

# Git - Basics



This document presents only the fundamental notions of Git necessary for the development of the project. Learning Git is not the purpose of the Operating Systems course, but Git is a useful tool in software development because of its advantages.

## 1. What is Git?

Git is a **version control system** that is used for tracking changes in source code during software development. It allows developers to easily collaborate on projects, keep track of different versions of their code, and roll back to previous versions if needed. Git also makes it easy for developers to share their code with others.

You can install Git following the steps from [the official site](#). You can learn more about Git from the manual pages accessible via the man interface: **man git**

In order to be able to track your changes in the logs, across time, you need to provide some basic information about yourself in the Git's configuration. In order to configure Git you can use the following commands:

```
git config --global user.name "Name Surname"
```

```
git config --global user.email name.surname@domain.com
```

## 2. What is a repository?

A repository is a central location where all the files, folders, and other assets related to a project are stored. The repository also stores information about who made what changes and when they were made.

## 3. What is GitHub

GitHub is a web-based hosting service for version control using Git. It offers all of the distributed version control and source code management (SCM) functionality of Git as well as adding its own features.

If you want to create an account on GitHub, you can follow this steps:

1. Go to the [GitHub website](#) and click on the "Sign Up" button.
2. Enter your email address, create a username, and choose a password.
3. Click on the "Create an account" button.
4. You will then be asked to verify your email address by clicking on a link sent to you in an email from GitHub.

If you want to create a new repository on GitHub, you can follow this steps:

1. Click the + button and then select New repository.
2. Type a name for your repository.

3. Choose either to make the repository public or private. Public repositories are visible to the public, while private repositories are only accessible to you, and people you share them with.
4. Click Create repository.

#### 4. Cloning a repository (git clone command)

Cloning a repository means copying the repository from a remote source (such as GitHub) to your local computer. This allows you to have a local copy of the repository, which you can then use to make changes and commit them back to the remote source.

Command: **git clone <repository\_url>**

#### 5. Staging area (git add command)

The **staging area** is an intermediate area in a version control system where changes to files are tracked before they are committed to the repository. It allows users to review their changes before they are committed and make sure that only the desired changes are included in the commit.

Command: **git add <file1> <file2> <file3> etc.**

If you want to add all new and modified files in the current directory and its subdirectories to the staging area you can use **git add .**

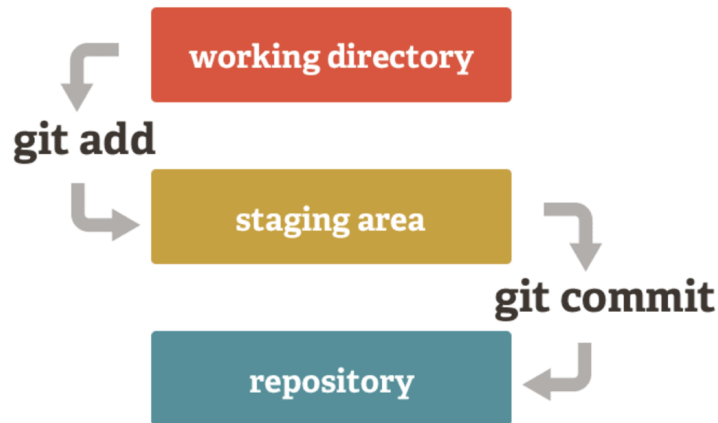
!!! It is important to note that Git tracks content, not files, so if the files added to the staging area will be further modified afterwards, the new changes will not be added to the index by default. On the contrary, they will have to be added explicitly, via a new git add command.

#### 6. Making a commit (git commit command)

In version control systems, a commit is a record of changes made to a file or set of files. A commit usually contains a comment that describes the changes made, and it is usually associated with an author's name and timestamp.

Command: **git commit -m <short\_comment\_about\_the\_changes>**

The command simply records the performed changes to the repository. **Only the changes already added to the staging area are considered when the commit operation is performed. (See the picture below)**



## 7. Push your changes (git push command)

In version control systems (VCS), "push" is the action of sending changes from a local repository to a remote repository. This command is useful to push local commits to the remote repository.

Command: **git push origin master** (Here it would be git push origin <branch\_name>, but for this project you can use only the master branch)

## 8. Incorporating remote changes (git pull command)

The pull command is used when we want to both retrieve and incorporate the changes from a remote repository into our current branch. For our remote repository and our master branch, we can perform the pull operation as following:

Command: **git pull origin master**

## Other resources

As we said in the beginning of the document, we have only presented the fundamental notions that you need in order to implement the project. If you want to learn more about Git (which we strongly recommend) you can see the following resources:

- [Pro Git, Scott Chacon, Ben Straub , 2nd Edition](#)
- [An Introduction to Git and GitHub](#)
- [Git commands](#)

Videos:

- [Git basics](#)
- [GitHub basics](#)