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

CS-499: Computer Science Capstone

Zhraa Al-Bayaa

September 15, 2024

A code review, by definition, is a peer review process where one developer systematically examines the code written by another. It serves as a form of software quality assurance, playing a crucial role in ensuring the quality of code throughout the development process. A productive code review involves several key components: setting clear objectives, thorough preparation, a structured approach, constructive feedback, manageable code size, continuous improvement, proper documentation, and adherence to industry standards. This practice is essential in software engineering and computer science as it helps deliver high-quality code and software products. The focus of the review is solely on the code itself, not the developer, ensuring the creation of robust and understandable code for other team members. Best practices for an effective code review include defining clear goals that align with the purpose and scope of the review. Preparation is also vital and may involve self-review, understanding the context, developing a checklist, and using appropriate tools. Feedback during the review should be constructive, specific, actionable, and balanced with a positive tone. Additionally, code reviews should maintain a manageable scope, with minimal changes to facilitate smoother adoption.

I have selected Camtasia 2022 for screen recording and Visual Studio Code for reviewing the C++ artifact. For the Database Project, I'll use MongoDB Atlas, and for the Contact Service Unit Testing, I'll use Eclipse IDE. For the code review, I've chosen a vector sorting application from CS300, and I plan to enhance it by implementing a hybrid of the selection sort and quicksort algorithms, which should outperform the quicksort in terms of performance. My approach to reviewing the three artifacts will involve setting specific objectives. For the vector sorting program, I will focus on improving the algorithm. In the database project, I'll introduce indexing to enhance search performance. For the Contact Service Unit Testing, I will refine the testing

methods, improve boundary condition checks, and ensure the consistency and accuracy of the contact service.