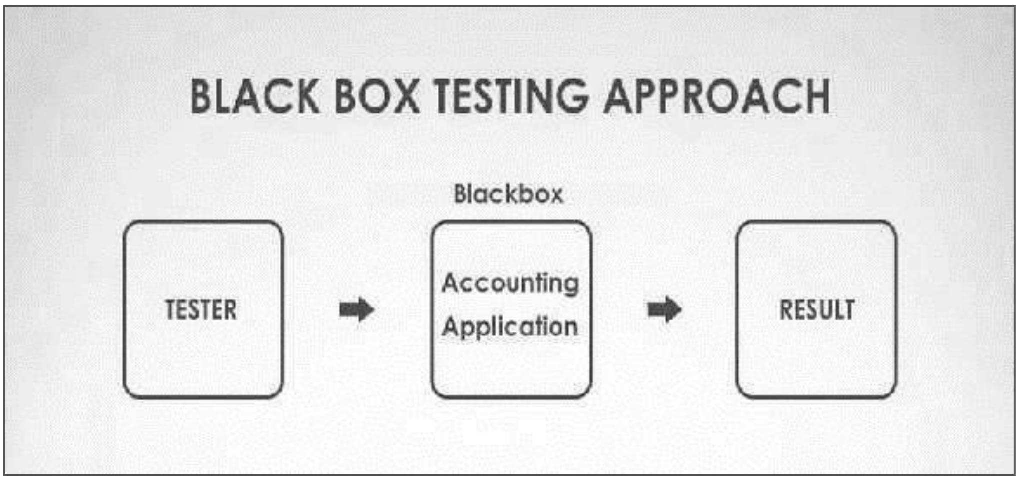
CHAPTER-6
TESTING

6.1 Testing Methods

6.1.1 Black Box Testing

Black Box Testing is also known as Behavioral Testing or Functional Testing. It is a technique of testing without having anyknowledge of the internal working of the application.

Black Box Testing treats the software as a “Black Box”-without any knowledge of internal working and it only examines the fundamental aspects of the system. This method of test can be applied to each and every level of software testing such as unit, integration, system and acceptance testing.



[Figure 7: Black Box Testing]

This method attempts to find errors in the following cases:

- Incorrect or missing functions
- Interface Errors
- Errors in structures or external database access
- Behavior or performance errors
- Initialization and termination errors

Advantages:

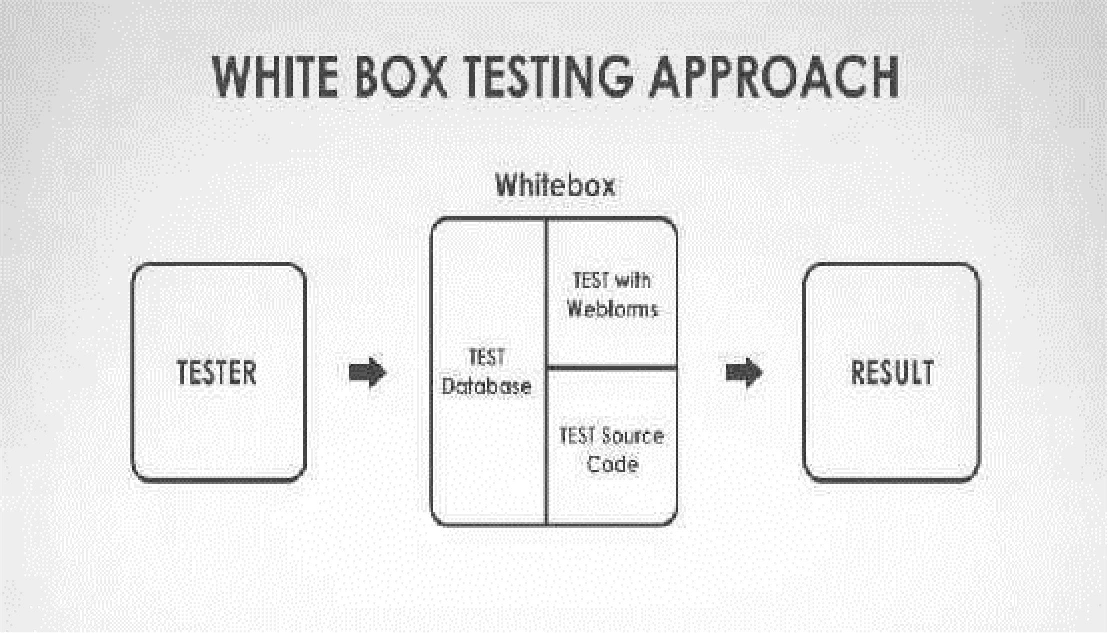
- Unbiased tests because the designer and tester work independently.
- Tester is free from any pressure of knowledge of specific programming languages to test the reliability and functionality of an application / software.
- Test is performed from a user's point-of-view and not of the designer's.
- Test cases can be designed immediately after the completion of specifications.

Disadvantages:

- Testing every possible input stream is not possible because it is time- consuming and this would eventually leave many program paths untested.
- Test cases are extremely difficult to be designed without clear and concise specifications.
- Results might be overestimated at time.
- Cannot be used for testing complex segments of code.

6.1.2 White Box Testing

White Box Testing is a software testing method in which the internal structure/design/implementation of the item being tested is known to the tester. The tester chooses inputs to exercise paths through the code and determines the appropriate outputs. This method is named so because the software program, in the eyes of the tester, is like a white/transparent box; inside which one clearly sees.



[Figure 8: White Box Testing]

The aim of this testing is to investigate the internal logic and structure of the code. That is why white box testing is also known as Structural Testing. Test Cases generated using White Box Testing can:

- Guarantee that all independent paths within a module have been exercised at least once.
- Exercise all decisions whether they are true or false.
- Exercise external data structure of the program.

Advantages:

- Code optimization by revealing hidden errors.
- Transparency of the internal coding structure which is helpful inderiving the type of input data needed to test an application effectively.
- Covers all possible paths of a code thereby, empowering a softwareengineering team to conduct thorough application testing.
- Enables programmer to introspect because developers can carefullydescribe any new implementation.
- Gives engineering-based rules to stop testing an application.

Disadvantages:

- Since tests can be very complex, highly skilled resources arerequired with a thorough knowledge of programming and implementation.
- Test script maintenance can be a burden if the implementationchanges too frequently.
- Necessity to create full range of inputs to test each path and condition make the white box testing method time-consuming.
- Exhaustive testing becomes even more complex using the whitebox testing method if the application is of large size.