Jon Green

Module 3.2 Version Control

Version control is one of the most important aspects in software development. This methodology is universal and can apply to any business model. Using version control keeps the project organized and allows the developers to keep track of the current progress of the project. Not only does it help the team keep track of current progress, but it also helps the developers see the previous changes that were made to get the project to the current state. Version control is a data log that contains documented changes to the project. Version control is mandatory when working in a team environment due to the massive number of changes that can occur when multiple people working on a project simultaneously. It makes it easy for everyone to stay up to date on the project and prevents redundant work by the team members.

Each of the articles shared similar ideas when it comes to the importance of version control. They all explained the core principles of version control in addition to why it is important when working as a team. The Washington.edu article was especially useful as it focused on Github and how to use it effectively in terms of version control. In my research, it appears that using Github to collaborate and save projects is a widespread practice in the software development world. This article elaborated on the key features such as pushing, pulling, and fetching data from the repository. It also established ground rules when committing changes to the project such as adding detailed notes.

The docuware article explained best practices for implementing version control. It was also interesting that it included a real word example of version control that related to production plans being revised for a manufacturing company. In the provided example, the organization was not using a version control system and confusion ensued due to the team not being aware of changes that were made by other members. This is a perfect example of how disorganization and wasted time can occur due to everyone not being aware of changes that are actively being made to a project. A version control system could have alleviated this entire issue by keeping everyone on the same page with the information.

Of all the articles that I used, all of them were relevant to today’s use cases. I have compiled of list of guidelines that I feel is important when implementing version control into a project. The first step would be to select an option that is automated such as Github. This eliminates the option of having to manually add documentation such as timestamps when changes occur. I would also suggest the use of audit logs. The logs will contain detailed information regarding the current state and changes that were made to the project. A retention policy is also important as it details how long previous versions of the project are retained. It would also detail what criteria needs to be met to have those previous versions deleted. A conflict resolution policy should also be implemented. The policy should contain stipulations that a file cannot be checked out for revision by multiple users simultaneously. This policy would also focus on highlighting user changes that could cause a conflict to ensure that they are resolved before the project is finalized.

Sources:

Atlassian. (2019). *What is version control*. Atlassian. <https://www.atlassian.com/git/tutorials/what-is-version-control>

*The ultimate guide to document version control*. (n.d.). Nulab. https://nulab.com/learn/collaboration/document-version-control/

‌

Ernst, M. (2022, March 7). *Version control concepts and best practices*. Washington.edu. https://homes.cs.washington.edu/~mernst/advice/version-control.html

‌

Honig, J. (n.d.). *What is Version Control & Why Is It Important?* Start.docuware.com. https://start.docuware.com/blog/document-management/what-is-version-control-why-is-it-important

‌