

Assignment

Day 2

Monday, 4 September 2023

By: Ibrahim Tarek

SV G2 / Intake #3

Table of Contents

1. Executive Summary	3
2. White Box vs Black Box Testing	4
2.1. White Box Testing Definition.....	4
2.2. Black Box Testing Definition	4
2.3. Differences between White and Black box testing.....	4
3. STLC.....	6
3.1. Definition.....	6
3.2. Role of STLC	6
3.3. STLC Phases	7
3.4. SDLC vs STLC	8
4. References	9

1. Executive Summary

This report discusses each of the following topics:

- 1- What is the difference between white box testing and black box testing.
- 2- What is STLC.

2. White Box vs Black Box Testing

Black box testing and white box testing are two different approaches to software testing.

2.1. White Box Testing Definition

is a software testing method in which the internal structure/design/implementation of the item being tested is known to the tester. Implementation and impact of the code are tested.

2.2. Black Box Testing Definition

is a software testing method in which the internal structure/design/implementation of the item being tested is not known to the tester. Only the external design and structure are tested.

2.3. Differences between White and Black box testing

Black box testing is a testing technique in which the internal workings of the software are not known to the tester. The tester only focuses on the input and output of the software. Whereas, White box testing is a testing technique in which the tester has knowledge of the internal workings of the software, and can test individual code snippets, algorithms and methods.

2.3.1. Testing objectives

Black box testing is mainly focused on testing the functionality of the software, ensuring that it meets the requirements and specifications. White box testing is mainly focused on ensuring that the internal code of the software is correct and efficient.

2.3.2. Knowledge level:

Black box testing does not require any knowledge of the internal workings of the software, and can be performed by testers who are not familiar with programming languages. White box testing requires knowledge of programming languages, software architecture and design patterns.

2.3.3. Testing methods:

Black box testing uses methods like equivalence partitioning, boundary value analysis, and error guessing to create test cases. Whereas, white box testing uses methods like control flow testing, data flow testing and statement coverage.

2.3.4. Scope:

Black box testing is generally used for testing the software at the functional level. White box testing is used for testing the software at the unit level, integration level and system level.

So we can consider that black box testing is It is mostly done by software testers, on the other hand white box testing is mostly done by software developers.

3. STLC

Software Testing Life Cycle (STLC) is a testing strategy that can help you efficiently meet software quality standards. STLC enforces systematic testing, which is performed in phases. STLC is often confused with Software Development Life Cycle (SDLC), but while STLC is focused on testing, SDLC encompasses all development stages. Read on for an in-depth look at STLC and its six phases.

3.1. Definition

Software Testing Life Cycle (STLC) is a process used to test software and ensure that quality standards are met. Tests are carried out systematically over several phases. During product development, phases of the STLC may be performed multiple times until a product is deemed suitable for release.

3.2. Role of STLC

An STLC helps teams:

- Refine the agile testing process, increasing consistency and effectiveness.
- Clearly define goals and expectations for each project aspect.
- Apply time constraints to project testing.
- Ensure that each feature is tested and passing before additional features are added.
- Verify that project requirements are met.

Some common activities included in a Software Testing Life Cycle framework are:

- Analysis of client and stakeholder system requirements.
- Creation of a traceability matrix (a document that determines project completeness by comparing requirements with features).
- Identification of testing techniques and types needed for each feature.
- Prioritization of features to be targeted.
- Analysis of how tasks can be automated.
- Identification of environment information.

Some advantage of STLC include:

- Testing is shifted left in the development cycle. This ensures that tests are designed meaningfully and that feature specs are clear.
- Systematic testing leads to a faster testing process and enables teams to uncover and address issues faster.
- Project progress is easier to track since testing is performed with well-defined goals and phases.

3.3. STLC Phases

Software Testing Life Cycle consists of 6 phases, each with defined entry and exit criteria and associated activities and deliverables. Entry and exit criteria define when a phase can start and when it can end. Activities and deliverables define what actions are performed and what the expected result is. Some of these phases can be performed simultaneously while others require previous phases to be completed first.

3.3.1. Requirement Analysis

During this phase, feature requirements collected in the SDLC process are evaluated to identify testable aspects. If necessary, testing teams may need to consult with stakeholders to clarify requirements. These requirements can either be functional or non-functional, defining what a feature can do or its characteristics respectively. The ability to automate testing is also evaluated during this phase.

- Entry Criteria—documented requirements, acceptance criteria, and intended product architecture.
- Exit Criteria—approved requirement traceability matrix (RTM) and automation feasibility report.

3.3.2. Test Planning

During this phase, the test strategy is outlined in a test plan document. This strategy includes tools needed, testing steps, and roles and responsibilities. Part of determining this strategy is a risk and cost analysis and an estimated timeline for testing.

- Entry Criteria—requirement analysis, RTM, and automation feasibility report.
- Exit Criteria—approved test plan including timelines and risk/cost analysis.

3.3.3. Test Case Development

During this phase, test cases are created. Each case defines test inputs, procedures, execution conditions, and anticipated results. Test cases should be transparent, efficient, and adaptable. Once all test cases are created, test coverage should be 100%. Any necessary automation scripts are also created during this phase.

- Entry Criteria—approved test plan including timelines and risk/cost analysis.
- Exit Criteria—approved test cases and automation scripts.

3.3.4. Test Environment Setup

During this phase, testing environments are configured and deployed. This phase may include a variety of testing tools, including TestComplete, Selenium, Appium, or Katalon Studio.

Sometimes, this phase also includes setting up test servers. Once environments are deployed, smoke tests are performed to ensure that environments are working as expected with all intended functionality.

- Entry Criteria: system design and project architecture definitions.
- Exit Criteria: a fully functional test environment and approved test cases.

3.3.5. Test Execution

During this phase, features are tested in the deployed environment, using the established test cases. Expected test results are compared to actual and results are gathered to report back to development teams.

- Entry Criteria—all exit criteria from previous steps.
- Exit Criteria—all tests are performed and results are documented.

3.3.6. Test Cycle Closure

This is the last phase of the STLC, during which a test result report is prepared. This report should summarize the entire testing process and provide comparisons between expected results and actual. These comparisons include objectives met, time taken, total costs, test coverage, and any defects found.

- Entry Criteria—test results and logging from all previous phases.
- Exit Criteria—delivered and approved test closure report.

3.4. SDLC vs STLC

You might have also heard about the software testing lifecycle (STLC). The STLC refers to the set of activities that ensure software quality by detecting bugs and defects before the product release. It has phases similar to the SDLC but with different objectives and deliverables.

There are several key differences between SDLC and STLC, such as:

1. **SDLC** is focused on software development, while **STLC** is focused on software testing.
2. **SDLC** aims to build a software product that meets the user requirements, while **STLC** aims to ensure that the software is bug-free and reliable.
3. **SDLC** consists of various phases, such as planning, design, coding, testing, and deployment, while **STLC** has different phases, such as test planning, test case development, test execution, and test closure.

4. References

- 4.1. <https://www.geeksforgeeks.org/differences-between-black-box-testing-vs-white-box-testing/>
- 4.2. <https://www.geeksforgeeks.org/software-testing-life-cycle-stlc/>
- 4.3. <https://theproductmanager.com/topics/software-development-life-cycle/>
- 4.4. <https://www.sealights.io/software-quality/an-introduction-to-software-testing-life-cycle-stlc-definition-and-phases/>