2-1 Journal: What Makes a Productive Code Review?

Mohamed Saleh

Southern New Hampshire University

CS 499: Computer Science Capstone

Prof. Wasim Alim

June 13,2025

To answer the question, "What is code review?", we first need to understand why we need it in the first place. What is the importance of code review? How can we conduct a successful review process?

Change is the only constant in the software world. Codebases are consistently modified by different changes—whether for a new client feature, a production bug fix, a code enhancement, performance optimization, or updating a dependency to prevent security vulnerabilities. Collaboration between developers working on a new system or feature introduces another layer of complexity. Modifying the same files—adding, removing, or rearranging lines—can be a nightmare for some developers. Due to these challenges, we need a process to ensure changes are made in a structured manner, following rules based on team discussions, best practices, and industry standards. This process is called code review. Code review can begin when starting a new project or be conducted continuously to avoid technical debt.

Technical teams need to establish a set of rules that suit their technologies to ensure high-quality code delivery. Some practices can be applied across any tech stack, while others are language-specific, such as those for C and C++. Code review can be performed during or at the end of the development cycle. Code review covers different aspects of the code, including code structure, documentation, variables, arithmetic operations, loops and branches and defensive programming. For each category, there is a set of checklists to follow during the review; examples include avoiding floating-point number comparisons, preventing rounding errors, Handling division by zero, eliminating repeated code, avoiding code smells, preventing code injections (e.g., SQL injection)

The code reviewer should highlight issues based on the agreed-upon checklist or guidelines and communicate them professionally and politely. In my opinion, code reviews should be conducted before merging branches into the main trunk. Additionally, developers should be mindful while writing code and adhere to the checklist to speed up the merging process.

For the code review process, I will use OBS (Open Broadcaster Software) for recording. My approach involves creating an outline or script for code reviews across three categories. I will list the necessary changes for each category per project and explain the pros and cons of the current code status and design decisions.

References

CS 499 Code Review Checklist Structure. (n.d.). https://snhu-media.snhu.edu/files/course_repository/undergraduate/cs/cs499/cs499_code_review_checklist.pdf