Rocco Vulpis

October 30, 2023

Git Workflow

When talking about git, it may seem overwhelming at first as there are many aspects that go into it. Whenever a conversation about Git arises, keywords get thrown into the conversation such as “push”, “pull”, “merge”, “commit”, and more. To somebody who has never used git before, all of these words will sound foreign. We’re going to look at some of these keywords within Git and decipher their meanings.

Before we get into the specific technicalities of Git, let’s briefly summarize what it is. Git is a “distributed version control system designed to handle everything from small to very large projects with speed and efficiency” (git-scm.com). So now the question you may be asking is what is a version control system? According to atlassian.com, “Version control, also known as source control, is the practice of tracking and managing changes to software code” (Atlassian.com). Git makes it easier for developers to collaborate on projects. This is because source code can easily be managed with the ability to track any changes made. If there were any mistakes that emerged, developers can go back to earlier versions of their source code to fix the mistake without interrupting the progress of other team members.

Now that we have an overview, we can now get into some of the aspects that make up Git. First, we’ll look at Git workflow. There are a few major workflows such as “Centralized”, “Feature branching”, “Gitflow Workflow”, and more. It’s important to consider team members and discover which workflow works best for everybody to enhance productivity. The most popular workflow of the few listed is the Centralized workflow. What does this workflow entail? To start, it uses a central repository to serve as a single point of entry for every change in the project. Developers clone this repository and store their changes locally. To publish the changes made, team members push their main branch to the central repository.

To make changes to this central repository, we need to see how to utilize “repositories”, “push”, “pull”, “merges”, “merge”, and “merge conflicts”. A repository is “a place where you can store your code, your files, and each file's revision history” (github.com). It is one of the most basic elements of Git. Permissions can be set to these repositories to either allow other users to be able to view or edit the elements within.

If a repository is public, a user can use a git “pull” to download content from the repository. From there, they can edit the code that they pulled on their local device. The pull command is a combination of another two commands which are “fetch” and “merge”. When pull is called, it first executes a fetch scoped at the local branch that the head is pointed at. The content from the repository is downloaded and then enters a merge workflow. The merge commit is created, and the head is updated to point at the new commit. If developers are editing the same branch simultaneously, they can run into a merge conflict. To resolve this issue, developers will work in separate branches. When they’re ready to upload their code, git merge will combine the separate branches to resolve editing conflicts.

However, code isn’t automatically uploaded to the repository. Once developers are ready to share their code, they use the “push” command to upload their work to the remote repository from their local repository. From there, developers can review their peers’ work and discuss any issues that were apparent. Mastering Git's concepts and workflows is essential for any developer aiming to streamline collaboration and efficiently manage changes in any software project.

Sources

Atlassian. “Git Workflow: Atlassian Git Tutorial.” *Atlassian*, www.atlassian.com/git/tutorials/comparing-workflows. Accessed 24 Oct. 2023.

“Creating and Managing Repositories.” *GitHub Docs*, docs.github.com/en/repositories/creating-and-managing-repositories. Accessed 24 Oct. 2023.

*Git*, git-scm.com/. Accessed 28 Oct. 2023.

“What Is Version Control?” *GitLab*, GitLab, 13 Apr. 2023, about.gitlab.com/topics/version-control/.