Experiment No: 07

● **Aim: write test cases for black box testing.**

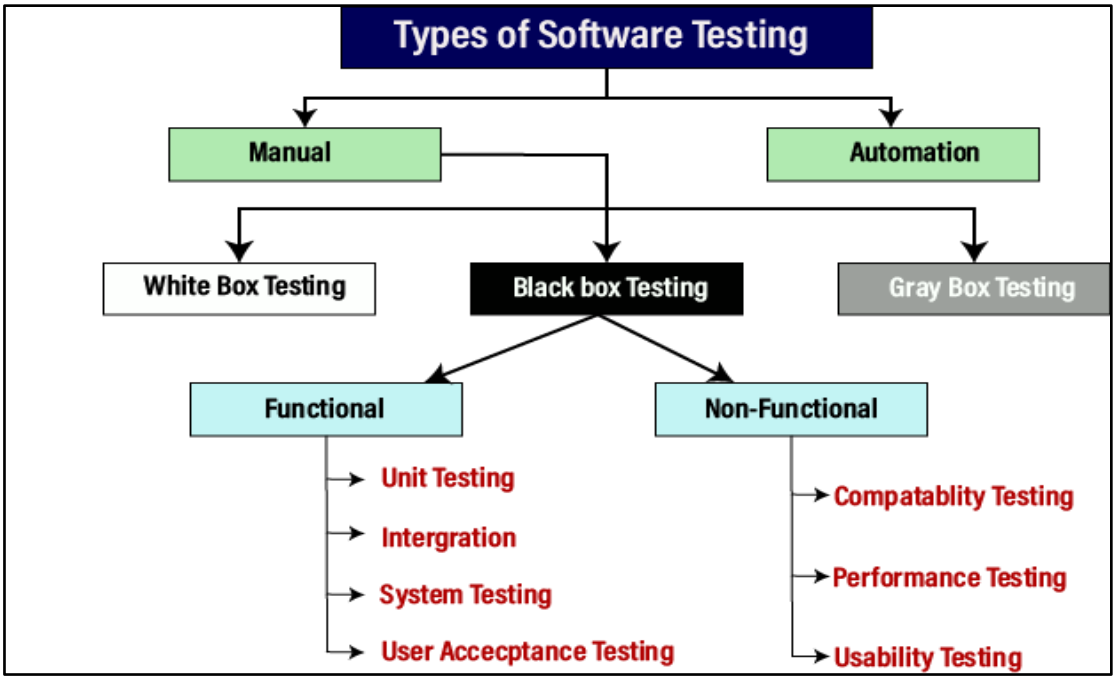
● **Theory:**

● **Project Name: - The QR CODE SCANNER**

Software testing:

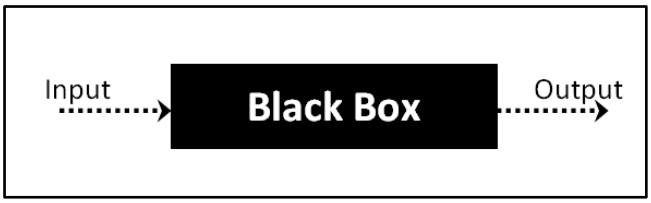
Software testing is a process of identifying the correctness of software by considering its all attributes (Reliability, Scalability, Portability, Re-usability, Usability) and evaluating the execution of software components to find the software bugs or errors or defects.

Type of Software testing:



Black box testing:

Black Box Testing is a software testing method in which the functionalities of software applications are tested without having knowledge of internal code structure, implementation details and internal paths. Black Box Testing mainly focuses on input and output of software applications and it is entirely based on software requirements and specifications. It is also known as Behavioural Testing.



Types of Black Box Testing:

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

- **Functional testing** – This black box testing type is related to the 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 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.

How to do Black Box testing?

When you get the basic understanding of black-box testing then the next question which comes up in mind is: How to perform the Black box testing? Below you can check the steps to perform this testing:

- The first step to black-box testing is to understand the requirement specifications of the application under test. An accurate and precise SRS document should be there.
- The next step is to evaluate the set of valid inputs and test scenarios to test the software. The goal is to save time and get good test coverage.
- Prepare the test cases to cover a maximum range of inputs.
- The test cases are run in the system to generate output, which is validated with the expected outcome to mark pass or fail.
- The failed steps are marked and sent to the development team to fix them.
- Retest the system using various testing techniques to verify its recurring nature or to pass it.
- The black box testing can be easily used to check and validate the entire software
- development life cycle. It can be used at various stages such as unit, integration, acceptance, system, and regression to evaluate the product.

Techniques Used in Black Box Testing:

- Decision Table Technique
- Boundary Value Technique
- State Transition Technique
- Equivalence partitioning
- Graph-Based Testing
- Error Guessing Technique.

Decision Table Technique:

Decision Table Technique is a systematic approach where various input combinations and their respective system behaviour are captured in a tabular form. It is appropriate for the functions that have a logical relationship between two and more than two inputs.

Boundary Value Technique:

Boundary Value Technique is used to test boundary values, boundary values are those that contain the upper and lower limit of a variable. It tests, while entering boundary value whether the software is producing correct output or not.

State Transition Technique:

State Transition Technique is used to capture the behaviour of the software application when different input values are given to the same function. This applies to those types of applications that provide the specific number of attempts to access the application.

Equivalence partitioning

Equivalence partitioning is a technique of software testing in which input data divided into partitions of valid and invalid values, and it is mandatory that all partitions must exhibit the same behaviour.

Graph-Based Testing:

It is similar to a decision-based test case design approach where the relationship between links and input cases are considered**.**

Error Guessing Technique.

Error guessing is a technique in which there is no specific method for identifying the error. It is based on the experience of the test analyst, where the tester uses the experience to guess the problematic areas of the software.

Testing

We follow manual testing here for the application, a type of software testing where testers manually write and execute test cases without using any automation tools. This method is the most primitive type and helps in finding bugs in the application code, ensuring that application is error free and is deliverable to the user/customer. Different Levels of Manual Testing Types of manual testing are:

- Unit Testing
- Integration Testing
- System Testing
- Acceptance Testing
- Black-box Testing
- White-box Testing

Black-Box Testing

As mentioned above, Black box testing is the testing method which is used to test the software without any knowledge of code or program. In short, the code under test environment is called as “Black-Box”. When performing the testing, the tester expects particular output/results, but not aware of how the application actually processes internally. Since this method is considered as the last step, it is to check whether the application is working as per user expectations.

Advantages of Black Box Testing:

- Efficient when used on large systems.
- Since the tester and developer are independent of each other, testing is balanced and unprejudiced.
- Tester can be non-technical.
- There is no need for the tester to have detailed functional knowledge of system.
- Testing helps to identify vagueness and contradictions in functional specifications.
- Test cases can be designed as soon as the functional specifications are complete.

Disadvantages of Black Box Testing:

- Test cases are challenging to design without having clear functional specifications.
- It is difficult to identify tricky inputs if the test cases are not developed based on specifications.
- It is difficult to identify all possible inputs in limited testing time. As a result, writing test cases may be slow and difficult.
- There are chances of having unidentified paths during the testing process.
- There is a high probability of repeating tests already performed by the programmer.

Pre-requisite:

1. Internet should be available.
2. Application should be installed on the device.

TEST CASE ID	FEATURE	STEPS TO EXECUTE	TEST DATA INPUT	EXPECTED RESULT	ACTUAL RESULT	STATUS
TC_001	Screen flash	1. Enter login Activity	Enter login Activity	Enter login Activity	Enter login Activity	Pass
TC_002	Login	1. Enter invalid login details. 2. Press Login button.	valid mobile no and password	user should be able to login	user is able to login	Pass
TC_003	Login	1. Enter invalid login details. 2. Press Login button.	valid mobile no and password	user should not be able to login.	user is unable to login	fail
TC_004	generate	1.click on "generate QR code btn?".	Generate	Inter text ,no, etc should be displayed.	Generate QR enter all credentials is displayed.	Pass
TC_005	generate	click on "generate	Not Generate	Inter text ,no, etc should not be displayed.	Generate QR enter all.	fail
TC_006	Scan	1. click on "Scan btn.	Scan	Open camera and scan	Display result.	Pass
TC_007	Scan	1. click on "Scan btn.	Scan	Not Open camera and scan	Display result.	fail

● Conclusion: -

Black box testing helps to find the gaps **in functionality, usability, and other features**. This form of testing gives an overview of software performance and its output. It improves software quality and reduces the time to market. This form of testing mitigates the risk of software failures at the user's end.