**GIT CONFIGURATION & USAGE**

Git is a version control system that is used by web-based platform GitHub to assist developers in sharing and managing their code. It offers a cooperative setting where numerous users can track changes over time, collaborate on projects, and contribute to open-source projects. Here's a brief overview:

Git: What Is It?  
Linus Torvalds developed the distributed version control system known as Git in 2005. It makes it possible for several developers to collaborate on a project at the same time without interfering with one another's work. It keeps track of all the changes made to the codebase, archives these changes, and assists in resolving conflicts that may occur when separate code versions are combined.

GitHub: What is it?  
Git repositories can be hosted by GitHub. It gives Git a web-based interface, which facilitates repository management, teamwork, and code sharing. In addition to being popular for open-source projects, GitHub allows users to host private repositories for businesses or individuals.

Step 1: Using command “sudo zypper install git” to install the git

A screenshot of a computer

Description automatically generated

Figure 1

Step 2: Using command “git --version” to check the version of git to ensure that is latest version.  
A screenshot of a computer

Description automatically generated

Figure 2

Step 3: Click create repository to create a new repository.  
A screenshot of a computer

Description automatically generated

Figure 3

A screenshot of a computer

Description automatically generated

Figure 4

Step 4: Create a folder for GitHub, then right click that and run on terminal.  
A computer screen shot of a computer

Description automatically generated

Figure 5

Step 6: Using command “git clone https link” to import repository to your OS.   
A screenshot of a computer

Description automatically generated

Figure 6

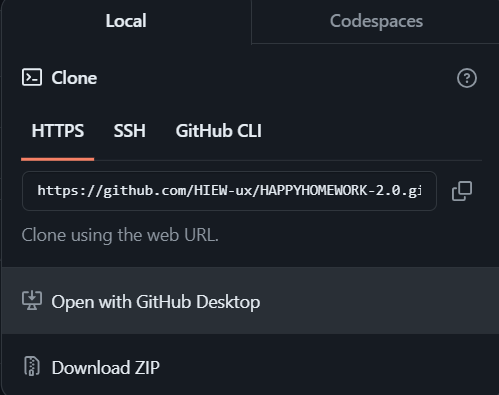


Figure 7

Step 8: Using command “ls” to check the files in folder, and using command “touch file\_name” to create a file.

A screenshot of a computer

Description automatically generated

Figure 8

Step 9: Insert some coding in the file you create.

A screenshot of a computer

Description automatically generated

Figure 9

Step 10: Using command “git add file\_name” to place the file in staging area before commit. After that using command “git status” to check if the file can be uploaded or not. (red means cannot, green means can)

A screenshot of a computer

Description automatically generated

Figure 10

Step 11: Using command git commit -m “comment” to save the file in repository.

A screenshot of a computer screen

Description automatically generated

Figure 11

Step 12: Go to GitHub website and then select settings, then click Developer settings. Then choose Token (classic) to build a token.

A screenshot of a computer

Description automatically generated

Figure 12

A screenshot of a computer

Description automatically generated

Figure 13

A screenshot of a computer

Description automatically generated

Figure 14: Name your token then adjust expiration time, then select all the scope.

A screenshot of a computer

Description automatically generated

Figure 15

Step 13: Using command “git remote set-url origin <https://Token_link@github.com/repository_link>” to manage the repository. Then using command “git push origin main” to allow that you can see your file in GitHub.

A screenshot of a computer

Description automatically generated

Figure 16

A screenshot of a computer

Description automatically generated

Figure 17

A screenshot of a computer

Description automatically generated

Figure 18