**What Makes a Productive Code Review**

Cydnie Fisher

Southern New Hampshire University

CS-499 Computer Science Capstone

Professor Mike Alesso

November 6, 2024

**What Is a Code Review, and Why Is It an Important Practice**

During a code review, members of the development team will look over code written by one of the members. This allows team members to offer critiques, potential improvements, and any questions they have regarding code (Thorndyke, 2021). Code reviews should be a balance between a strictly documented process and a collaborative environment to ensure maximal effectiveness and efficiency (Flow Transformation Team, 2023). They should generate open communication and allow team members to share knowledge. Overall, code reviews help with discovering errors in code, improving code quality, and ensuring the code adheres to coding standards.

**Best Code Review Practices**

To ensure the efficiency of code reviews, no more than 400 lines of code should be read at a time, with an inspection rate of no more than 500 lines of code an hour (Smartbear, n.d.). This ensures that the brain can properly process the information found in the code and find as many errors as possible. Using a checklist is also important as it effectively decreases frequently made errors, and can provide a clear expectation of what should occur during a code review. It is also important to maintain a positive environment while performing the code review. Peer reviews should only focus on reviewing the code, and not evaluating team members. Also, rather than viewing finding defects as something negative, the process should be seen as positive, since it provides an opportunity to enhance the code, and learn from fellow programmers. Code reviews should occur as early as possible during the development process. Finding errors early on can stop minor errors from escalating into major errors (Flow Transformation Team, 2023). Minor errors can be quick to fix, while a major error could cause the project to be delayed.

**Software I Used to Record My Code Review**

To record my code review, I have decided to use OBS. OBS allows me to record both my screen and audio at the same time. It also allows me to alter the video quality and format, and doesn’t have a watermark or any time constraints. I will then use iMovie to remove the background noise and any long pauses.

**My Approach to Creating an Outline for My Code Review**

To write my outline, I began by separating my code review into three parts; explaining the code, the actual code review, and my plans for the three category enhancements. To explain the code, I wrote an outline briefly explaining the program's functionality as a whole. I then went through all of the code files, and wrote an outline explaining in further detail the specific functionality of each class. When writing the outline for the code review, I began by answering all of the questions in the code review checklist. I then used my answers to write the outline. To explain my enhancement plans, I adjusted what I wrote in the Module One assignment so that it was more thorough, and used this document as part of my script.

**References:**

Flow Transformation Team. (2023, April 28). *Code review checklist: 7 steps to level up your review process*. Pluralsight. <https://www.pluralsight.com/resources/blog/software-development/code-review-checklist>

Smartbear. (n.d.). *Best practices for peer code review*. smartbear.com. <https://smartbear.com/learn/code-review/best-practices-for-peer-code-review/>

Thorndyke, K. (2021, June 17). *How to review someone else’s code: Tips and best practices*. Codecademy Blog. <https://www.codecademy.com/resources/blog/code-review-best-practices/>