1. **Compare software error, software fault, and software failure. Provide examples of each.**
   * A software error is the original human mistake made by a designer or programmer, like a typo or a flaw in logic. For example, a developer might accidentally use a > sign instead of a >= sign when coding a condition. A software fault (or defect) is the manifestation of that error within the code itself, which causes it to malfunction in specific situations. Using the previous error, the faulty code would now be a line that incorrectly compares values. A software failure occurs only when a user or system activates that faulty code, leading to a visible disruption. In our example, the failure would happen when a user inputs a specific value that triggers the flawed comparison, causing the application to crash or produce a wrong result. Essentially, an error causes a fault, and an activated fault leads to a failure.
2. **Discuss the similarity of the ISO/IEC model with McCall’s model.**
   * The ISO/IEC 25010 model and McCall’s model are very similar in their core purpose, to categorize what makes software "good." Both frameworks break down the broad concept of quality into specific, measurable factors. For example, several factors are almost directly carried over from McCall's model into the ISO/IEC standard. Reliability in both models deals with the software's ability to avoid failures. Maintainability refers to the ease of fixing and updating the software, and Portability concerns how easily the software can be adapted to new environments. The ISO/IEC model modernizes and consolidates these ideas, showing that McCall's foundational concepts remain highly relevant for understanding software quality today.
3. **The responsibilities of project management may be classified into professional hands-on tasks and management tasks. List the project manager’s tasks and explain the objective of each task.**
   * For professional hands-on tasks, the project manager is deeply involved in the technical execution. This includes like preparing project and quality plans to provide a clear roadmap for the team. They also participate in joint customer-supplier committees to ensure everyone's expectations are aligned and also, they provide close professional follow-up on implemented solutions to guarantee technical excellence and support the team when they face complex challenges. The objective of these tasks is to ensure the project is built correctly according to specifications and professional standards.
   * For managerial tasks, the focus shifts to oversight and control. The manager controls the project schedule and budget to ensure the project is delivered on time and within financial constraints. They oversee review activities and subsequent corrections to catch defects early and also, they manage customer requests and satisfaction to maintain a good relationship and ensure the final product meets the client's needs. The objective here is to ensure the project is managed efficiently, mitigating risks and keeping all stakeholders informed and happy.
4. **The structure of an SQA unit varies according to the type and size of the organization. Describe the tasks performed by each area unit.**
   * Area 1 (SQA Process Implementation) is responsible for the foundational framework. Their tasks include establishing and updating the organization's SQA processes and preparing comprehensive SQA plans (SOAP) to outline all quality activities. Their objective is to build the system that everyone else will follow.
   * Area 2 (Product Assurance) focuses on the tangible output. Their main task is to evaluate the finished software products and their documentation to ensure they conform to all established requirements. They also measure the quality of software operation services, like customer support. Their goal is to act as a final checkpoint, verifying that what was built actually meets the defined standards.
   * Area 3 (Process Assurance) looks at how the work is done. They evaluate whether the project's plans and chosen development processes are suitable for meeting contract requirements. They also assess the software engineering environment and support corrective actions to improve processes. Their objective is to ensure that the methods and tools used throughout development are effective and are being followed correctly, which in turn prevents defects.

References

<https://elms.sti.edu/files/6767994/01_Handout_1.pdf?lmsauth=c9cdbbd5ecc4e40428c1ec8836362cff0f9e6d0f>

<https://elms.sti.edu/files/6767994/02_Handout_1.pdf?lmsauth=55f235a08d0d81a6a97e375930f545352d32d782>

<https://elms.sti.edu/files/6767994/03_Handout_1.pdf?lmsauth=26a759045a6b8b4d2ee4bee151fac3df20d86852>