**Git Version Control**

**Version Control**

* It is a system that keeps track of changes made to a file or group of files over time so that you may later refer back to any particular versions.

**Version Control Systems**

* VCS allows you to compare changes over time, identify who last edited something that might be causing a problem, who introduced an issue and when, and more. It also allows you to roll back selected files or the entire project to a previous state.

**Local Version Control Systems**

* This method is quite popular since it is easy, but it is also prone to errors. It is simple to write to the wrong file or copy across files unintentionally if you forget which directory you are in.



**Centralized Version Control Systems**

This method has many clients that check out files from one central location and a single server that contains all the versioned files. It is also the standard for version control. Aside from its advantages, there are also some serious downside effects such as the whole history of the project will disappear if the hard drive the central database is stored on becomes corrupted and sufficient backups haven't been preserved, with the exception of any individual snapshots that individuals might keep on their own computers.



**Distributed Version Control Systems**

In this method, any of the client repositories can be copied back up to the server to restore it if any server dies and these systems were collaborating through that server. Every copy actually serves as a complete backup of all the data.

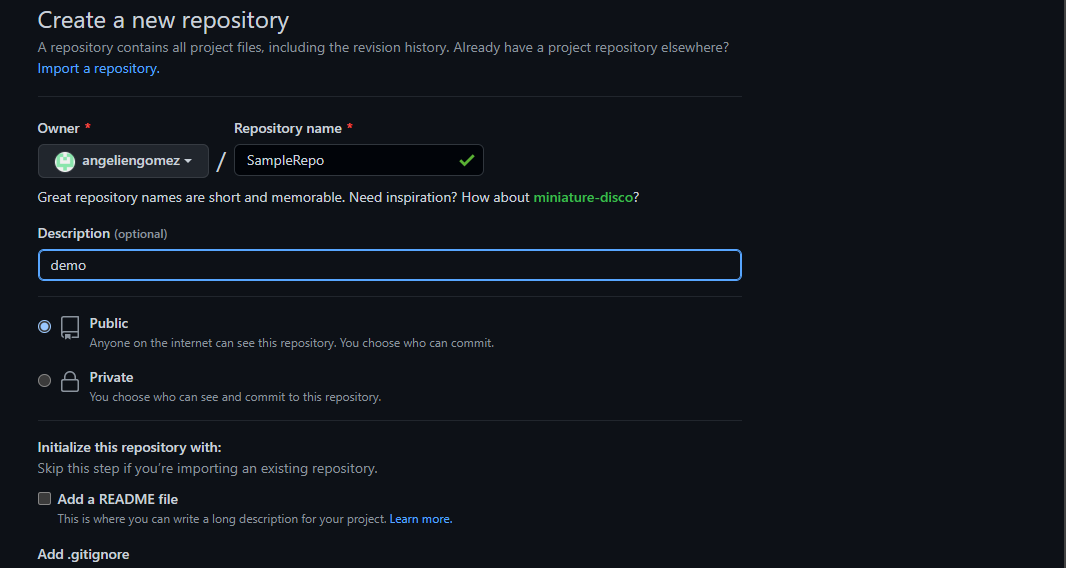


**How does version control work?**

* It works through a repository and a workflow. **Repository** means a storage or a main folder that is used for storing the files of the project. **Workflow** refers to a process that can be summarized by the “commit-pull-push” mantra. **Commit** means changes you have made to files in the repository will be saved as version of the repo and have copied it online. **Pull** means it should be done before you make changes. You need to make sure that you are up to date with the latest version. **Push** means if you are up to date your local and online copy will be the same.

**How to make your first repository and apply the Commit-Pull-Push mantra.**

1. **Create a repository on GitHub.**

****

1. **After you create your repository, navigate to your file explorer, locate and create a folder that will store your files.**
2. **After you have created your folder, type cmd to file path.**

****

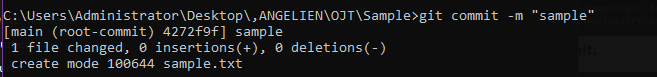
1. **Type “git init” to initialize the repository.**

****

1. **Type “git add .” to add a file to commit.**

****

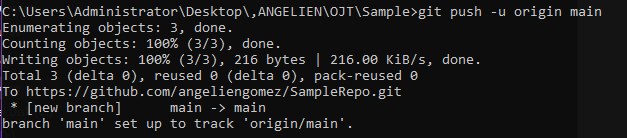
1. **Type “git commit -m “your message”to confirm changes and will save it to the local Git repository.**

****

1. **Type “git branch -m main” to create a branch that will store the changes**

****

1. **Type “git remote add origin <<repository link>>” to link it to your repository**
2. **Type “git push -u origin main” to be added on your repository.**

****