**Version Control**

Kristina Vasquez

Bellevue University

CSD380-J308 Devops (2253-DD)

Sue Sampson

11/10/2024

We regularly encounter software updates on our devices, but behind each update is a team managing version control to ensure the latest features and bug fixes are properly integrated. Version control, as defined by the article *What is Version Control?* is the practice of tracking and managing changes to software code, allowing software teams to track modifications over time (Atlassian, n.d.). To explore best practices for implementing version control, I reviewed guidelines from four sources: *What is Version Control?* from Atlassian, *Document Version Control: Best Practices and Tools for Managing Document Versions* from Filestage, *Version Control Guidelines* by the NIH (National Center for Complementary and Integrative Health), and *Document Version Control Made Easy (With Examples)* from Rebels Guide to PM.

The *What is Version Control?* guide from Atlassian emphasizes best practices for software development, such as creating detailed commit histories and using branches to maintain a stable main codebase (Atlassian, n.d.). These practices ensure that every change is tracked and that developers can work on separate features without interfering with the main project. In contrast, Filestage's *Document Version Control* guide focuses on document management and underscores the importance of clear file naming conventions and maintaining an audit trail for compliance purposes (Filestage, n.d.). Although these guidelines are tailored for document management, many principles are applicable to software development, particularly when documents are part of the project.

The NIH's *Version Control Guidelines* (2015) offer a broader perspective, combining software and document versioning. They recommend centralized version control systems to track changes in a central repository. While this approach was relevant in the past, it is less ideal in modern development environments where distributed version control systems (DVCS), such as Git, are preferred for their flexibility and collaborative advantages.

Filestage's guidelines are still highly relevant today, especially with the rise of cloud-based document management tools like Google Drive, which automatically track changes and maintain version history (Filestage, n.d.). However, some older guidelines, such as those from NIH, suggest centralized systems, which are less suited to the needs of today's distributed workflows (National Institutes of Health, 2015). In contrast, Git's distributed model, along with services like GitHub, has become the industry standard due to its ability to support collaborative workflows and handle complex development scenarios more effectively. Git allows teams to manage multiple versions efficiently while maintaining the integrity of the codebase.

Based on my research, the most critical version control guidelines today are as follows: 1) Use distributed version control systems like Git for better collaboration and flexibility; 2) Implement clear and consistent commit messages to explain the purpose of each change; 3) Use branches for feature development to avoid disrupting the main project; 4) Track and document all changes to maintain a clear history; 5) Adopt consistent naming conventions for files and commits to reduce confusion; 6) Encourage frequent, small updates instead of large, infrequent ones to minimize errors and improve traceability. These guidelines are crucial for efficient management of project changes, fostering clear communication, and minimizing errors in collaborative environments.

In conclusion, version control is essential in both software development and document management. While some older guidelines—such as those advocating for centralized systems—may still hold value, today's best practices lean toward using distributed systems like Git. By following modern version control guidelines, teams can ensure smoother collaboration, maintain traceable changes, and achieve a more organized, error-free workflow, all of which are critical for successful project management.

**Citations:**

Atlassian. (n.d.). *What is version control?* Retrieved from <https://www.atlassian.com/git/tutorials/what-is-version-control>

Filestage. (n.d.). *Document version control: Best practices and tools for managing document versions.* Retrieved from <https://filestage.io/blog/document-version-control/>

National Institutes of Health. (2015). *Version control guidelines.* Retrieved from <https://files.nccih.nih.gov/s3fs-public/CR-Toolbox/Version_Control_Guidelines_ver2_07-17-2015.pdf>

Rebels Guide to PM. (n.d.). *Document version control made easy (with examples).* Retrieved from <https://rebelsguidetopm.com/how-to-do-document-version-control/>