Faizah Mas Mohd Khalik
CS 499 Computer Science Capstone
Journal 2-1: What Makes A Productive Code Review?

Part 1:

1. **What is code review?**

   A code review is a systematic evaluation of a piece of code carried out through convening with one's fellow programmers and checking each other's code for mistakes, things that can be improved and what should be avoided for next time. It is an act of quality assurance, particularly before the code is merged into an upstream branch in verifying correctness, readability, maintainability and also security.

2. **Why is it an important practice for computer science professionals?**

   Code review is important to computer science professionals for a number of reasons. Reviewing code early helps to catch any problems before they make it to production, thereby saving time and reducing the costs associated with fixing bugs (OWASP, 2017). They also promote knowledge sharing and collaboration (a culture of learning) among team members so that they all get to improve together (Smartbear, 2024). In addition, guidelines like OWASP (2017) ensure that security vulnerabilities such as cross-site scripting (XSS), injection attacks and improper handling of sensitive data are identified and addressed early on.

Lastly, the knowledge that code will be reviewed encourages developers to write cleaner and more secure code - i.e. taking accountability and responsibility for the work that they put in. In my personal experience, code reviews have been helpful for me, especially in learning to build Discord Bots. The confidence that I've been able to build has helped me successfully create and manage the one that I am responsible for now, with over 1200 users. For example, one of my developers pointed out that I could have used Docker to deploy my app, which had multiple functionalities to it. They also pointed out that my database has some security vulnerabilities, which meant that anyone could access the information. Had they not spotted this, I would have placed my hard work into a potentially dangerous situation. Now, when I work on my Discord bots, I'm careful to make sure that none of my data is exposed in the same way as it was before.

3. **What are some code review best practices that you read about in the resources that are crucial to include in a code review? Include when a code review should occur in the development process with a rationale as to why.**

Code reviews should occur immediately after completing a unit of work (such as implementing a new feature) and before merging it into the main branch of the code. The timing of the code review makes sure that errors are detected early and can be fixed, while also maintaining quality. Taking from both Smartbear (2024) and OWASP (2017), the following are some best practices that may be adopted for effective code reviews:

1. Reviewing fewer than 400 lines of code (LOC) to maintain focus and detect up to 90% defects (Smartbear, 2024).
2. Keeping inspection rates to fewer than 500 LOC per hour to maximise defect detection rates (Smartbear, 2024) means that reviewers can carry out more thoughtful reviews if they go slower.
3. Limiting review sessions to 60 minutes (Smartbear, 2024) as performance decreases after focused activity of that amount of time.
4. Ensuring code security through using OWASP's Code Review Checklist (OWASP, 2017) to check for critical security issues like authentication and authorisation.
5. Using checklists and automated tools like the ones provided by OWASP standardises the code review process. The automated tools assist in detecting common issues, especially those related to security.

Part 2:

4. **What software have you chosen to use to record your code review?**

   I chose to use a mix of OBS Studio and the QuickTime Mac on-screen recorder to record my code review, depending on the computer that I will have with me at the point in time when I record it. In both cases I will put my clips on Google Drive and then edit them on one of my laptops. OBS Studio is a flexible and high-quality recording software while QuickTime is an easy-to-use option.

5. **Describe your approach to creating an outline or writing a script for your code review for each of the three categories that you will be reviewing based on the rubric as well as the code review checklist.**

Instead of writing a full script, I will create a structured outline highlighting the key points that need to be covered. I have printed copies of the code review checklist provided by SNHU, and as I go through my review I will tick off the checkboxes. Pre-recording preparation included making sure that I crossed off anything that isn’t relevant at the point for my code. Given my recording environment, I may not be able to close all other programs, so I will take steps to protect any personal or private information. I aim to maintain a professional yet casual tone throughout my recording so that it doesn't sound like I'm just reading off a script.

For each of the three categories, I will talk about the functions of the code, analyse what it does (check if it works), then speak a little bit about its background. My checks will be done in accordance with the checklist provided by SNHU. Thereafter, I will touch on the enhancement I want to make to these artifacts.

References

Conklin, L., & Robinson, G. (2017, July). *OWASP Code Review Guide*. https://owasp.org/www-project-code-review-guide/

Smartbear. (2024). *Best Practices for Code Review*. https://smartbear.com/learn/code-review/best-practices-for-peer-code-review/