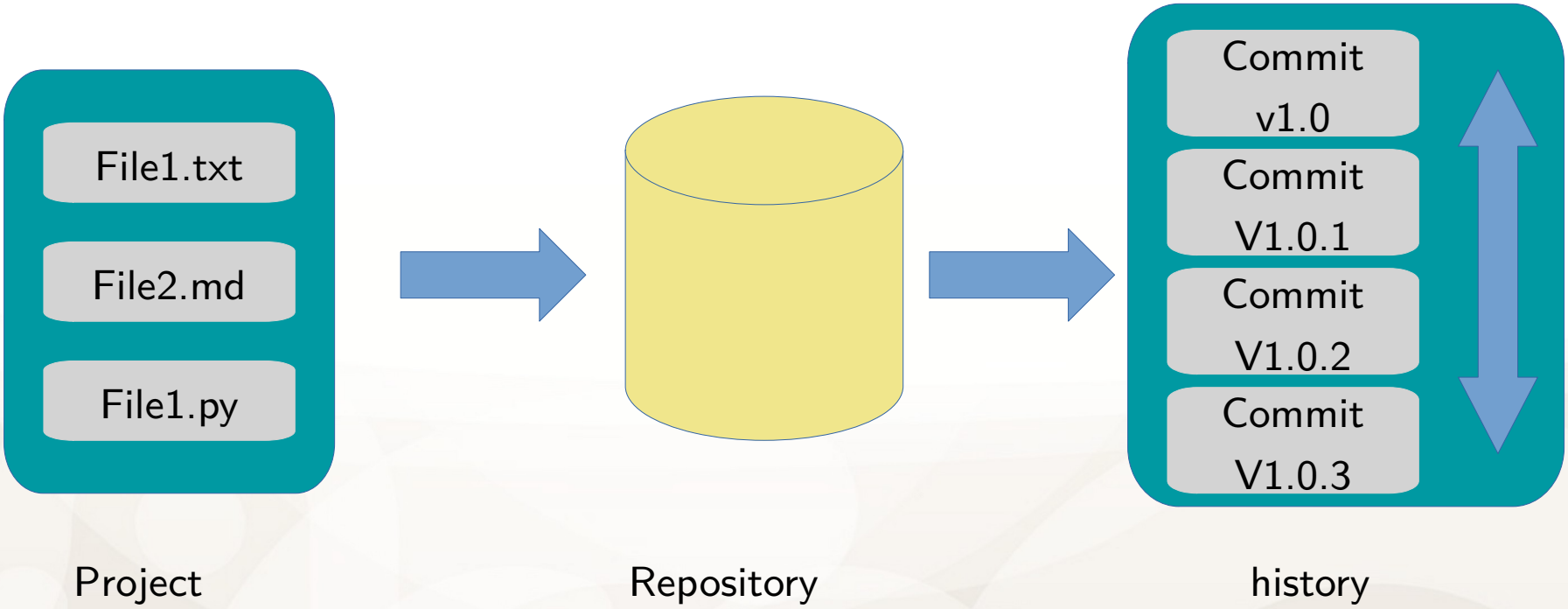
Sharing code



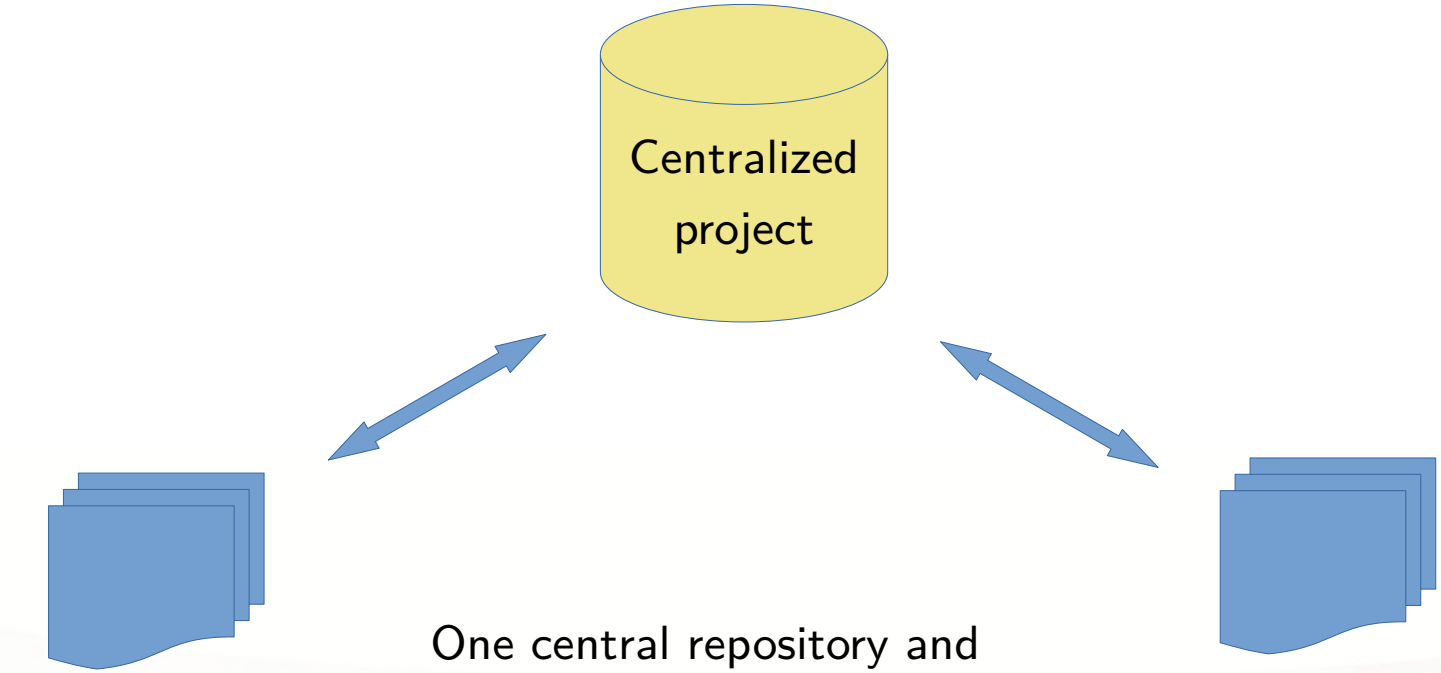
Sharing code



Git a version control system



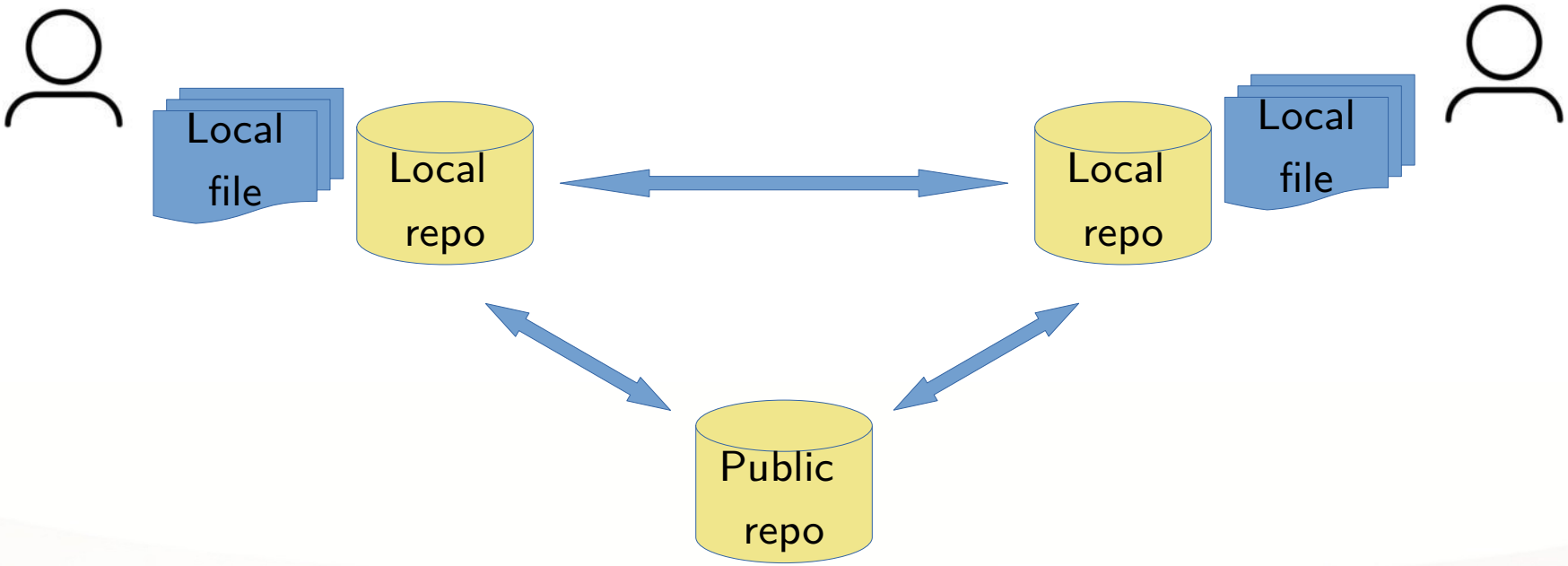
Centralized version control



One central repository and developers can only commit on it.

Example: Subversion, Team Foundation Server

Distributed version control



All repositories are interconnected and can be edited at same time.

Example: Git, Mercurial

Git

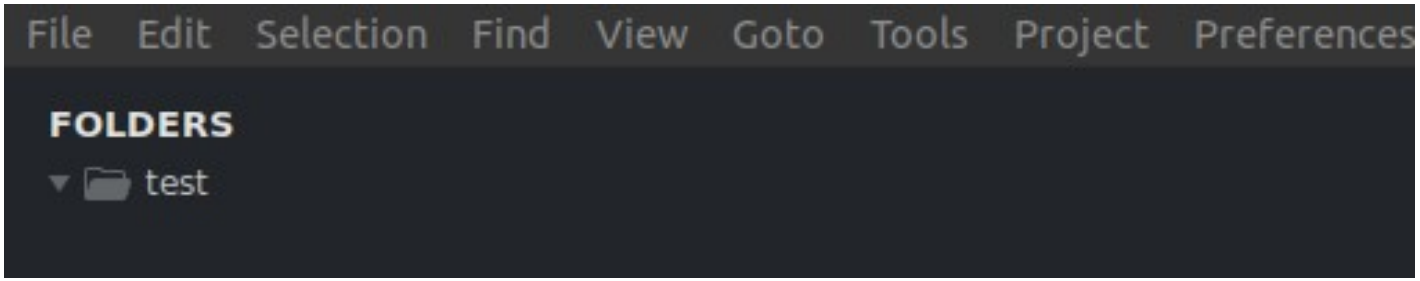
Git is a free and open source distributed version control system.

Handle any type of project, fast and efficiency.

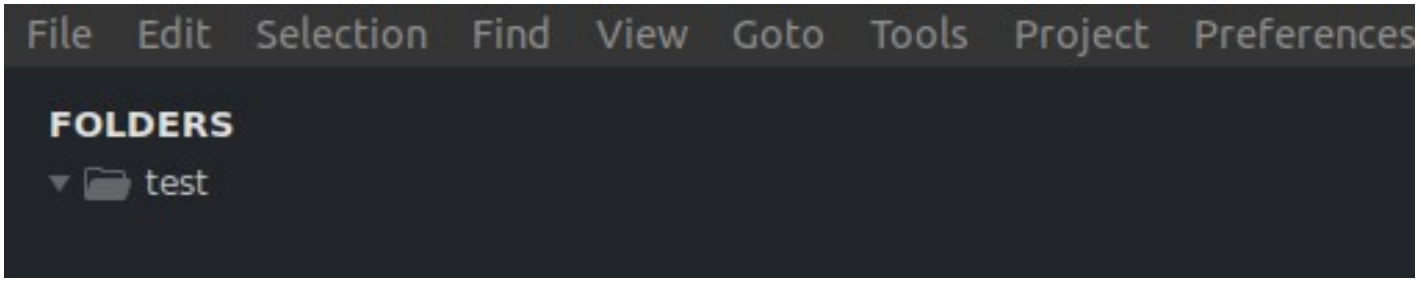
Created in 2005 by Linus Torvalds

UNIVERSITÉ Clermont Auvergne

Git: Initialisation



Git: Initialization

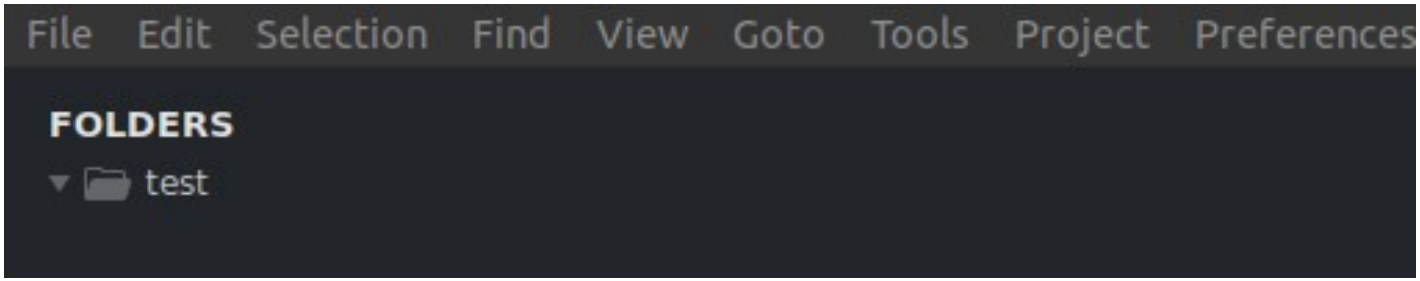


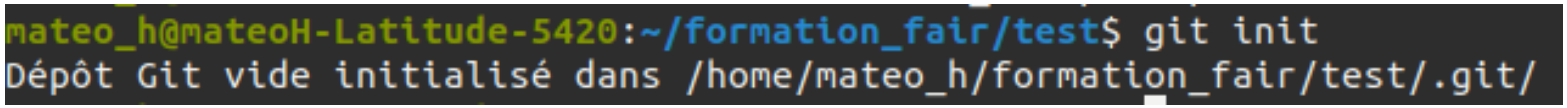

```
mateo_h@mateoH-Latitude-5420:~/formation_fair/test$ git init
Dépôt Git vide initialisé dans /home/mateo_h/formation_fair/test/.git/
```

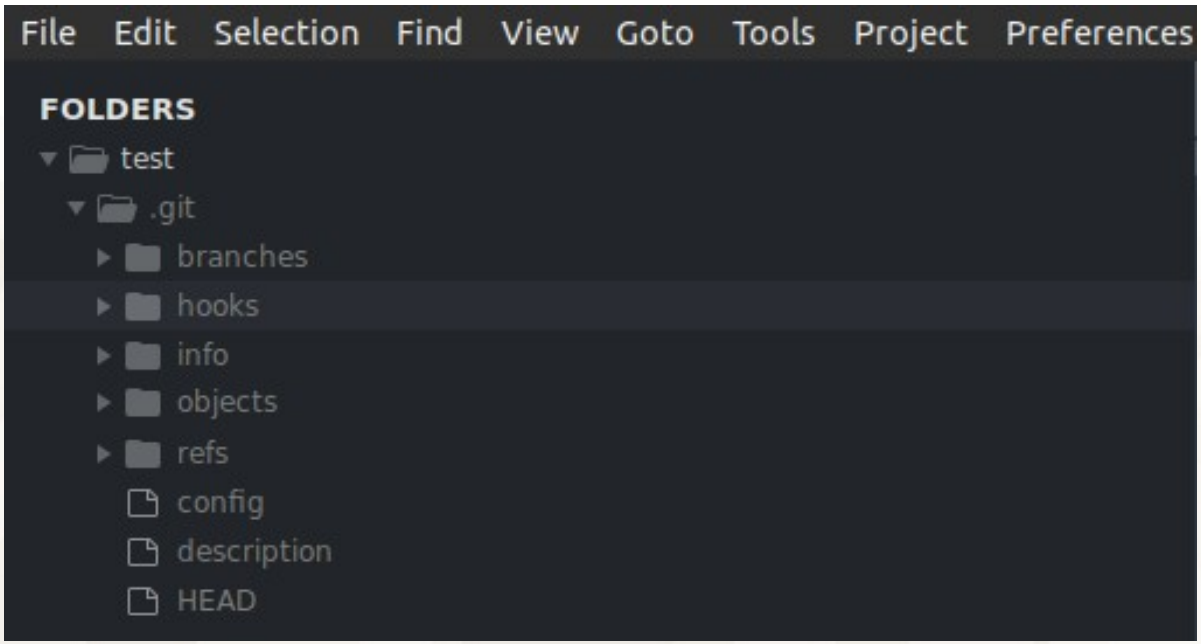
"Git init " to initialize a git repository in any folder

"Git clone" can also be used to import an existing repository

Git: Initialization



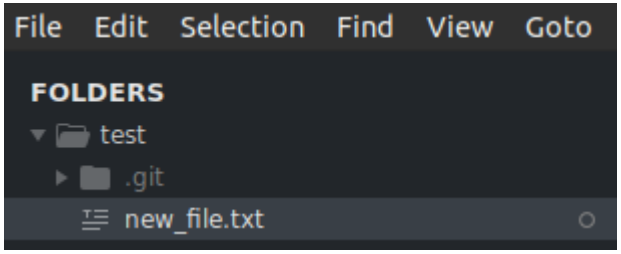




"Git init " to initialize a git repository in any folder

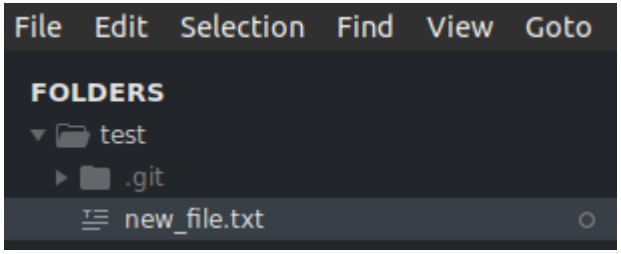
"Git clone" can also be used to import an existing repository

Git: Commit



Create new file in project

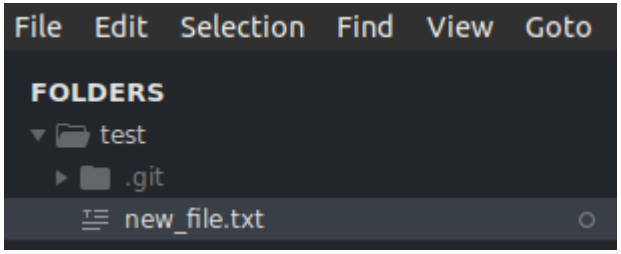
Git: Commit



Create new file in project

```
mateo_h@mateoH-Latitude-5420:~/formation_fair/test$ git add new_file.txt
mateo_h@mateoH-Latitude-5420:~/formation_fair/test$ git commit -m "first commit"
[master (commit racine) 59647f6] first commit
 1 file changed, 1 insertion(+)
 create mode 100644 new_file.txt
mateo_h@mateoH-Latitude-5420:~/formation_fair/test$
```

Git: Commit

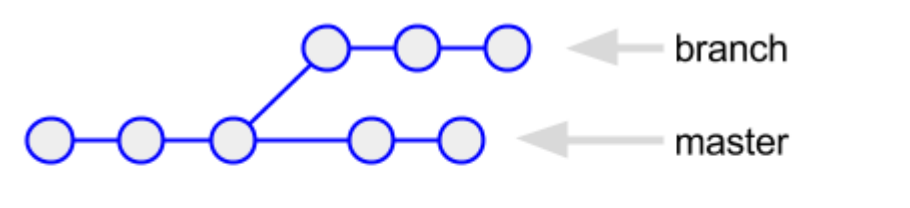


Create new file in project

```
mateo_h@mateoH-Latitude-5420:~/formation_fair/test$ git add new_file.txt
mateo_h@mateoH-Latitude-5420:~/formation_fair/test$ git commit -m "first commit"
[master (commit racine) 59647f6] first commit
 1 file changed, 1 insertion(+)
 create mode 100644 new_file.txt
mateo_h@mateoH-Latitude-5420:~/formation_fair/test$
```

```
mateo_h@mateoH-Latitude-5420:~/formation_fair/test$ git log
commit 59647f6ac1250276f91d7312b25e52c7974e616d (HEAD -> master)
Author: HIRIART <mateo.HIRIART@uca.fr>
Date:   Wed Nov 23 15:25:26 2022 +0100

    first commit
```



Commits are snapshot of your files (code)

They all have a unique id

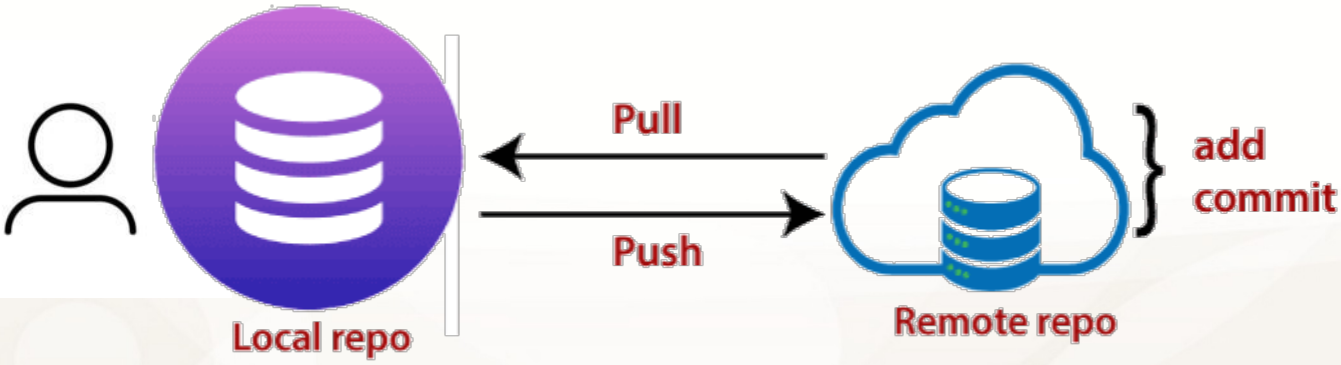
And are linked to their parent

Can always go back to previous commit

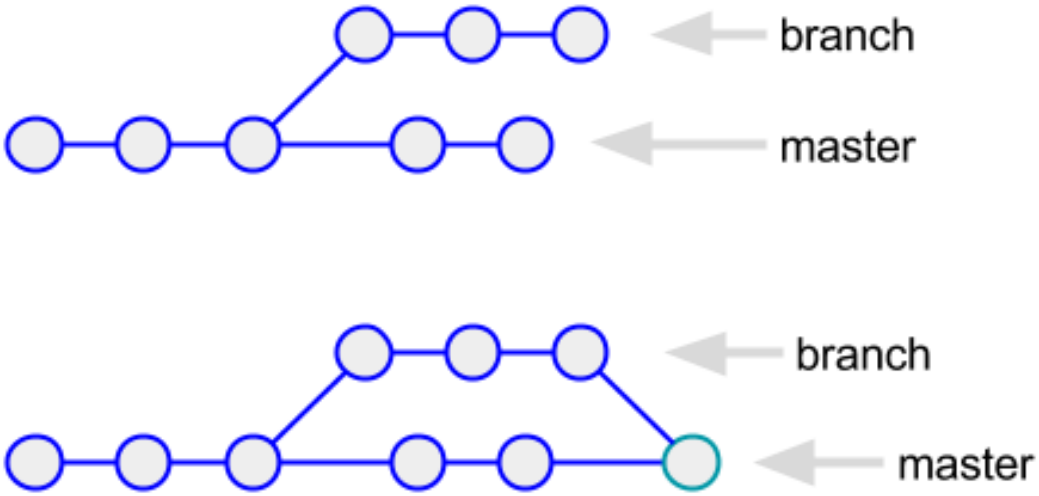
Git: Push and Pull

Git pull imports modification from another repository if it has the same origin.

Git push exports modification to another repository if it has same origin.



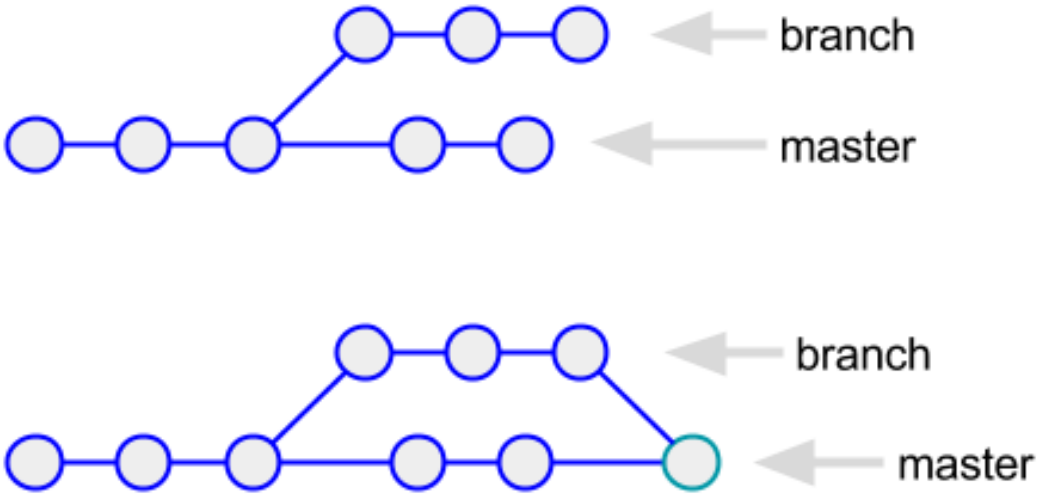
Git: branches



Git branch allows parallel development

And still applies it latter with code control

Git: branches



Git branch allows parallel development

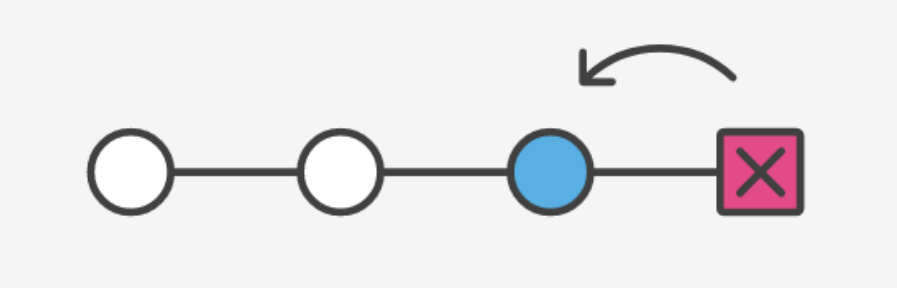
And still applies it latter with code control

```
mateo_h@mateoH-Latitude-5420:~/formation_fair/test$ git branch new_branch
mateo_h@mateoH-Latitude-5420:~/formation_fair/test$ git branch
* master
  new_branch
mateo_h@mateoH-Latitude-5420:~/formation_fair/test$ git switch new_branch
Basculement sur la branche 'new_branch'
mateo_h@mateoH-Latitude-5420:~/formation_fair/test$ git branch
  master
* new_branch
```

Git: Revert

Git reset: reset repository back to the specified HEAD

Git restore: reset current modification



```
mateo_h@mateoH-Latitude-5420:~/formation_fair/test$ git status
Sur la branche new_branch
Modifications qui ne seront pas validées :
  (utilisez "git add <fichier>..." pour mettre à jour ce qui sera validé)
  (utilisez "git restore <fichier>..." pour annuler les modifications dans le
répertoire de travail)
        modifié :         new_file.txt

aucune modification n'a été ajoutée à la validation (utilisez "git add" ou "gi
t commit -a")
```


UNIVERSITÉ Clermont Auvergne

Git: Revert

Git reset: reset repository back to the specified commit

Git restore: reset current modification

```
mateo_h@mateoH-Latitude-5420:~/formation_fair/test$ git add .
mateo_h@mateoH-Latitude-5420:~/formation_fair/test$ git commit -m "update my new_file"
[new_branch 8ad9fa7] update my new_file
 1 file changed, 3 insertions(+), 1 deletion(-)
```

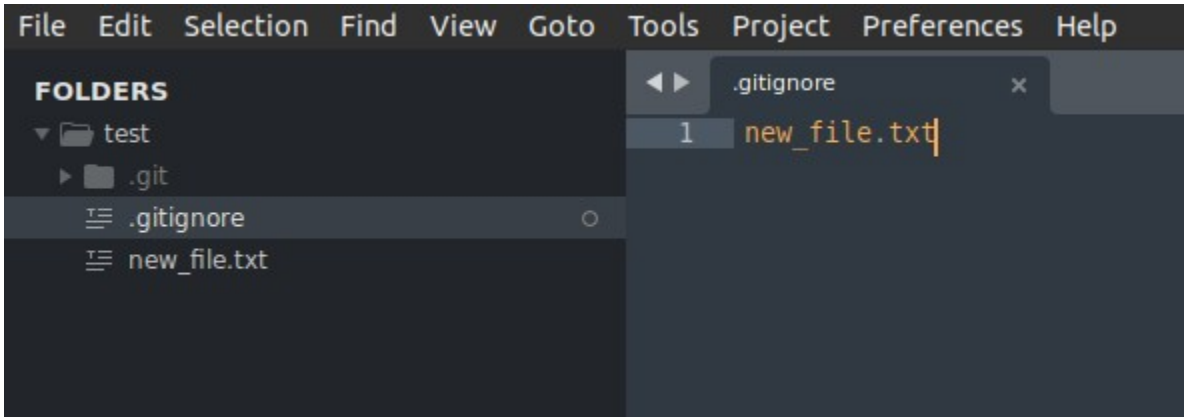
```
mateo_h@mateoH-Latitude-5420:~/formation_fair/test$ git log
commit 8ad9fa7e0c1c91a5d4469d2aa883a0d2052fc18d (HEAD -> new_branch)
Author: HIRIART <mateo.HIRIART@uca.fr>
Date:   Fri Nov 25 11:34:28 2022 +0100

    update my new_file

commit 59647f6ac1250276f91d7312b25e52c7974e616d (master)
Author: HIRIART <mateo.HIRIART@uca.fr>
Date:   Wed Nov 23 15:25:26 2022 +0100

    first commit
mateo_h@mateoH-Latitude-5420:~/formation_fair/test$ git reset --hard 59647f6ac1250276f9
1d7312b25e52c7974e616d
HEAD est maintenant à 59647f6 first commit
```

Git: .gitignore



Specific file that tells git to untrack any files listed inside.

Very useful especially with project that required other packages or if you have any file that stores sensitive data.

Git: Useful command

Git help: display multiple git command with a description

Git log: display commit history of the current branch

Git status: display current state off your local repository

Git diff: display code difference between current local repository and HEAD

Documentation link for git command:

https://git-scm.com/doc

https://docs.github.com/fr

Github and Gitlab

Public repository hosting server.

Both free version with pricing for enhance features.

Released in 2008

Owned by microsoft

Fast, more popular

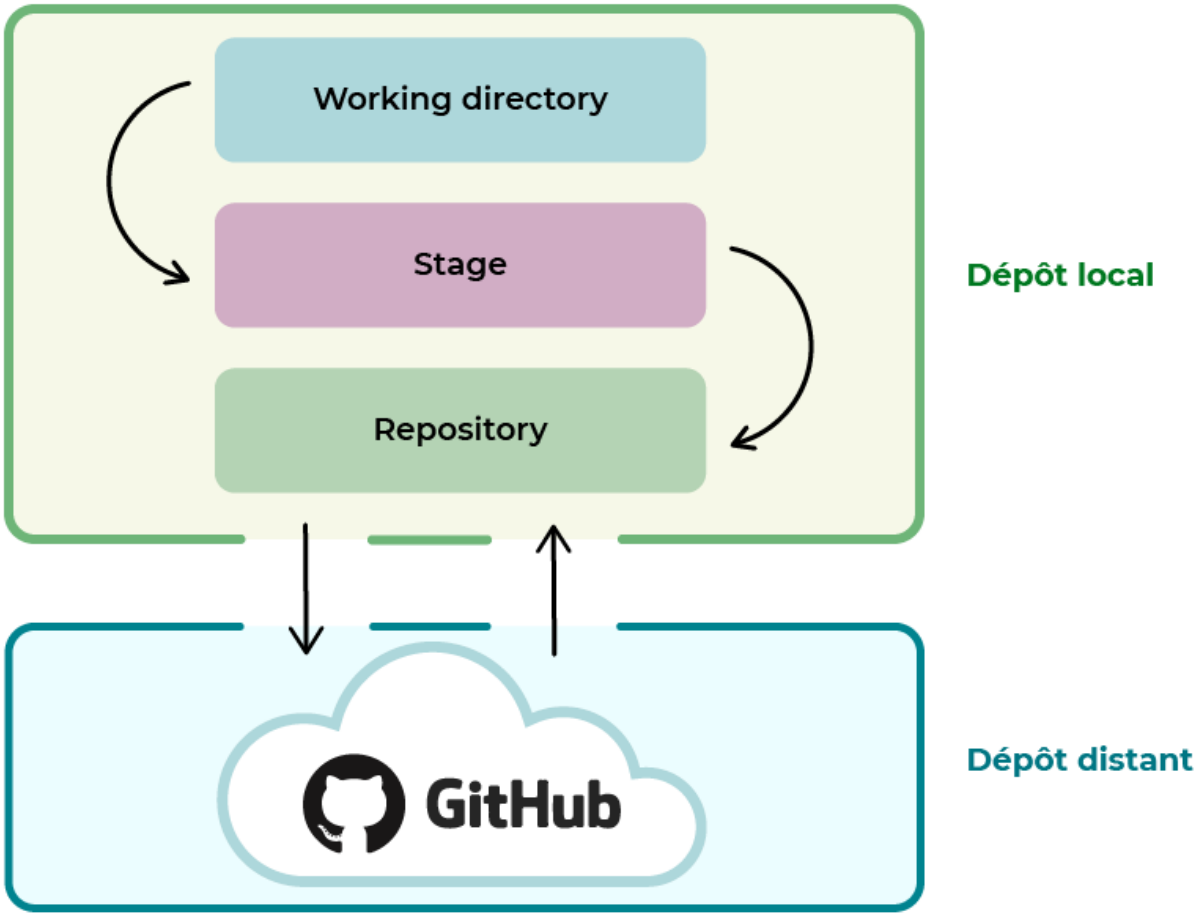
Released in 2014

Open source

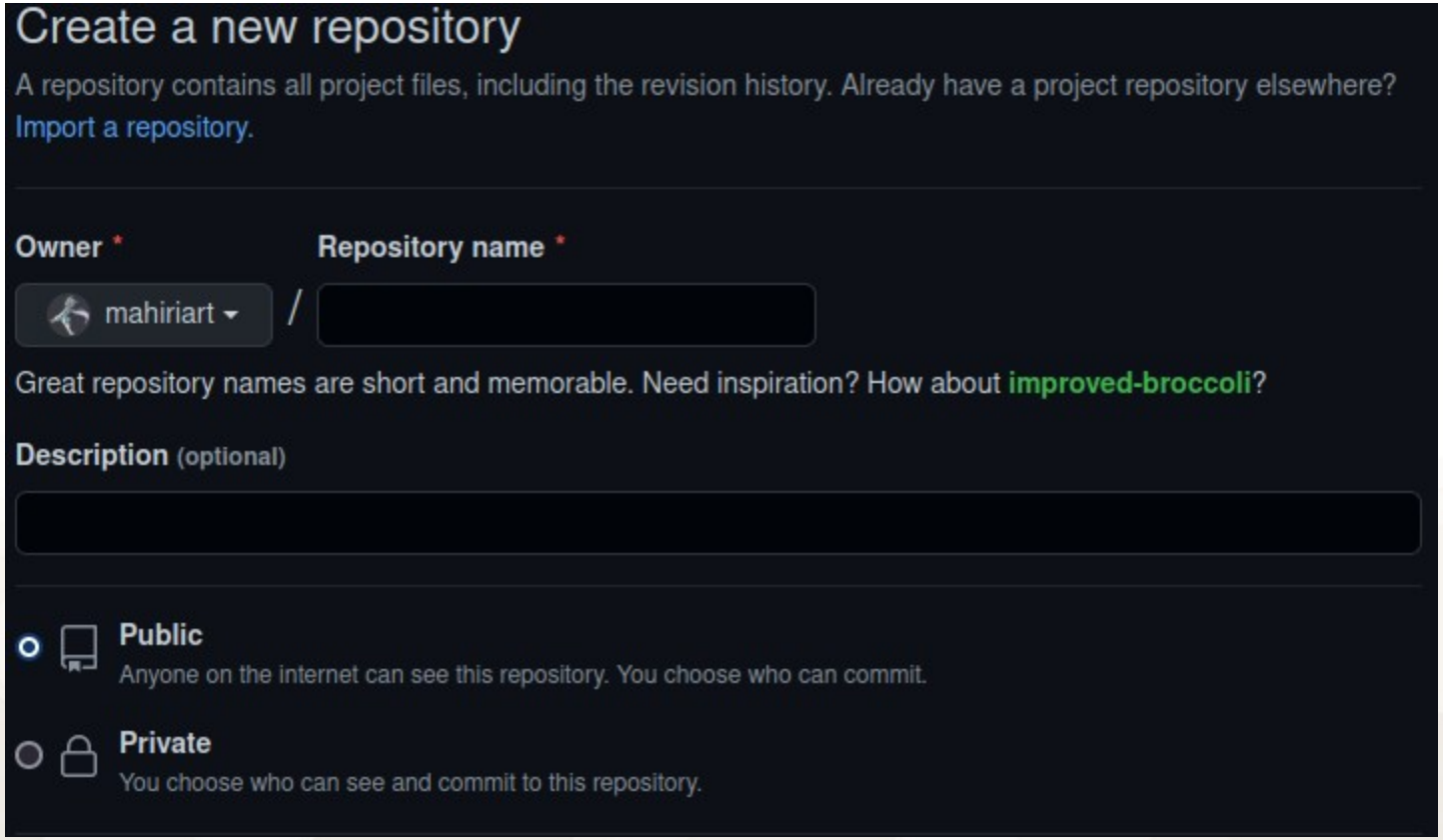
Built-in CI/CD, Less restriction in free version

UNIVERSITÉ Clermont Auvergne

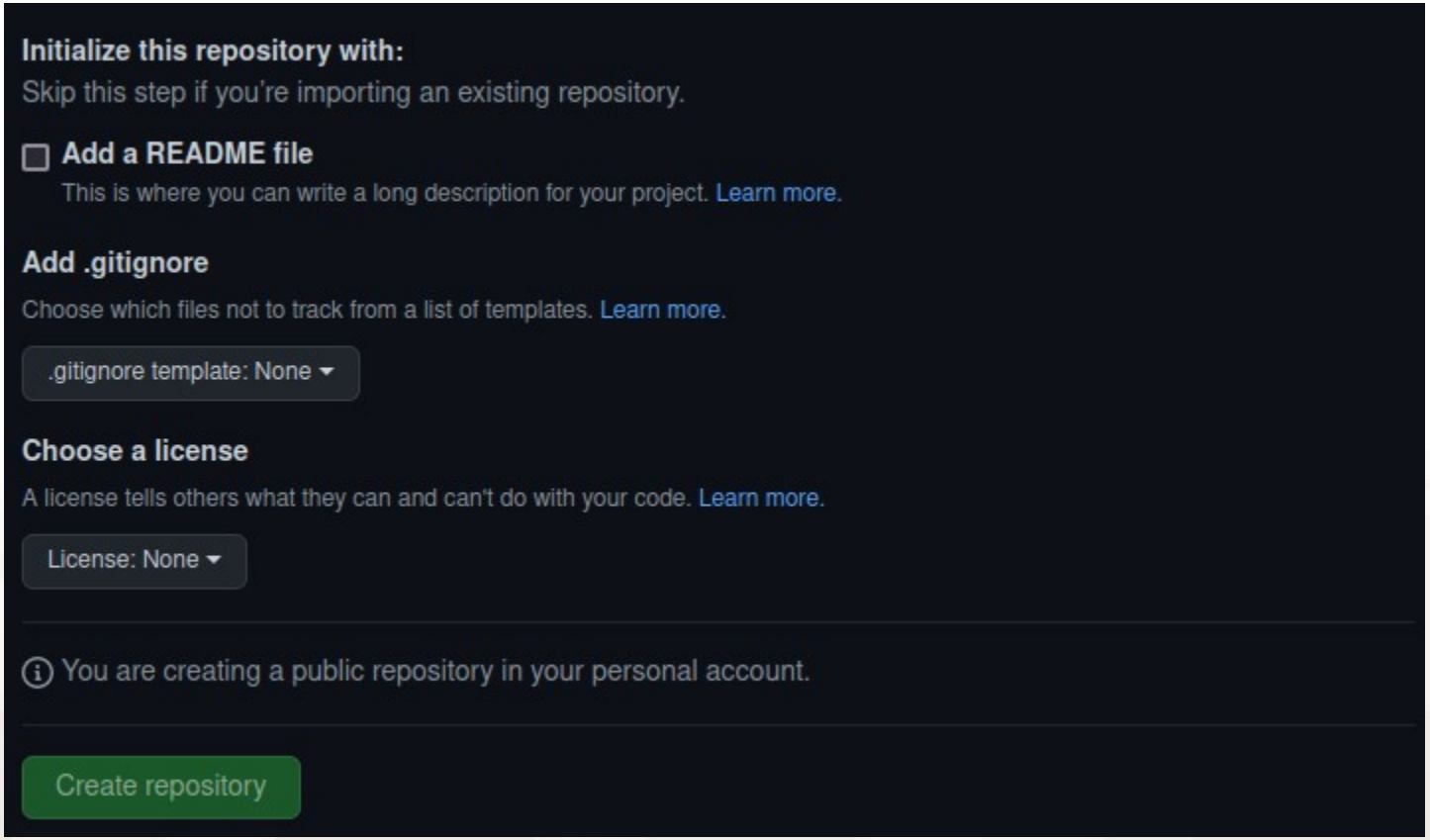
Github and Gitlab



Create your own repository



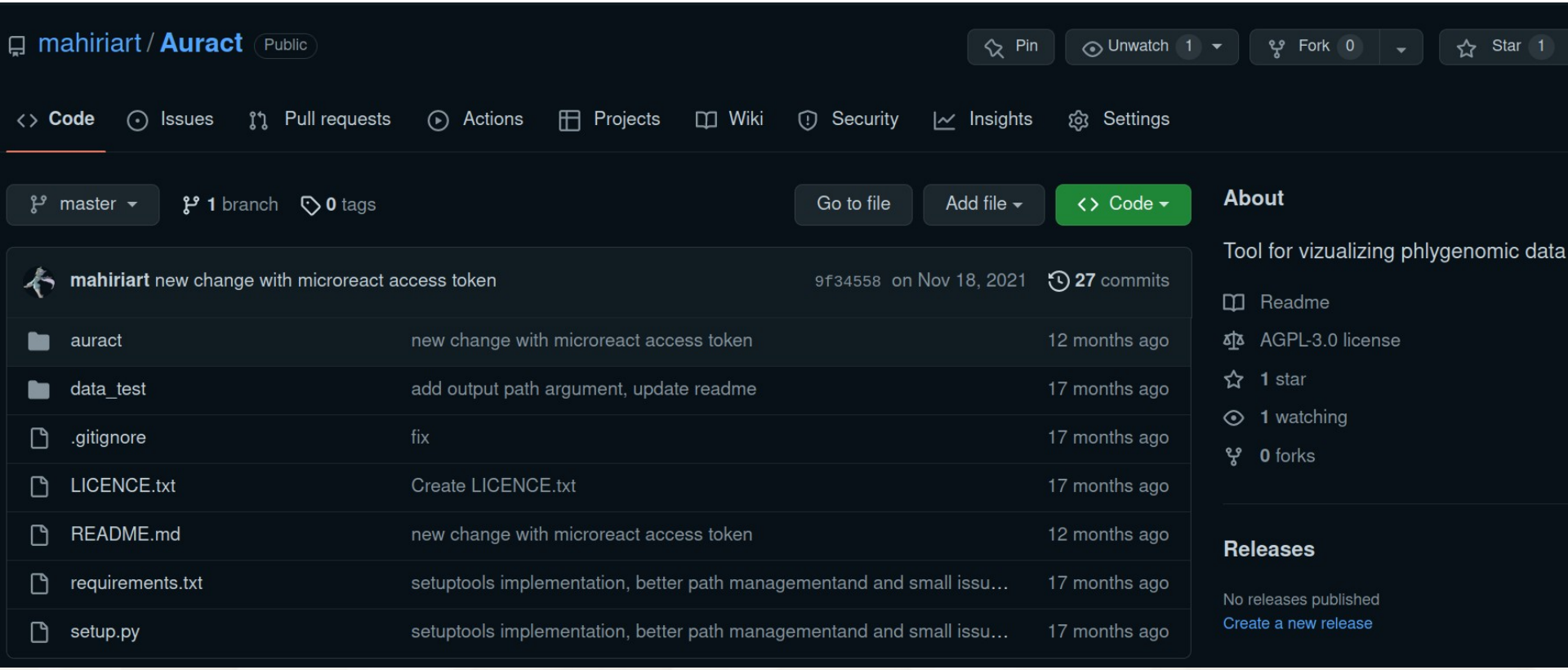
Create your own repository, with base features



UNIVERSITÉ Clermont Auvergne

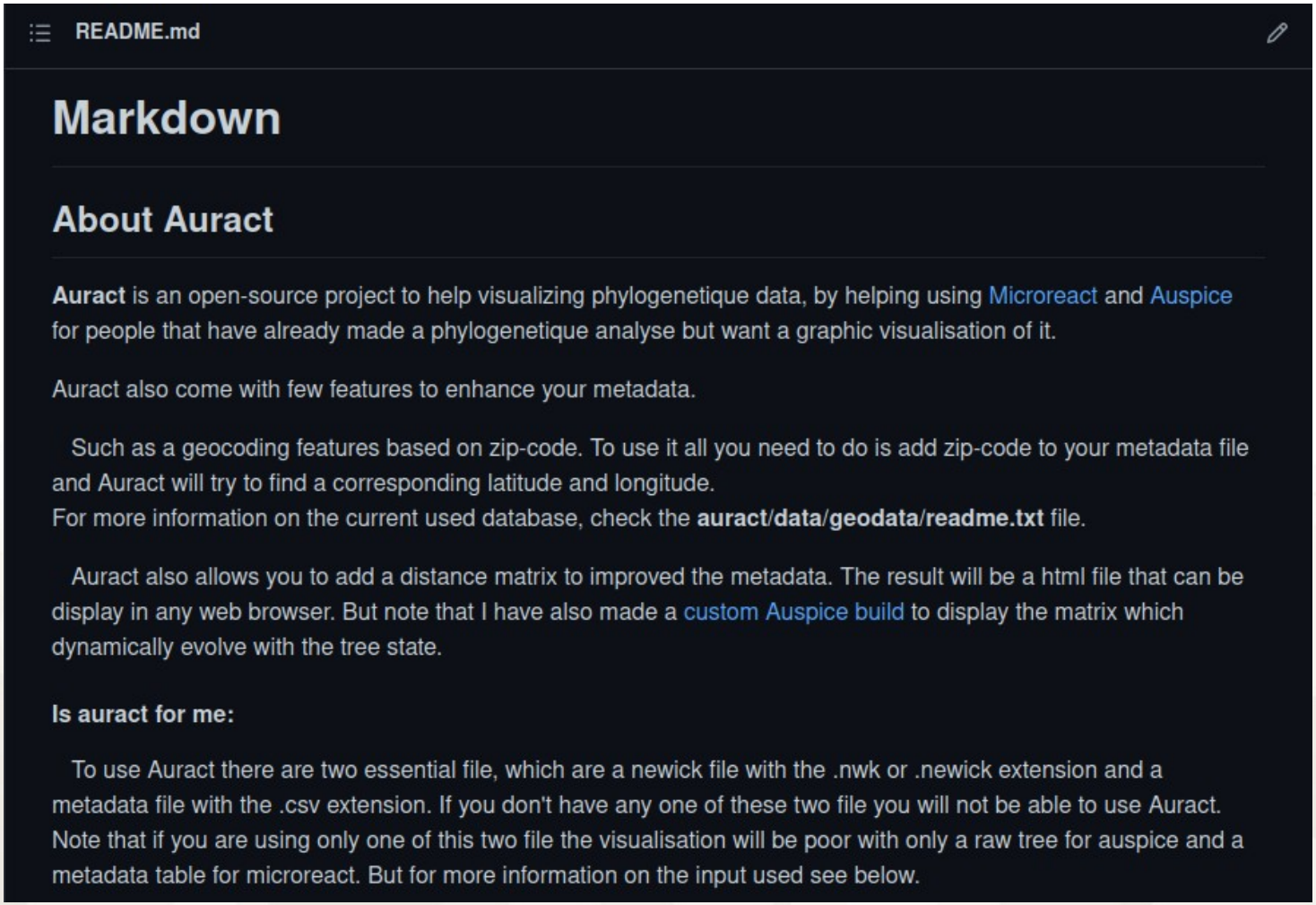
Github/Gitlab: Basics

Main page of a repository



Github/Gitlab: Improvement

Readme usually presents the repository and also has documentation or guide on installation and usage.


README.md

Markdown

About Auract

Auract is an open-source project to help visualizing phylogenetique data, by helping using Microreact and Auspice for people that have already made a phylogenetique analyse but want a graphic visualisation of it.

Auract also come with few features to enhance your metadata.

Such as a geocoding features based on zip-code. To use it all you need to do is add zip-code to your metadata file and Auract will try to find a corresponding latitude and longitude.
For more information on the current used database, check the **auract/data/geodata/readme.txt** file.

Auract also allows you to add a distance matrix to improved the metadata. The result will be a html file that can be display in any web browser. But note that I have also made a custom Auspice build to display the matrix which dynamically evolve with the tree state.

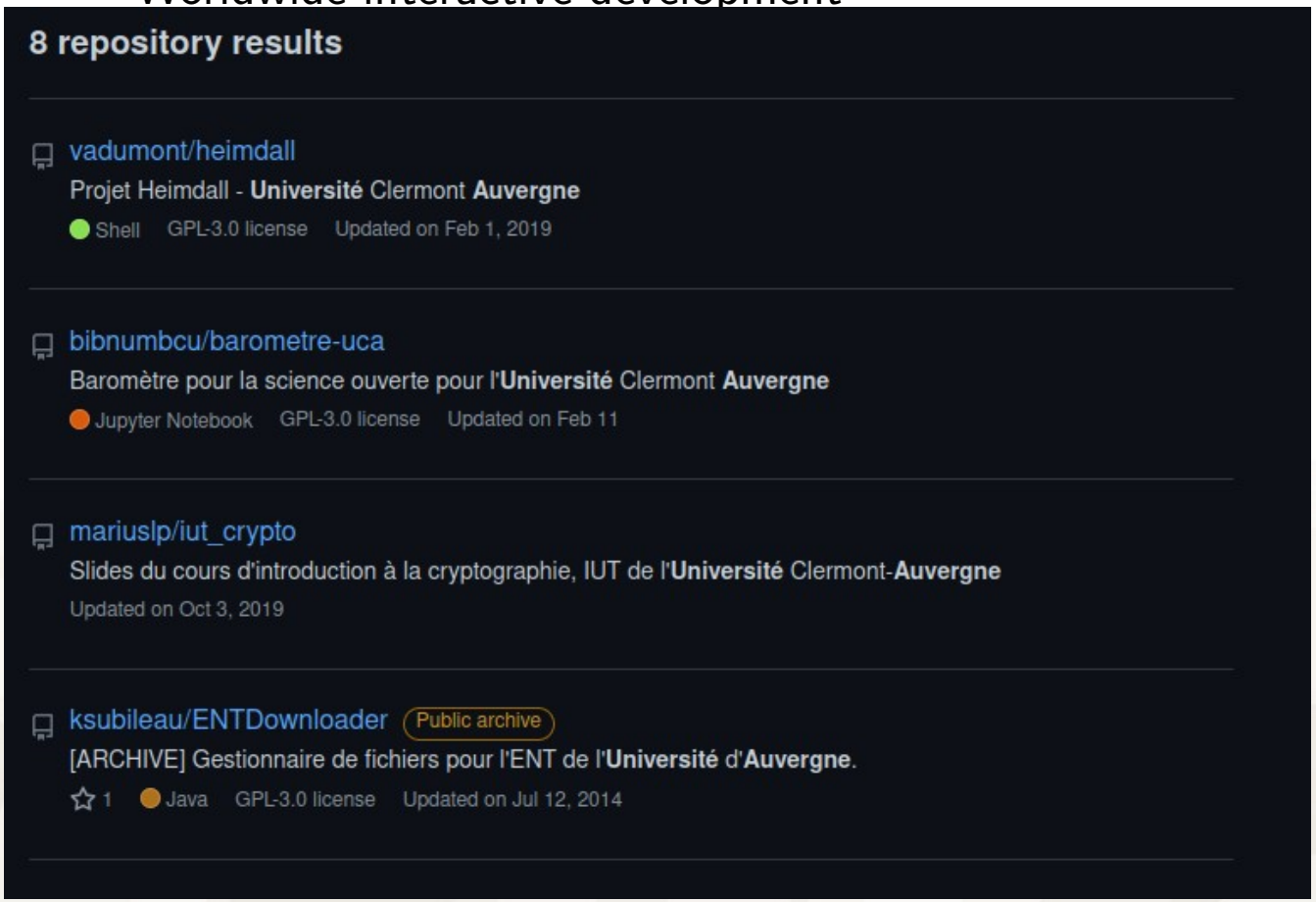
Is auract for me:

To use Auract there are two essential file, which are a newick file with the .nwk or .newick extension and a metadata file with the .csv extension. If you don't have any one of these two file you will not be able to use Auract. Note that if you are using only one of this two file the visualisation will be poor with only a raw tree for auspice and a metadata table for microreact. But for more information on the input used see below.

Github/Gitlab: Open source project

Find open source project and publish yours

Worldwide interactive development

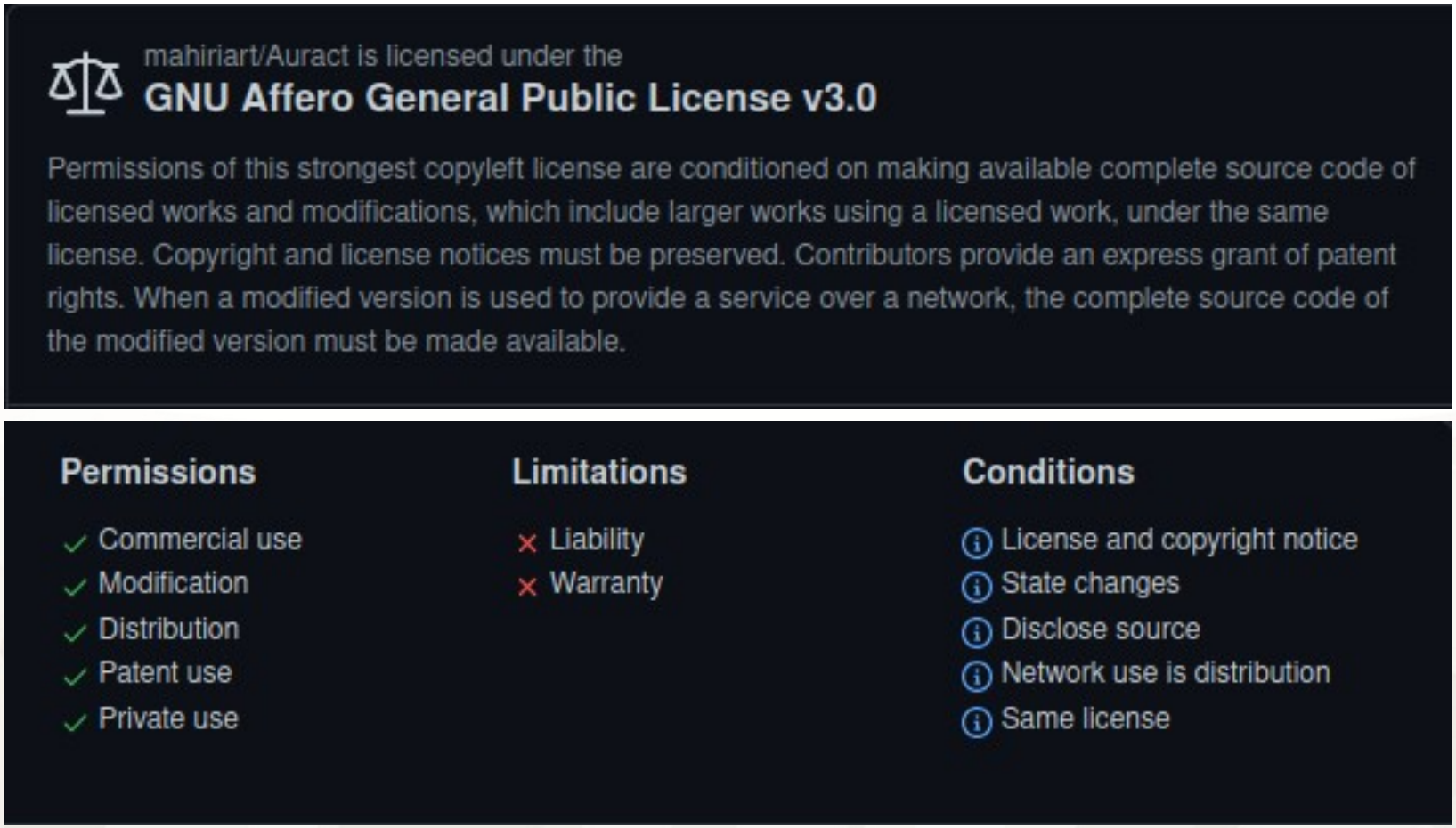


UNIVERSITÉ Clermont Auvergne

Github/Gitlab: Licenses

Key element for public repository, allow then to truly be open source.

Allow others to freely use, change and distribute your code.



Github/Gitlab: Licenses

Many licences available

Searching GitHub by license type

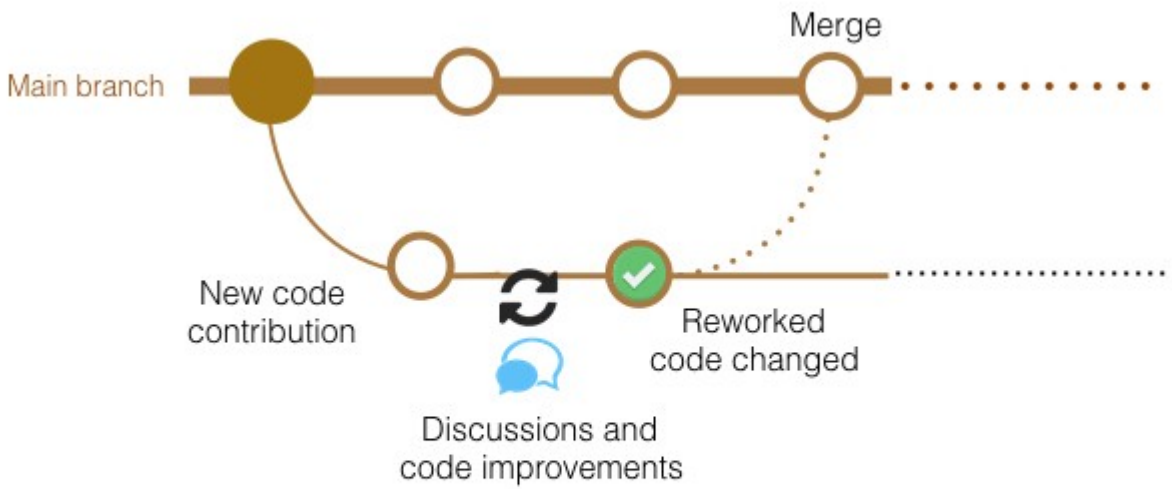
You can filter repositories based on their license or license family using the `license` qualifier and the exact license keyword:

License	License keyword
Academic Free License v3.0	`afl-3.0`
Apache license 2.0	`apache-2.0`
Artistic license 2.0	`artistic-2.0`
Boost Software License 1.0	`bsl-1.0`
BSD 2-clause "Simplified" license	`bsd-2-clause`
BSD 3-clause "New" or "Revised" license	`bsd-3-clause`
BSD 3-clause Clear license	`bsd-3-clause-clear`
Creative Commons license family	`cc`

Github/Gitlab: Issues and pull request

Issues are features to point out problem in code or possible enhancement. Usually solved in a pull request.

Pull requests are code contributions in a review state.



Simplified Pull Request process

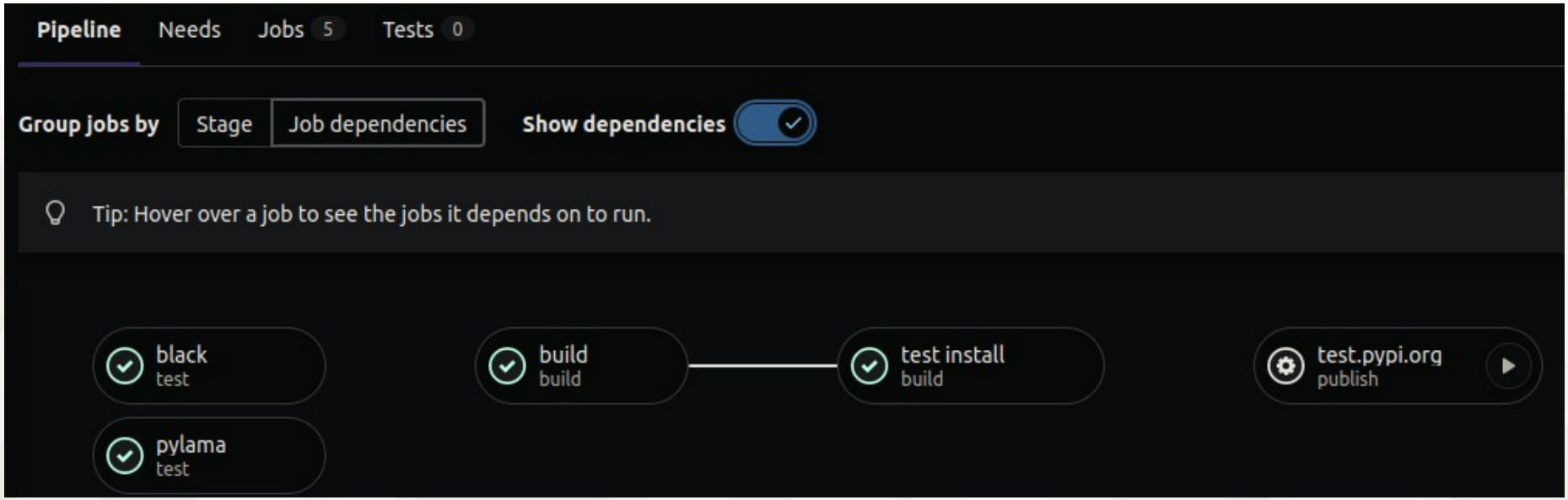
Github/Gitlab: Pipeline CI/CD

Continuous Integration (CI)

Continuous Delivery (CD)

Can automatically build, test, deploy, and monitor your applications

Example of CI pipeline



UNIVERSITÉ Clermont Auvergne

Github/Gitlab: Pages

Easily build own web site for your project. To highlight your work.

Hosted by Github or Gitlab.

Multiple templates can be found.

https://pages.gitlab.io/hugo/

End

https://github.com/mesocentre-clermont-auvergne/formation_fair_2022/tree/main/fair_git