Quick Guide to Github - GEFAN

Pedro Dedin Neto

What are Git and Github

- **Git** is a Version Control System (The system that you use to organize the versions of your code)

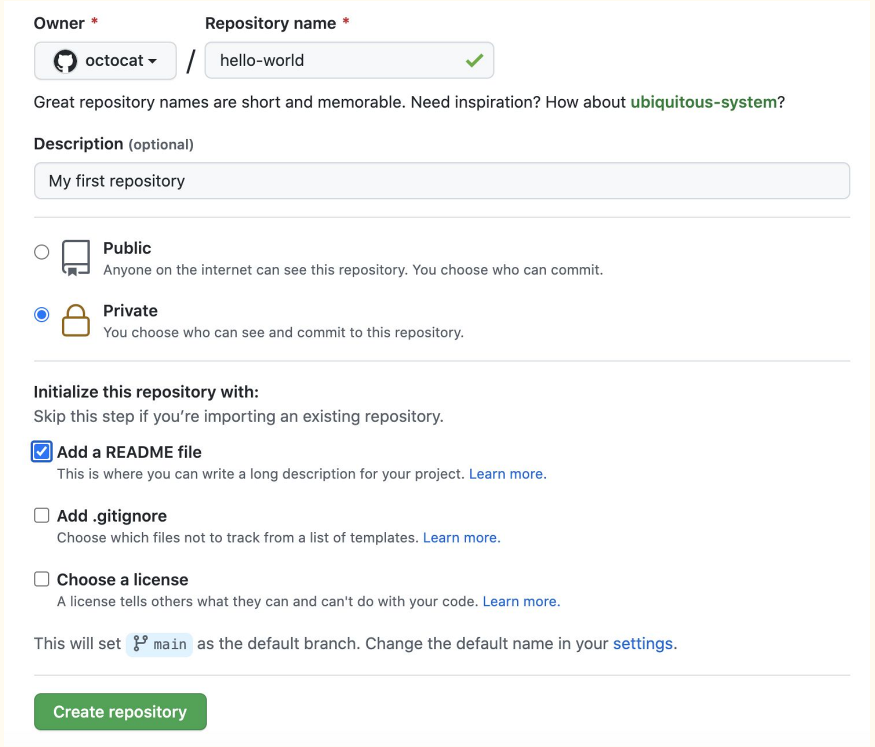
- **Github** is a specific provider of Internet hosting for software development and version control with Git (Where you can host your code, i.e., its repository)

What you need to use them?

- Install git locally on your computer (https://git-scm.com/)
- Create an account on Github (https://github.com/)

How to create a repository?

- You may create your repository in the Github and make it public or private.



Owner * Repository name *

octocat / hello-world

Great repository names are short and memorable. Need inspiration? How about ubiquitous-system?

Description (optional)

My first repository

( ) Public
Anyone on the internet can see this repository. You choose who can commit.

[✓ Private]
You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

[✓] Add a README file
This is where you can write a long description for your project. Learn more.

[ ] Add .gitignore
Choose which files not to track from a list of templates. Learn more.

[ ] Choose a license
A license tells others what they can and can't do with your code. Learn more.

This will set `main` as the default branch. Change the default name in your settings.

Cloning a repository

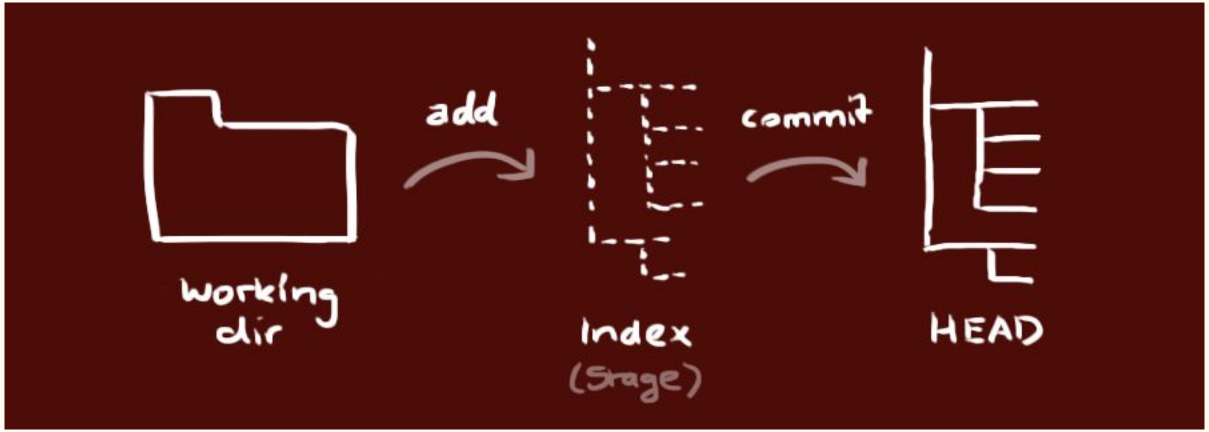
- You can clone (download) a repository using the following command:

```
git clone “repository link”
```

- It may request username and password if the repository is private.
- Now, you have the code locally on your computer.

Working Flow

Locally, you have 3 different “stages” of you code:



Add

- When you want to add files/modified files to your **stage,** you can use the command:

`git add <file>` or `git add *`

- Now, the files are ready to be committed.

Obs: You may use `git status` to view track which files have been added or not.

Commit

- When you want to commit the files/modified files to your **local** repository, you can use the command:

```
git commit -m "Comments about the changes"
```

- Now, all the committed files are updated in your local repository.
- However, you still didn't send it to your remote repository.

Push

- To send your commits to the **remote repository**, you can use the command:

```
git push origin
```

- Now your remote repository is up to date.

Obs: Some modifications may be needed in this command if you are working in another branch

Pull

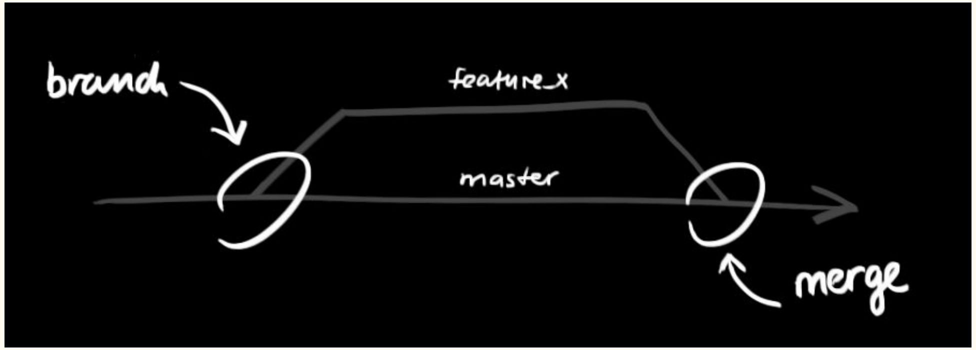
- To update your local branch, you may use the following command:

```
git pull
```

- It will **merge** your local files with the remote ones.

Branches

- You can create parallel versions of the code called “Branches” to work in features that may be implemented in the main code, what is called a “Merge”.
- Creating a Branch: `git checkout -b name_of _the_branch`
- Returning to the main branch: `git checkout master`
- To merge your current branch with another: `git merge <branch>`



Organizations

- “Organizations are shared accounts where businesses and open-source projects can collaborate across many projects at once. Owners and administrators can manage member access to the organization's data and projects with sophisticated security and administrative features.” (Github Docs)

- GEFAN’s Organization Account: https://github.com/GEFAN-Unicamp
- You can create repositories using the organization account or you can allow the account to import your repository to be hosted by it.

Useful Links

- Github Docs: https://docs.github.com/en
- git - guia prático: https://rogerdudler.github.io/git-guide/index.pt_BR.html