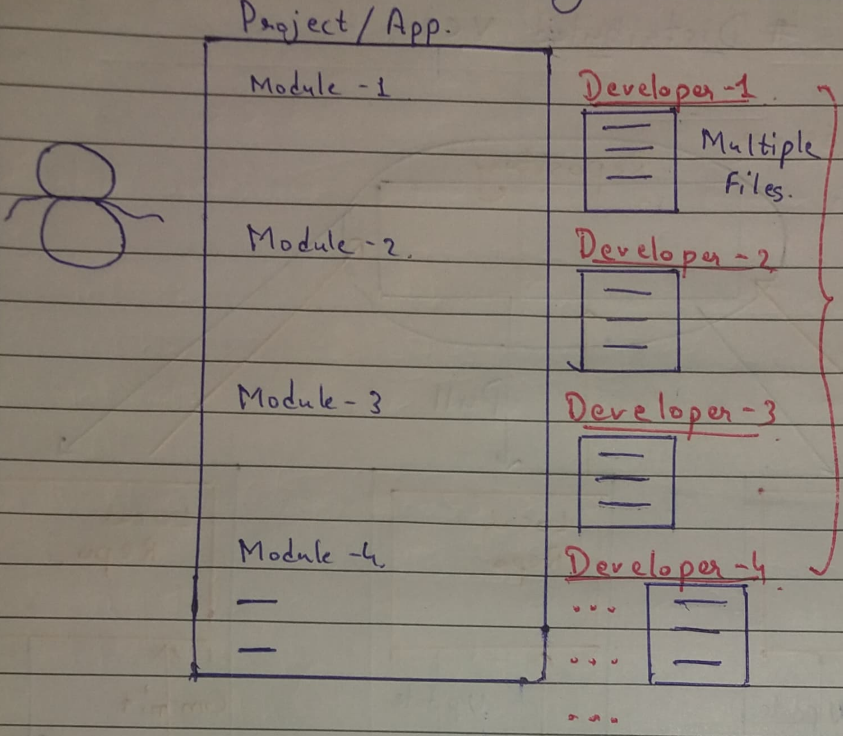
Version Control System. - Git.



→ Code Tracking.
→ Collaborative Environment.

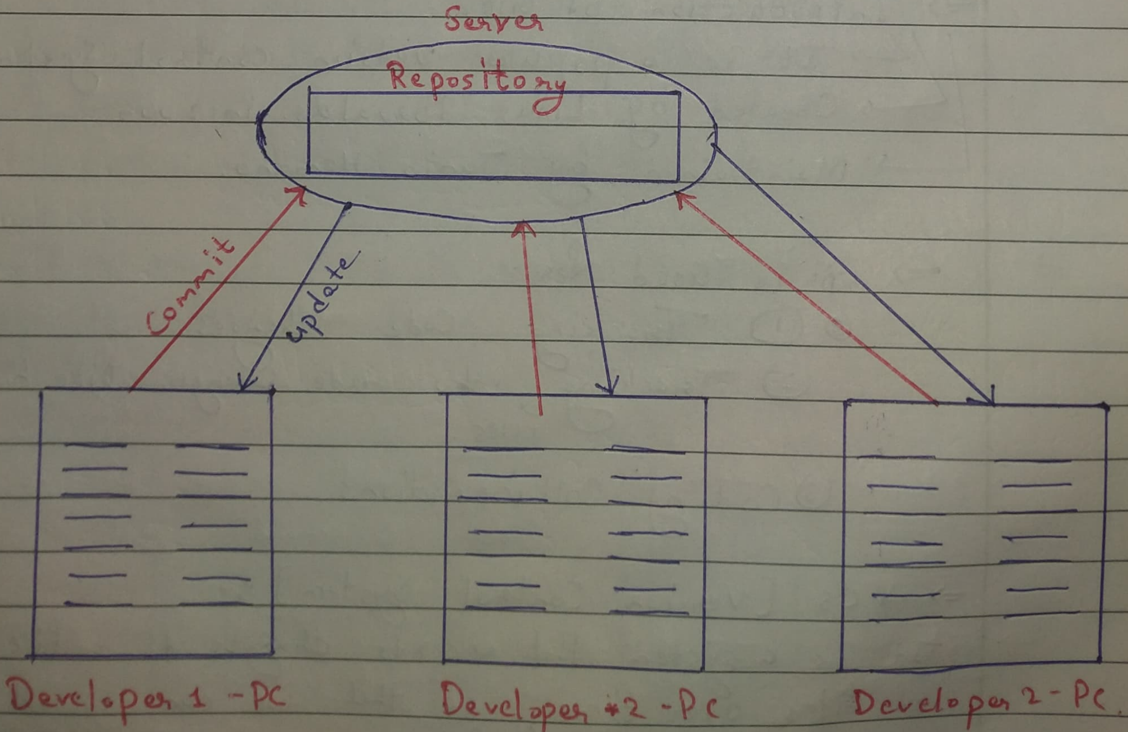
Version-1 + Changes

Version 2.
- 3
- 4
- 5
- 6
→ New Changes + Version-2.

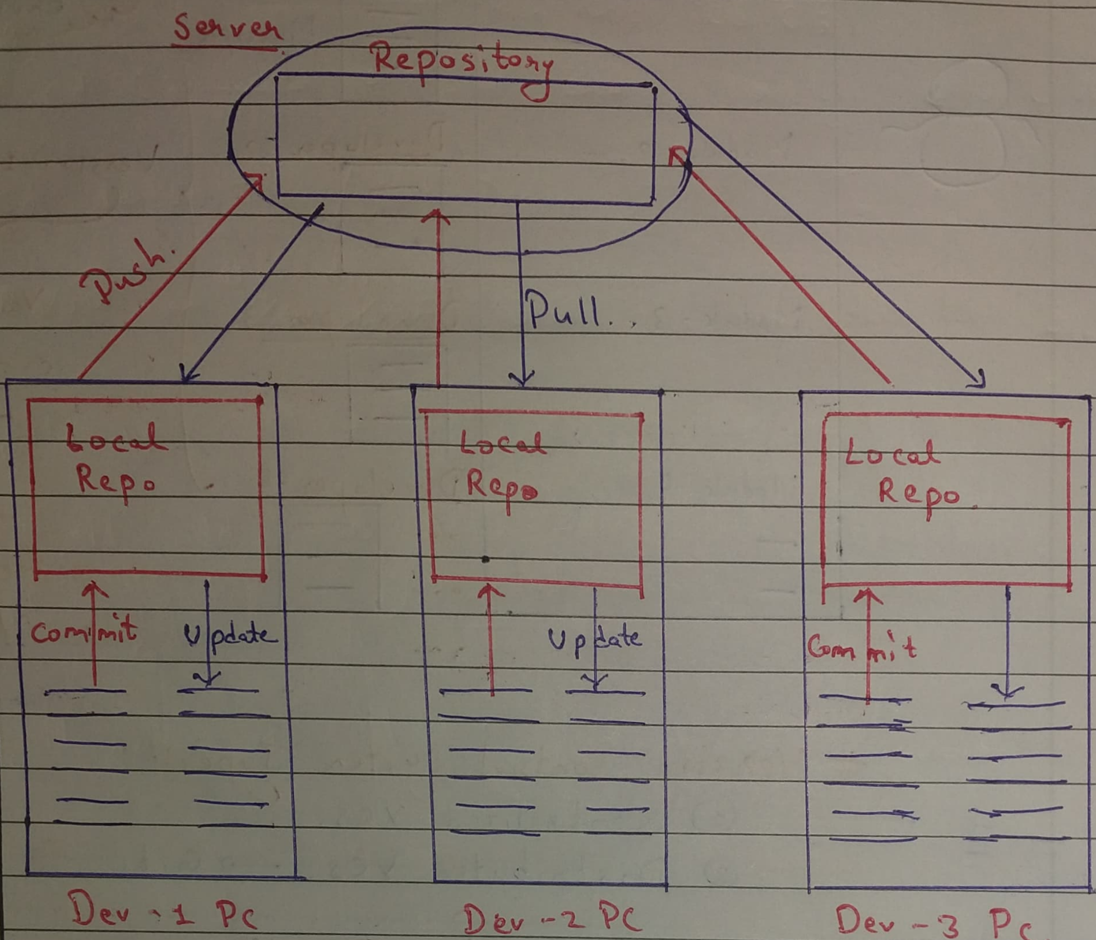
Version Control system Types!.

1. Centralized VCS.
2. Distributed VCS. → Git.

Centralized VCS



Distributed VCS.



=> Introduction of Git.

- -> It is a popular version control system [VCS],
- -> Created by Linus Torvalds. in 2005.
- -> Maintained by Junio Hamano.

=> Git is used for,

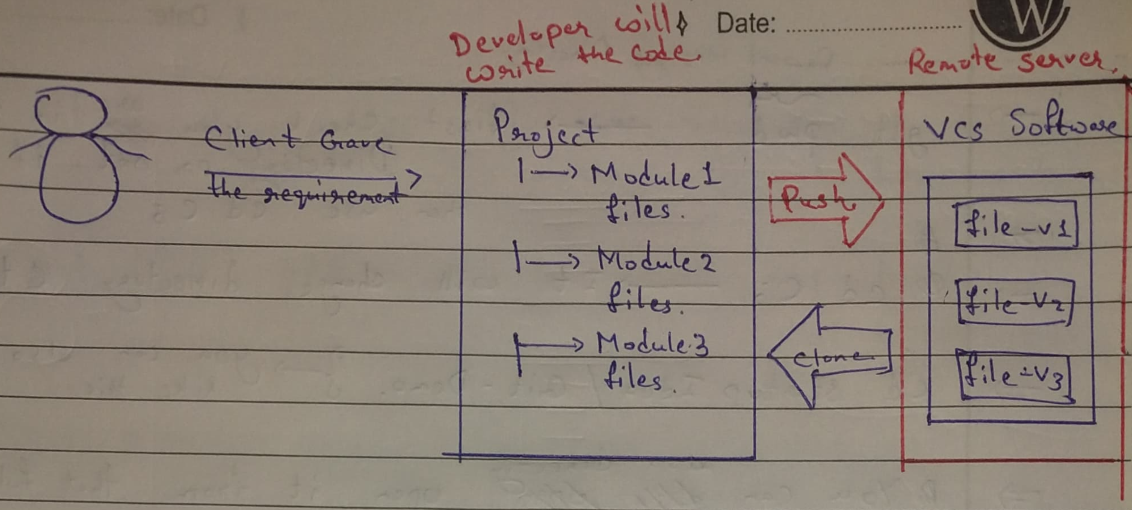
① Tracking Code Changes.

② Tracking who made Changes like history of the files.

③ Coding Collaborations

=> VCS [Version Control System]. :-

It is a system that records changes to a file or set of files over time, so that we can recall specific versions later, i.e, for every source code changes in a file a new version will ~~[illegible]~~ be created



=> Centralized [VCS] ——> Developers can Collaborate & do the changes.
↳ Eg:- CVS, SubVersion, Perforce

=> Distributed Version Control System.

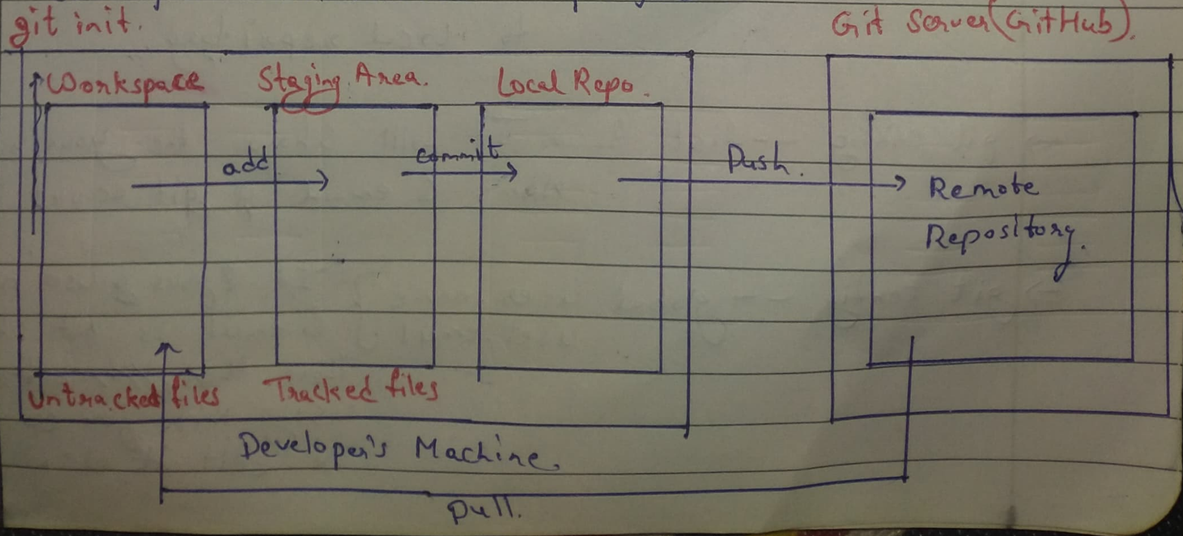
—> DVCS are Git, Mercurial, Darcs, Bazaar, etc

—> Developers don't check out the latest snapshot of the files rather they fully mirror the repository including its full history.

—> if the main server dies, then the local System will maintain a copy of the main repository which has full backup of data.

—> If the remote repository is down, then the developer can make changes in the local repository & when the main repoitory is up the code can be pushed to remote Repository from local repository

=> In Industry Jira tool is used to assign a particular task to developer. in project.



cd => Change directory

Present working Directory.

=> git pwd ——> first check you ~~go~~ are in wri~~gh~~te Directory or not. if not then use cd C:

=> ~~cd~~ cd C: ——> It will change directory ~~c~~ to C drive

=> cd startup Idea / Git-Demo. } ——> you can also write like this.

=> ~~a~~ You can also ~~dis~~ ~~Open~~ directly open it from that folder.

=> git init ——> It will create the empty local repository to that ~~flo~~ folder.

=> ~~g~~ git add . ——> After creating empty local repository, you need add . all files to staging Area (Tracked file).

you can also use particular file name

=> git rm --cached ~~chase~~ [file name]. } ——> You can remove ~~remove~~ the particular file which you don't need to be tracked..

=> git status ——> It will show the status of your files.

=> git restore [filename ~~Htx~~] ——> ~~Bu~~ If you have ~~do~~ done some Changes in file then you can also restored it to - staging Area.

=> git commit -m " message. " } ——> By using this you can ~~save~~ transfer all the files to local repository.

=> git config --list } ——> It will show the your user name & email of git account

=> git config -- global user.name } If ~~f~~ your ~~g~~ user name
user.email } & email is not showing there then ~~you need to~~ by using this you can it of edit ~~it~~ also

=> git branch -m main }→ By using this you need to Change branch ~~f~~ master to main.

=> git ~~push~~ remote add origin Your github repository link. }→ Need to add your repository.

=> git push -u origin main }→ Using this you can push your files to your remote repository. It will take little time.

=> git clone. github ~~set~~ repository link }→ Another person can get all the files ~~f~~ of that repository to their Local repository.

=> git pull github repository link }→ It is use to just pull the files which are modified or newly added to Repository.