

BLACK BOX TESTING

FOR TRAINING INSTITUTE MANAGEMENT SYSTEM

What is Black Box Testing?

Black box testing is a software testing techniques in which functionality of the software under test (SUT) is tested without looking at the internal code structure, implementation details and knowledge of internal paths of the software. This type of testing is based entirely on the software requirements and specifications.

In BlackBox Testing we just focus on inputs and output of the software system without bothering about internal knowledge of the software program.



The above Black-Box can be any software system you want to test. For example: an operating system like Windows, a website like Google, a database like Oracle or even your own custom application. Under Black Box Testing, you can test these applications by just focusing on the inputs and outputs without knowing their internal code implementation.

Black Box Testing - Steps

Here are the generic steps followed to carry out any type of Black Box Testing.

- Initially requirements and specifications of the system are examined.
- Tester chooses valid inputs (positive test scenario) to check whether SUT processes them correctly . Also some invalid inputs (negative test scenario) are chosen to verify that the SUT is able to detect them.
- Tester determines expected outputs for all those inputs.
- Software tester constructs test cases with the selected inputs.
- The test cases are executed.
- Software tester compares the actual outputs with the expected outputs.

- Defects if any are fixed and re-tested.

Types of Black Box Testing

There are many types of Black Box Testing but following are the prominent ones -

- **Functional testing** - This black box testing type is related to functional requirements of a system; it is done by software testers.
- **Non-functional testing** - This type of black box testing is not related to testing of a specific functionality, but non-functional requirements such as performance, scalability, usability.
- **Regression testing** - Regression Testing is done after code fixes, upgrades or any other system maintenance to check the new code has not affected the existing code.

Tools used for Black Box Testing:

Tools used for Black box testing largely depends on the type of black box testing you are doing.

For Functional/ Regression Tests you can use - [QTP](#), [Selenium](#)

For Non-Functional Tests, you can use - [Loadrunner](#), [Jmeter](#)

Black box testing strategy:

Following are the prominent Test Strategy amongst the many used in Black box Testing

- **Equivalence Class Testing:** It is used to minimize the number of possible test cases to an optimum level while maintains reasonable test coverage.
- **Boundary Value Testing:** Boundary value testing is focused on the values at boundaries. This technique determines whether a certain range of values are acceptable by the system or not. It is very useful in reducing the number of test cases. It is mostly suitable for the systems where input is within certain ranges.
- **Decision Table Testing:** A decision table puts causes and their effects in a matrix. There is unique combination in each column.

Black box testing has its own life cycle called Software Test Life Cycle (STLC) and it is relative to every stage of Software Development Life Cycle.

- **Requirement** - This is the initial stage of SDLC and in this stage requirement is gathered. Software testers also take part in this stage.
- **Test Planning & Analysis** - Testing Types applicable to the project are determined. A Test Plan is created which determines possible project risks and their mitigation.
- **Design** - In this stage Test cases/scripts are created on the basis of software requirement documents
- **Test Execution**- In this stage Test Cases prepared are executed. Bugs if any are fixed and re-tested.

OUTPUT :

Check All possibility both positive and negative inputs for Login Page.