DeJanae Faison

M3.2 Assignment – Version Control Guidelines

Due 8/31/25

# Version Control Guidelines

Resource 1: [Daily Dev- Documentation Version Control: Best Practices 2024](https://daily.dev/blog/documentation-version-control-best-practices-2024)

Daily Dev covers the best practices for version control and explains the action and purpose, method and description as well as task and why it matters. First, they outline that the best practices for version control include:

1. Set up clear version control plan
2. Commits with clear messages
3. Use branches for features or releases
4. Implement regular reviews
5. Keep documentation in line with code changes
6. Use CI for automated updates
7. Implement security measures and access controls
8. Regularly test recovery processes

Version Control helps mitigate mistakes since tracking changes makes it easier to spot and fix. It also speeds up the process since less mistakes means less time fixing them and allows for multiple people work on the project at once. When following the first step of the process they break down to:

1. Choose a system
2. Set up storage
3. Names files clearly
4. Pick a version system

All these steps will help in organization thus minimizing mistakes made and creates a ‘master book’ of how the project will be executed. Each segment they explain essentially boils down to keeping things up to date, properly documented, modular, automated, and doing regular check ins with not only the software but with the team.

Resource 2: [RDM Guide- Version Control](https://rdm.elixir-belgium.org/version_control#version-control-methods)

RDM describes that version control allows for the deletion of drafts or redundant versions, reducing duplications and misunderstanding. They explain a version control system should be agreed from the start and adopted by the entire team or group. That depending on project, how much depth needs to be put into the version control. Though they focus mainly on version control for documents they cover that using a table can also be used as a way to keep track of versions, status, changes and who is accountable for the change.

Resource 3: [DocuWare- The Ultimate Guide to Document Version Control](https://start.docuware.com/blog/document-management/what-is-version-control-why-is-it-important)

DocuWare states in one of their best version control practices is to develop a clear SOP or Standard Operating Procedure and ensure its enforced. Developing a clear standard of what is expected of naming conventions, version numbering and archiving should be established and maintained. DocuWare also list having Access Controls, Audit logs, Automated approval workflows, centralized storage and having archiving and retention polices make for good practice. They also state implementing automated approval workflows. Not to be automatically approved but ensure the files are sent to the correct reviewers and approvers to reduce any sort of bottle necking.

Compare and Contrast Guidelines:

|  |  |  |  |
| --- | --- | --- | --- |
| Aspect | Daily Dev | RDM Guide | DocuWare |
| Scope | Documentation workflows | Research data governance | Document control in enterprise document management systems |
| Planning | Clear versioning, naming and system choice | System adoption from the start | Implied planning through system design/tracking |
| Tools | Git,SVN, CI, hosting tools | Governance, method documentation | Document managing system features like audit trails  Git & Guthub |
| Security/Compliance | Access controls, secure system | Governance compliance | Focus on permissions, audits and integrity |

Most Important Guidelines and Reasoning:

1. Establish a plan and systems used from the start the team agrees upon
   1. Doing it from the start will eliminate any confusion or misunderstandings later on.
   2. Establish: System, tools, Version numbering, naming conventions, commentary, archiving policies, coding compliances and access controls
   3. I felt the team agreeing upon a system is important since it gives everyone a say and helps highlights any strengths and weaknesses within the team.
2. Adhere to plan however if plans need to change reconvene with the team and ensure all documentation, software and other resources are informed and up to date
   1. If things do change, for instance switching systems, it does not hinder process
3. Check Back up protocols regularly
   1. Let’s say a new feature is to be released. Check before launching if the feature can be undone without altering the entire system
   2. Ensuring a back up system is working will help have a ‘safety net’ if anything is wrong
4. Commits should have clear messages
   1. As a solo developer I will admit when hitting commit to a file I have typed ‘fsbrgwy53’ in the comment or title section. However, when working in a group it is important to create clear message on:
      1. What was changed
      2. Why it was changed
      3. What was added
      4. What was removed
   2. It helps keeps things organized.

Reference:

*Documentation Version Control: Best Practices 2024*. (2024). Daily.dev. https://daily.dev/blog/documentation-version-control-best-practices-2024

‌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

OpenAI. (2025). ChatGPT (40 mimi). <https://chat.openai.com/chat>

‌*RDM Guide*. (2025). Elixir-Belgium.org. https://rdm.elixir-belgium.org/version\_control#version-control-methods