GitHub

What is Version Control System(VCS)?

VCS, also known as Source Control, is a process of tracking and maintaining any code changes to a software on the cloud.
It is basically used for development teams to collaborate while working on same/different teams. It keeps track of the current changes as well as the history of the changes made(So, in case of any mistake, we can revert back to the previous version).

Some basic info and actions that VCS gives us -

1. A complete long-term change history of every file.

2. Branching and merging. Merging is basically like, when multiple people are working on the same piece of code, and they have to commit their changes, it provides a facility to combine all their work.

3. Traceability - Ablility to trace each change made to the software and connect it to project management and bug tracking software such as Jira.

Git & GitHub

Git is by far most widely used VCS.
Git is a version control system that lets you manage and keep track of your source code history. GitHub is a cloud-based hosting service that lets you manage Git repositories.

Git installation -

Windows -

1. Get the latest Git installer - https://git-for-windows.github.io/

2. Install it

3. Open the Git Bash

4. Configuring the username and password
    1. git config --global user.name "Blah Blah" $ git config --global user.email "email@gmail.com"

5. Install the credential helper if we need to store the passwords(optional)

Linux -
Ubuntu/Debian -

1. Update the repositories
    1. sudo apt-get update

2. Install git
    1. sudo apt-get install git

3. Verify the installation
    1. git --version

4. Configure the username
    1. git config --global user.name "Emma Paris"

5. Configure the password
    1. git config --global user.email "email@gmail.com"

Setting up the repositories in GitHub -

Repositories are folders which contains the files and folders of your software.

Login to github.com and create a new repository

Basic commands -

Create a new repository on the command line

1. git init

2. git add README.md

3. git commit -m "first commit" (Commit with a message)

4. git branch -M main (Setting the branch as main, this is the default branch)

5. git remote add origin https://github.com/shubhambengani22/Software-Engineering.git

6. git push -u origin main (Pushing the committed code to GitHub)

Pushing into an already existing repository -

1. git remote add origin https://github.com/shubhambengani22/Software-Engineering.git (this is just an example, you can find the git link by pressing button Code, or it will appear directly on the screen the first time).

2. git branch -M main

3. git push -u origin main