SE489 DevOps Engineering

Lab 1

# Lab 1: Version Control Systems

Objective: After successfully completing this lab session, student will be able to understand, distinguish between Git and Github. This lab will train student how to download, customize, install and configure Git, lab further introduces students with basic git commands and operations.

## Git

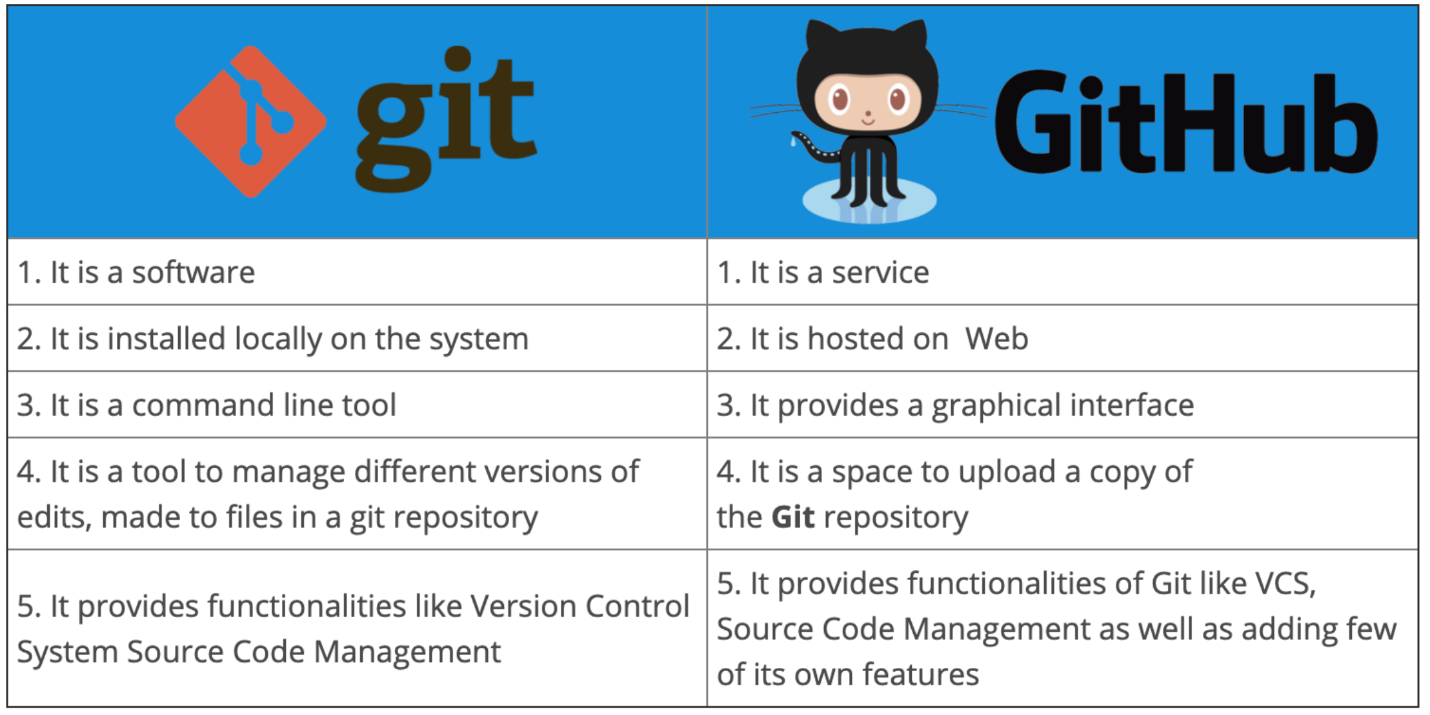
### Definition1:

Git is software for tracking changes in any set of files, usually used for coordinating work among programmers collaboratively developing source code during software development. Its goals include speed, data integrity, and support for distributed, non-linear workflows. – Wikipedia

Definition2:

Git is a DevOps tool used for source code management.

It’s equally interesting to note that *Linus Torvald* himself created git in 2005 for maintaining the code of Linux Kernel.

As opposed to the common perception, git and github are two different entities, following tabulated figure shows major difference between the two

But both of them work in tandem to achive code maintenance, as shown in the following figure-



Before you install git on your system, make sure

1. You have administrative privileges on the system, you want to install git.
2. There is enough space in the hard disk of the system

To install Git at your system, there are few steps which are to be followed

Steps shown below are for windows 11, but they will work on almost all versions of windows with minimal changes

***Step1: Download git***

Download, latest *standalone* version of the git from its website. (<https://git-scm.com/download/win>), at the time of this manual, it was 2.36.1Graphical user interface, text, application

Description automatically generated

Application

Description automatically generated with medium confidence

***Step 2: Run Installer***

Right click on the installer just downloaded and select, Run as Administrator

Graphical user interface, text, application

Description automatically generated

Click on next

Graphical user interface, text, application, email

Description automatically generated

Approve or change the default location

Graphical user interface, text, application, email

Description automatically generated

Click on next to accept the default selections or change some additional one, as I have selected additional icons

Graphical user interface, text, application, email

Description automatically generated

Select start menu folder

Graphical user interface, text, application, email

Description automatically generated

***Step 3: Select default text editor***

Text editors are programs used for editing text files, by default git shows Vim as default text editor, but it is for linux systems, select Notepad or Notepad++ as per your choice (Notepad++ needs to be installed separately whereas Notepad is supplied with windows)

Graphical user interface, text, application

Description automatically generated

***Step 4: Deciding name of the Initial branch in the repository***

By default, it is “master” leave it as it is unless you want to change it

Graphical user interface, text, application, email

Description automatically generated

***Step 5: Adjusting PATH environments***

Graphical user interface, text, application, email

Description automatically generated

***Step 6: Selecting SSH executable for the git and repository***

Graphical user interface, text, application, email

Description automatically generated

***Step 7: Selecting HTTPS transport backend***

Graphical user interface, text, application, email

Description automatically generated

***Step 8: Configuring the line ending conversions***

Graphical user interface, text, application, email

Description automatically generated

***Step 9: Configuring the terminal emulator***

Graphical user interface, text, application, email

Description automatically generated

***Step 10: Chose the default behavior of ‘git’***

Graphical user interface, text, application, email

Description automatically generated

***Step 11: Choose credential manger***

Graphical user interface, text, application, email

Description automatically generated

***Graphical user interface, text, application, email

Description automatically generatedStep 12: Configuring extra options***

***Step 13: Configuring experimental options***

Leave them unchecked, unless you want to use them

Graphical user interface, text, application, email

Description automatically generated

***Step 14: Click on Install, and let the installation be complete***

Graphical user interface, text, application, email

Description automatically generated

***Step 15: Click finish to complete the setup***

Graphical user interface, text, application

Description automatically generated

This completes the installation of the Git. Git can be started in command line interface as well as in GUI mode, we will explain here, both.

Launching Git in Command Line Interface

Click on Windows start menu and type git bash and press enter, alternatively, you may also click on the icon appearing.

Graphical user interface, application

Description automatically generated

Launching Git in Graphical User Interface

To launch git in GUI, write at Windows Start menu, git gui and press *Enter*, or alternatively, click at git gui icon if shown.

Graphical user interface, application, Teams

Description automatically generated

Working with the Local Repositories:

1. Setting a Local Repository

Local Repositories 🡪 Local working directory/folder

\*\*when git bash is called upon from start menu, it starts at **c:\Users\<username>**, however if you want to create repository at some other location, there is a work around.

Navigate to the folder you want to make local repository with windows explorer, and once at the required location, at the Address bar, write cmd and press Enter, command window will open at the same location. In our case it is *d:\DevApp Tools\Lab Manual.*

When Command Window appears, write **git init** and press *Enter*

Git make current folder, local repository, you will see a folder with **.git** name appears in the current folder, this means that you have successfully made current folder as local repository in git.

Text

Description automatically generated

Now create a sample java program file with three print statements and save it.

A screenshot of a computer

Description automatically generated with medium confidence

1. Configuring git with username and user email

Before files and edits can be made and maintained, we need to configure git with username and user email id, so as git can track records precisely, what changes have been made by whom.

Although, individual repositories can be configured separately, it is wise to make configuration global if single user is going to use them all.

**git configure --global user.name <username>**

and

**git configure -- global user.email <user email>**

**Text

Description automatically generated**

1. Adding files to the staging area

To control the versioning of the file, git has provisions of ***Local repository 🡪Staging Area🡪Global Repository***

Now let’s push this file (DEMO.java) to the **Staging area**.

At command window write **git add .**

Some messages like below will appear

Text

Description automatically generated

1. Checking status of the repository (git status)

To check the status of the repository, **git status** is the command

Text

Description automatically generated

Obviously, from the output no commits have been performed, and all files are in the staging area.

1. Performing Commits (git commits)

To make changes permanent, we commit these files, commands used for this is

**git commit -m “<marker message>”**

**marker message** 🡪 these are the messages we write to easily identify different versions of the files, if we don’t provide messages here, git will ask for one later with popups, a commit without message will be aborted.

Text

Description automatically generated

1. Now check status again (git status)

It will show following output, master is the default name of the main branch of the project development

Graphical user interface

Description automatically generated with medium confidence