Git and GitKraken

What Git is, How to Stage and Commit, and How to Push and Pull

**A Report by Tom Callister**

**January 16, 2017**

Git is a version control system that helps track all of the changes that each team member makes throughout a project, allowing the data and assets to be stored in one central location while multiple team members access those files to work on the project. When working locally, you can view the file history, view differences, and commit changes. When working remotely, you share files with your team on a common repository that is not on your personal machine.

When changes are made to a file within the local repository, GitKraken will highlight those changes, but they will not be available on the remote repository until the changes are committed and pushed. GitKraken will recognize new or changed files and will list them in the “Unstaged Files” box on the top right side of the screen. First, stage the files to get them ready to commit, which will move the files to the lower box on the right side of the screen. You can then write a descriptive title and message about the changes you are about to commit and click the large green “Commit” button at the bottom right side of the screen.

GitKraken shows a visual tracking diagram to monitor the flow of the files in your repository. An icon of the “Octocat” indicates that a file is located on the remote repository. An icon of a desktop monitor indicates that a file is only located locally because it has not been pushed to the remote. When a file is pushed successfully, the two icons will line up next to the same (most recent) file. At this point, GitHub and your local files are in sync.

When other team members make changes on GitHub, you will see the icons on the flow chart showing that your local repository is no longer in sync with the remote. This is when you need to click the “Pull” button. This will update your local files. The “Fetch” function is similar to “Pull” but does not merge the files. “Fetch” allows you to first see what the changes are.

If you make a mistake, GitKraken provides an “Undo” button that will reverse a commit after it is made. You can then “Unstage” the file and edit it to correct the mistake. This is only if you have not pushed the commit.

Rather than pushing to the master branch directly, create a branch of your own to be a “sandbox” to work in. When your work is ready to add to the team’s main project files, ask your Git Master to merge your work into the master branch by creating a pull request. You can create a pull request by dragging and dropping your sandbox branch icon on the flow chart onto the master branch icon. Then select the “Start a pull request…” option on the list that pops up. If you make changes after the pull request has been made but before that branch has been merged into the master branch, you do not need to make an additional pull request. The Git Master will be able to see that further changes have been made. It is generally a good idea to delete the sandbox branch after it has been merged into the master branch.