Jeffrey Reid Module 3.2 January 26, 2025

Version Control Guidelines

Source one by Filestage suggest that version control is best operated under 4 distinct practices:

1. Naming conventions should be defined: This involves a consistent format so that each user can be communicated the surface level information of a file. This is great for determining version differences as well as informing others as well as providing a consistent level of organization. Examples like Legal\_Policy\_v1 help to easily differentiate to others what version you are currently operating with.
2. Define clear version lines: This is great for determining when feedback stops and ends. It would not benefit many people if feedback for a particular version was still being given when the next version is already underway. This solves this by providing windows for said feedback during an appropriate window.
3. Centralize feedback retrieval and discussion: Without a centralized form of discussion, chaos can ensue as different people communicate with one another and possibly even modify different versions of software. This can make it difficult to keep track of everything and even set back overall work. Centralized feedback allows everyone to collaborate together and blocks stray edits to software.
4. Make version comparison easy: The big benefit to this is that everyone can see the changes being made across the different versions of software. With proper documentation, everything is in plain view in one location so everyone can see all of the different types of feedback and changes. This cuts back on time spent sifting through previous work in different locations.

Source two by “Image API” provides 5 different guidelines to version control success:

1. Standardized naming convention: Like the first source, it’s suggested to develop a consistent naming system for your work. This provides basic information such as version number and time/date of last edit for everyone to see.
2. Use the right version control: Not every version control out there is right for you. Make sure that you're using the right version control that has features tailored to your business needs. Consider things like audit histories, archiving, collaborative access, and version preservation. This boosts overall teamwork and provides important information for everyone.
3. Use Permissions and Access Control: Not everyone should be able to access everything. Limit access to certain people to enhance security.
4. Formalize versioning protocols: This brings the entire group under the same rules of understanding regarding how to work with different versions of the software.
5. Appropriate Training: Make sure everyone is trained in order to bring everyone under the same page. This increases group understanding and boosts compliance and overall product use.

Source three by “Progress Sharefile” also provides some similar guidelines to the other two:

1. Know your needs: You can’t start working unless you understand what tools you need to accomplish your objectives. Keep compliance and the complexity of the project in mind when choosing your version control.
2. Implement the correct version control management: This is key for providing various security and audit features. It also enhances overall collaboration with the team and should integrate with existing tools.
3. Set versioning guidelines: This includes setting things like naming conventions, archiving methods, and version numbering techniques. With these guidelines, everyone is under the same page and can identify different versions faster.
4. Up to date Training: To further get everyone on the same page, make sure training is implemented so that all version control features are understood. This also boosts security and compliance and cuts back on simple errors.
5. Process Monitoring: Listen to feedback and make improvements to the overall process as you go. Success is dependent on the ability of everyone to adapt and improve as time goes on.

There were a lot of similarities between each source. The key things that stuck out to me were Training and maintaining a framework for version identification. These two things help provide the appropriate work knowledge to everyone to create a consistent method of identifying different versions of the software you’re working on. Centralized feedback is also equally important as this can be used to maintain appropriate communication across the team and identify changes before the next version of software has begun work. Some sources had differences on smaller things though. For example, ImageAPI focuses on setting permissions to control the access to different parts of the software. This differed from FileStage who stressed the importance of setting up a strong system to compare different versions. Overall, most things were repeated but there were a few niche differences between the three.

For me personally, the most important thing to keep in mind is the version control naming system. This immediately informs every one of the different versions that have been worked on and most importantly, the current version. Having this set up to be consistent prevents different naming schemes that can cause confusion too. Following that, choosing the right version control based on your needs is just as important. It would not serve anyway to choose a version control that does not suit the overall needs of the business. Find the one right for you and your team and stick with that. Just as important, all feedback should be centralized and monitored. If everyone can see the different types of feedback in one location, this can boost overall collaboration and provide appropriate documentation for all. If you need to reference this information, it’s in one place and you won’t have to go searching everywhere for it. The last thing I would implement would be consistent training so that everyone is on the same page. This ensures everyone has the necessary skills to operate the version control and decreases overall confusion. I personally don’t think

Citations

*Document version control best practices to make your life easier*. (2022, September 9). Filestage.io. <https://filestage.io/blog/document-version-control/>

*Document version control: a guide to effective version management*. (2024, December 10). Sharefile.com; ShareFile. https://www.sharefile.com/resource/blogs/document-version-control-guide

*5 Document Version Control Best Practices & Guidelines [2024]*. (n.d.). Www.imageapi.com. https://www.imageapi.com/blog/version-control