Name: Shekhar Chaudhary

Date: 09/12/2025

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

## Part 1

1. What is code review?  
Code review is the systematic examination of source code written by one developer and evaluated by other developers to identify bugs, improve code quality, and ensure adherence to project standards before merging into the main codebase.

2. Why is it an important practice for computer science professionals?  
Code review is important because it promotes collaboration, knowledge sharing, and consistency across a project. It also serves as a safeguard to catch errors early, reduce technical debt, and improve maintainability. For professionals, it builds team trust and reinforces coding best practices.

3. What is some code review best practices that are crucial to include in a code review?  
Some best practices include keeping reviews focused and manageable, providing constructive and respectful feedback, and using checklists to cover important areas such as readability, security, and performance. Code reviews should ideally occur before merging into the main branch to prevent defects from entering production. This stage is critical because it ensures issues are caught early when they are easier and less costly to fix.

## Part 2

4. What software has you chosen to use to record your code review?  
I have chosen GitHub as the primary tool to record code reviews. GitHub’s pull request system allows for inline comments, discussions, and approval workflows, making it an effective platform to manage and document code reviews.

5. Describe your approach to creating an outline or writing a script for your code review.  
My approach begins with referencing the code review checklist provided in the course rubric. I structure the review into three categories: readability and maintainability, adherence to standards, and functionality/performance. For each category, I prepare guiding questions and comments to ensure feedback is thorough and constructive. This organized method ensures fairness, reduces bias, and makes the process productive.

## References:

Rigby, P. C., & Bird, C. (2013). Convergent Contemporary Software Peer Review Practices. In Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering (pp. 202–212). ACM.  
Bacchelli, A., & Bird, C. (2013). Expectations, Outcomes, and Challenges of Modern Code Review. In Proceedings of the 2013 International Conference on Software Engineering (pp. 712–721). IEEE.  
GitHub. (n.d.). About pull requests. Retrieved from https://docs.github.com/en/pull-requests