GIT

Git :-
Git is a free, open source version control System software. It was created by Linus Torvalds in 2005.

Git is a distributed version-control system for tracking changes in any set of files, originally designed for coordinating work among programmers cooperating on Source code during software development. Its goals included speed, data integrity and support for distributed, non-linear workflows.

There are many version control Systems then Why Git is most popular?

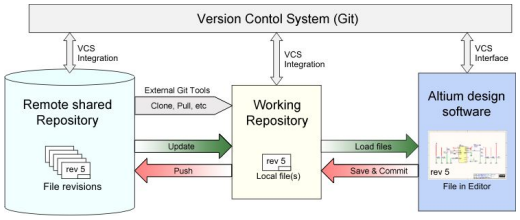
1. Free and Open Source
2. Distributed so that connectivity doesn't block work
3. Easy so that learning its commands can happen progressively
4. Team-Centric so that collaboration happens naturally
5. Collaborative History Tracking
6. Flexible so that it fits your need instead of the other way around.

Installation:

- Link for download git - https://git-scm.com/downloads

- After completing installation - Configure your username and email

How Git Works?



Working directory, staging area, and local repo:
Every local repository has three different virtual zones. These are:
Working directory
staging area
commit area.
The working directory is where new files are created, old files are deleted, or where changes are made to already existing files.
Once changes are made, they are added to the staging area. The staging area is also sometimes called the index.

Working Directory ----> Staging Area ----> Repository

Once the changes are complete, the staging area will contain one or more files that need to be committed. Creating a commit will cause Git to take the new code from the staging area and make the commit to the main repository. This commit is then moved to the commit area.

Getting Started:

- To get started with Git, go to your terminal and run the following command in your project directory. This initializes a project directory.

git init

- Run the following command to add files for Git to track. This will add these files to the staging area.

git add <filename_one>

- Run the following command to commit your changes to these files.

git commit -m "<add a commit message here>"

- We can push our changes through once we're done.

git push

Making any more changes in the master branch will require these changes to be committed again.

Why Git Version Control System?

1. Git focuses on content, not files
2. Opt in, not opt-out
3. Open, not locked
4. Distributed, not centralized
5. Conversations, not cutoffs
6. People, not tools
7. Journal, not backup
8. Anywhere, not just online