Madisyn Kovacic

CSD 380

Version control is an important part of today’s software development. It helps teams keep track of code changes, understand the history of their work, and work together better. This paper shares simple guidelines for version control.

Atlassian's research shows that writing clear commit messages, making changes often, and managing your project branches well are super important. They suggest writing commit messages that explain why you made changes, using separate branches to keep your main project steady, and merging your work regularly to avoid big, complicated updates. GitLab's guides on keeping track of your code highlight teamwork and using automation. They say it's important to run tests automatically when you make changes to keep the code good, have code reviews for everyone’s changes to help share knowledge, and use tags and version numbers to mark different updates clearly. Also, IEEE Software Standards for version control remind us to document everything properly and keep track of our changes. They recommend keeping detailed logs of what changes were made, using consistent names for branches across teams, and routinely checking for any old or unused branches.

These guidelines have a lot in common, like stressing the importance of clear commit messages and managing branches. However, some parts are designed for specific situations. For instance, GitLab's focus on automation works great with DevOps workflows, while the IEEE's detailed logging is important for following rules and regulations. On the other hand, using hand-written

change logs, which IEEE recommends, might not be as useful nowadays since we have automated systems that can track changes much better.

First, when you make changes, write clear notes about what you did and why. It helps everyone understand later. Next, use separate branches for new features so the main work stays safe and everyone can collaborate without stepping on each other's toes. Reviewing each other's code is helpful too. It improves what we write, helps us learn, and catches mistakes early because we're all participating in a checks and balances kind of dynamic. Also, running tests automatically before we combine changes makes sure we don't mess things up, and labeling our versions helps everyone keep track of different software updates. Finally, merging our changes often stops big problems from happening and keeps the project moving forward, and cleaning up old branches helps us keep everything tidy.

These tips were chosen because they help make things clearer when keeping track of changes in a project. Simple things like using understandable descriptions for changes and having separate branches for new features help everyone work together better. Also, using automated tools and having others check your code keeps the quality high. Making sure to regularly combine changes and keep old versions helps keep everything organized in a project.

Sources

Atlassian. "Git Workflow | Atlassian Git Tutorial." *Atlassian*, 2010, www.atlassian.com/git/tutorials/comparing-workflows.

Spinellis, D. “Version Control Systems.” *IEEE Software*, vol. 22, no. 5, Sept. 2005, pp. 108–109, https://doi.org/10.1109/ms.2005.140 Accessed 8 Jan. 2020.

“What Are Git Version Control Best Practices?” *GitLab*, about.gitlab.com/topics/version-control/version-control-best-practices/.