|  |  |  |
| --- | --- | --- |
|  | **BAHRIA UNIVERSITY, (Karachi Campus)**  *Department of Software Engineering*  **Quiz 1 - Spring 2023** |  |



COURSE TITLE: **SOFTWARE QUALITY ENGINEERING** COURSE CODE: **SEN-321**

Class: **BSE-6 (B)** Shift: **Morning**

Course Instructor: Sohaib ur RehmanTime Allowed:  **20 min.**

Date: 17th March 2023Max. Marks: **2.5 Marks**

**Question No. 1 [CLO1: 2.5 Marks]**

**Define the following:**

1. Software Quality Assurance
2. Verification and Validation
3. Walkthrough
4. [Integration Testing](https://softwaretestingfundamentals.com/integration-testing/)
5. Three categories/levels of McCall’s Model

Solution:

**Software Quality Assurance (SQA):** Software Quality Assurance refers to the systematic process of ensuring that software products and processes meet specified requirements and standards. It involves planning, monitoring, and controlling the activities and tasks throughout the software development lifecycle to ensure the desired level of quality is achieved. SQA encompasses various activities such as quality planning, quality control, quality improvement, and quality assessment.

**Verification and Validation**: Verification and Validation (V&V) are two distinct processes within the software development lifecycle.

Verification: Verification involves evaluating the software product or system to determine whether it complies with specified requirements. It focuses on checking whether the software is built correctly and meets the intended design specifications. Verification activities include inspections, reviews, walkthroughs, and static analysis techniques.

Validation: Validation, on the other hand, involves evaluating the software during or at the end of the development process to determine whether it satisfies the user's needs and expectations. It focuses on checking whether the software fulfills its intended purpose and performs as expected in the user's environment. Validation activities include testing, user acceptance testing, and beta testing.

**Walkthrough:** Walkthrough is a review process used in software development to gather feedback, identify defects, and ensure the understanding of the software product or document. It involves a step-by-step examination of the software or document, where the author or a designated presenter guides the participants through it, explaining the functionality, design, or requirements. Walkthroughs are typically interactive and encourage discussion, clarification of doubts, and identification of potential issues. The primary goal of a walkthrough is to improve the quality and accuracy of the software or document being reviewed.

**Integration Testing:** Integration testing is a software testing technique that focuses on testing the interactions and interfaces between different components or modules of a software system. It aims to identify defects that may arise when multiple components are integrated and interact with each other. Integration testing ensures that the integrated components work together as expected, follow defined protocols, and exchange data correctly. This type of testing helps uncover issues related to data communication, interoperability, performance, and behavior when components are combined.

**Three categories/levels of McCall's Model:**

McCall's Model is a software quality model that classifies quality characteristics into three categories:

Product Operation: This category focuses on the software's functionality and how well it meets the user's requirements. It includes aspects such as correctness, reliability, efficiency, integrity, usability, and maintainability.

Product Revision: This category focuses on the software's ability to undergo changes and adaptations. It includes characteristics such as modifiability, flexibility, testability, portability, and reusability.

Product Transition: This category focuses on the software's ability to transition from one operational environment to another. It includes characteristics such as installability, interoperability, adaptability, and conformance to standards.