Github Essay

Git can be defined as a version control system which is widely used in modern time. It allows developers to collaborate and track changes throughout the software projects they’re creating. A git workflow helps developers use git in the most efficient manner to them. It helps determine how changes are made, like reverting to older versions, how people collaborate with each other (independently or with each other in multiple ways), and it sets an example for the team to follow throughout the process of development.

The git commit command serves its purpose by capturing a moment in time from the program being developed in question. It saves the changes at a certain point in time, allowing developers to track progress and understand milestones, bettering the program from beginning to finish. It may sound like it is ‘frozen in time’, and that’s because it is, git will never change the file unless the user asks it to. Instead of tracking solely differences, git keeps track of the entire file after each commit is done.

The git push command allows developers to move their local code into a remote repository. Basically, the local commits are transferred from the local area to the online repository. Not to be confused with git fetch moves commits to local repositories, it moves commits to remote repositories. Git fetch and pull can be viewed as the download counterparts to push which is an upload tool. Users must exercise caution with push since it overwrites with the changes and can cause the team to lose commits.

The git pull command basically takes (or fetches) content from a remote repository, downloading it into the user’s local repository and updating it to match the content they downloaded. Git pull can be seen as a combination of both git fetch and git merge. It will first fetch the content, then leads into a merge commit.

The git merge command combines branches together. Basically, say there is a main branch and as the work gets done, all of it needs to be put together into one branch, so the merge command helps do that. It allows developers to combine all work and branches into one. However, there is an issue that people can run into called merge conflicts. If two developers are working on the same file and try to change the same lines, it will lead to a merge conflict. The same thing occurs if someone deletes a file while someone else is working on it. Git lets the developer know of the conflict. Problems may arise if pending changes are carried over by commits. Both of these cases show that a merge conflict may occur during a merge or before it even starts, preventing it to start the process.

In conclusion, git is a useful tool that developers use today. The commit command allows project development to be enhanced and tracked properly as each commit shows a snapshot of the past. The push command allows developers to easily send their code from their local files to the remote repository, sharing their work. The pull command allows them to gather content from a remote repository into their own, to work on directly. The merge command allows them to do what it says: merge branches in repos together which sometimes can lead to annoying errors.