COMP3211 Assignment 1

Oct. 9th

ZHOU Siyu 21094655D

1. ACM/IEEE SE Code of Ethics and Professional Practice

Clause 3.08. Ensure that specifications for software on which they work have been well documented, satisfy the users' requirements and have the appropriate approvals.

This clause may be violated by a software engineer by failing to maintain reliable documentation of the specifications of the software currently under development. This may occur while the engineer feels pressed on deadline or pays little attention to detail. Lack of concise specifications can result in misunderstandings, ineffective communication, and subsequently a product that does not satisfy the requirements of the users.

Software engineers should prioritize the documentation process and allocate enough time for it in order to avoid this circumstance and violation. They must guarantee that every relevant participant are involved in the approval process and have a comprehensive understanding of the requirements for the users. A organized method can help guarantee completeness and clarity in requirements, such as the use of standard documentation templates. The team and stakeholders should regularly review the documentation to find any gaps or inconsistencies. Software engineers can avoid potential problems and make sure that the finished solution complies with the consumers' expectations by devoting time and attention to carefully documenting specifications.

Clause 8.04. Improve their understanding of the software and related documents on which they work and of the environment in which they will be used.

This clause may be violated by a software engineer by not having a thorough understanding of the software he is developing and the environment in which it is to be used. This could happen if the engineer doesn't prioritize learning about the specific context in which the program will be utilized or fails to stay updated with emerging technologies. This may lead to poor software that fails to adequately handle the requirements or factors of the environment it was designed for.

Software developers should actively work to improve their knowledge of the software and its surroundings to avoid the infringement. This can be accomplished through continuing your education, staying current with industry trends, attending pertinent conferences, and conversing with peers and professionals. Software engineers may make sure they are well-equipped to create software that is adapted to its unique environment and complies with the appropriate standards and requirements by continually investing in their knowledge and understanding. This method helps to prevent potential flaws and guarantees that the software created is applicable and useful in its intended environment.

2. Software Processes

One software project I worked on before was the development of a mobile application for a clothing retail company. The application aimed to improve the customer experience by allowing them to browse products, make purchases, and receive personalized promotions.

For this project, an agile method would be more appropriate. This is because the mobile application development requires frequent iterations and continuous feedback. Agile methods focus on delivering value to the customer in short iterations, which aligns well with the iterative and customer-centric nature of mobile app development.

Communication, simplicity, iterative development, and continuous testing are important to eXtreme Programming. These considerations are critical for a mobile application project that must quickly react to evolving market demands and consumer feedback. Additionally, agile methods enhance collaboration among developers, designers, and stakeholders, leading to a more efficient and effective development process. Overall, the flexibility and customer-centric approach of eXtreme Programming made it an appropriate choice for this project.

3. Verification and Validation

Verification means checking if system conforms to its specifications.

Validation means checking if system meets the requirements of the system customers.

In our course project, Unit test is a verification activity. And the testing performed customer to checking system is a validation activity.

4. eXtreme Programming

Pair Programming: The difficulties encountered include the stronger student completing the majority of the work, the weaker student failing the exam, and the impact of different personality types and genders on pair programming outcomes.

The author discusses the issue of pair programming in an educational setting in the section where they relate their experience with implementing pair programming in a big class of first-year computing students.

There is a possibility that the stronger student may dominate the task while the weaker student would struggle to keep up. This might lead to discrepancies during group projects and impede the learning process for the weaker members. In my future course group projects, we must ensure that each group member contributes equally and successfully learns and understands.

Process and Discipline: The tendency of university students to prefer methods that allow them more flexibility to experiment, as well as the risks of using a lightweight development process such as XP in an educational setting.

The authors suggest that a less flexible development process with additional guidance is more suitable for students in the section where they discuss.

This lesson shows that when it comes to maintaining discipline over a longer length of time, a more planned and led method may be more suited for me.

5. Requirements Specification

Functional Requirement: The chat window shall enable real-time messaging, allowing users to immediately send and receive text messages. (The chat window, similar to WhatsApp/WeChat, shall allow users to conduct smooth, timely chats.)

Non-Functional Requirement - Reliability: The chat window should be up and running at least 99.9% of the time, guaranteeing that users could use and access the messaging function without substantial interruptions or downtime. (This need focuses on the system's capacity and reliability in terms of uptime and ensuring users' ability to interact without pauses in service.)