**Why Use Git?**

It is important to listen closely to the story he shares about his first web site project. How he struggled managing the changes, reacting to the clients needs. His story exemplifies the reason solutions like Git evolved and has become a critical tool used in primary workflow strategies implemented by the majority of development efforts today. Repositories provide a way to capture moments in time in the development cycle. Without one you will find yourself making copy after copy of your work and labeling it with the date and time appended to the name. Through using a repository you are positioned to manipulate your code confidently knowing you can always step back (revert) to different points in a project with little effort.

**Things You Should Know**

**What a Version Control System is.**

* A Version Control System is software that tracks changes in code over time by maintaining copies of each revision submitted.
  + you submit changes to the system to take a snapshot of your files and it permanently saves them to be recalled at any time
* There are many benefits of using a Version Control System the following are those one should become familiar with when getting started:
  + eliminates the need to keep multiple copies of code/files on your computer through versioning
    - each version has a description about the changes, such as fix a bug or add a feature
    - code stored in versions can be viewed and restored from version control at any time as needed
  + maintains a history of changes to files as a team saves new versions of code
    - history can be reviewed to find out what, who, why, and when changes were made
    - history establishes the confidence to experiment since you can roll back to a previous good version at any time
  + workflows implemented by these systems drastically decreases the chaos of everyone using their own development process with different and incompatible tools
    - version control systems provide process enforcement and permissions based process steps that keeps everyone stays on the same page
  + code submissions are synchronized and checked for conflicts allowing multiple team members to simultaneously work in the same code
    - conflicts are when code/files have changes that appear on the same lines or remove code that is in one version and not the other
    - helps to resolve and prevent conflicts, even when people make changes at the same time
  + A version control system provides opportunities for automated tasks reducing opportunities for mistakes and inconsistencies
    - automation features save your team time and generate consistent results
    - automate testing, code analysis , and deployment when new versions are saved to version control

**What Git is.**

* Git is a [free and open source](https://git-scm.com/about/free-and-open-source) distributed version control system designed to handle everything from small to very large projects with speed and efficiency.
* Git is [easy to learn](https://git-scm.com/documentation) and has a [tiny footprint with lightning fast performance](https://git-scm.com/about/small-and-fast).
* Git outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like [cheap local branching](https://git-scm.com/about/branching-and-merging), convenient [staging areas](https://git-scm.com/about/staging-area), and [multiple workflows](https://git-scm.com/about/distributed).

**What GitHub is.**

* GitHub is a web-based hosting service for version control using git.
* GitHub mostly used for computer code but can be used for most any files.
* GitHub centralizes all of the functionality of Git in one location.
* GitHub adds additional features and functionalities to the existing Git features and functionalities.

[– Atlassian Git Tutorials – Git Merge](https://www.atlassian.com/git/tutorials/using-branches/git-merge)