

CSE 423: Software Engineering

Software Testing

Tajkia Nuri Ananna

May 29, 2023

Lecturer, Metropolitan University

1. Software testing

Software Testing

Definition

- Software testing is a procedure of implementing software or the application to identify the defects or bugs.

Software Testing Principles

Software Testing Principles

- All the tests should meet the customer requirements.
- To make our software testing should be performed by a third party.
- Exhaustive testing is not possible. As we need the optimal amount of testing based on the risk assessment of the application.
- All the tests to be conducted should be planned before implementing it.
- It follows the Pareto rule (80/20 rule) which states that 80% of errors come from 20% of program components/modules.
- Start testing with small parts and extend it to large parts.

Advantages of Software Testing Life Cycle (STLC)

Advantages of Software Testing Life Cycle (STLC)

- The testing team gets involved right from the initial phases of SDLC. This helps in a better understanding of the application which results in overall better testing.
]
- It improves the quality of each SDLC phase and also adds to transparency of each process.
- STLC ensures lesser project cost and timely delivery since the testing team can uncover bugs and anomalies quickly by following the systematic test process.
- Since each STLC phase has a well-defined goal and deliverable so it helps in measuring the progress with the different milestones achieved in each phase.

Software Testing Life Cycle (STLC)

There are 6 phases in Software Testing Life Cycle

Requirement Analysis

- First step
- Tester analyses requirement document of SDLC to examine requirements stated by the client
- After examining the requirements, the tester makes a test plan to check whether the software is meeting the requirements or not.
- The scope of testing is defined as to what all features can and will be tested and what all features will be out of the scope of the testing activities

Test Planning and Control

- Test planning: defining the test specifications in order to achieve the project requirements
- Test Control: Continuous monitoring of test progress with the set plan and escalating any deviation to the concerned stakeholders.

Test Case Development

- Actual test case creation.
- Also involves the specification of test data and automated test script creation using different automation tools like Selenium WebDriver, Katalon Studio, etc.

Test Environment Setup

- Creation of a test environment closely simulating the real-world environment.
- The testing team uses this environment to test the whole application.

Test Execution

- Manual and automated test case execution
- Any deviation from the expected result leads to the creation of defects in a defect management tool or manual logging of bugs in an excel sheet.
- Development team fixes the bugs, the QA team retests the bugs for validation.

Test Closure

- Marks the formal closure of testing
- Checking if all the project deliverables are delivered, archiving the testware (everything involved in testing like test plan, test case, automation scripts), test environment, and documenting the learning.

Software Quality Assurance vs Software Testing

Objective and Focus

- Quality Assurance: Proactive process that focuses on preventing defects and ensuring that the development process follows established standards and best practices. It involves activities aimed at maintaining and improving the overall quality of the software throughout its lifecycle.
- Testing: Reactive process that involves executing the software to identify defects and ensure that it functions as expected. It focuses on finding errors, bugs, and issues in the software application.

Software Quality Assurance vs Software Testing

Timing

- Quality Assurance: QA activities are typically performed throughout the entire software development lifecycle, from the initial planning and requirements gathering phases to post-release maintenance. QA professionals are involved in defining processes, standards, and guidelines, as well as conducting audits and reviews.
- Testing: Testing activities are usually performed during specific stages of the development process, such as the verification of individual components or the overall system. It often occurs after the development phase and before the release of the software.

Software Quality Assurance vs Software Testing

Scope

- Quality Assurance: QA is concerned with the overall quality of the software product and the processes used to create it. It encompasses activities like establishing quality standards, defining testing methodologies, conducting reviews, and ensuring compliance with industry regulations.
- Testing: Testing focuses on identifying defects and ensuring that the software meets functional and non-functional requirements. It involves activities like test planning, test case design, test execution, and defect reporting.

Software Quality Assurance vs Software Testing

Responsibility

- Quality Assurance: QA is a responsibility shared by the entire development team, including project managers, developers, and testers. QA professionals provide guidance, establish processes, and monitor compliance with quality standards.
- Testing: Testing is primarily the responsibility of dedicated testers or testing teams. Their main role is to execute test cases, report defects, and ensure that the software meets the specified requirements.

Software Quality Assurance vs Software Testing

Goal

- Quality Assurance: The goal of QA is to prevent defects by identifying and addressing issues early in the development process. It aims to improve the efficiency, effectiveness, and reliability of the software development process.
- Testing: The main goal of testing is to identify defects and ensure that the software meets the defined requirements. It aims to validate the functionality, performance, usability, and reliability of the software.

Black box testing vs White box testing

Black box testing

- The tester has no knowledge about the internal workings of the code
- Testing is done from end user perspective
- Test cases are designed based on the SRS
- Allows independent testing, as it does not require knowledge of the internal workings of the system.
- Limitation: May not cover all possible paths within the system

Black box testing vs White box testing

White box testing

- The tester has knowledge about the internal workings of the code
- The internal logics, functions and codes are inspected
- Testing is done from developer perspective
- Test cases are designed based on internal structure and code
- Allows in-depth testing of the internal workings of the system
- Limitations: Requires knowledge of the internal structure and code, which may limit its independence.

Levels of testing

Unit Testing

The first level of testing involves analyzing each unit or an individual component of the software application.

Integration Testing

- It is mainly used to test the data flow from one module or component to other modules
- Multiple functionalities are tested to test compatibility
- When each component or module works separately, we need to check the data flow between the dependent modules, and this process is known as integration testing.

Levels of testing

System Testing

It is end-to-end testing where the testing environment is parallel to the production environment. In the third level of software testing, we will test the application as a whole system.

Acceptance Testing

The last and fourth level of software testing is acceptance testing, which is used to evaluate whether a specification or the requirements are met as per its delivery.

Automation Testing vs Manual Testing

Automation Testing

Process of changing any manual test case into the test scripts by using automation testing tools, and scripting or programming language is called automation.

Manual Testing

- Manual testing is testing, where the tester can test the application without any knowledge of any programming language
- Manual test engineers always search for the fault or bugs in the product before the product released in the market, yet delivered software still has defects

