SOFTWARE TESTING

BY

R.EZHILVEL

WHAT IS SOFTWARE TESTING?

- Software testing is the process of evaluating and verifying that a software product or application .

- The benefits of testing is preventing bugs, reducing development costs and improving performance.

METHOD OF TESTING

- There are two method of testing
- Static method(verification)
- Dynamic method(validation)

  manual testing

  automation testing

STATIC METHOD

- Static Testing is to check the defects in software without actually executing the code of the software application
- Static testing is performed in early stage of development to avoid errors as
- Static testing techniques offer a great way to enhance the quality and efficiency of software development.

REVIEW TESTING

- A review in a Static Testing is a process or meeting conducted to find the potential defects in the design of any program. Another significance of review is that all the team members get to know about the progress of the project
- In static testing, **reviews** can be divided into **four different parts**,
- **Informal reviews**
- **Walkthroughs**
- **Technical/peer review**
- **Inspections**

TYPES OF REVIEW

- Informal:

In informal review the creator of the documents put the contents in front of team member and everyone gives their opinion and thus defects are identified in the early stage.

- Walkthrough:

It is basically performed by experienced person or expert to check the defects so that there might not be problem further in the development or testing phase.

- Peer review/technical:

  Peer review means checking documents of one-another to detect and fix the defects. It is basically done in a team of colleagues.

- Inspection:

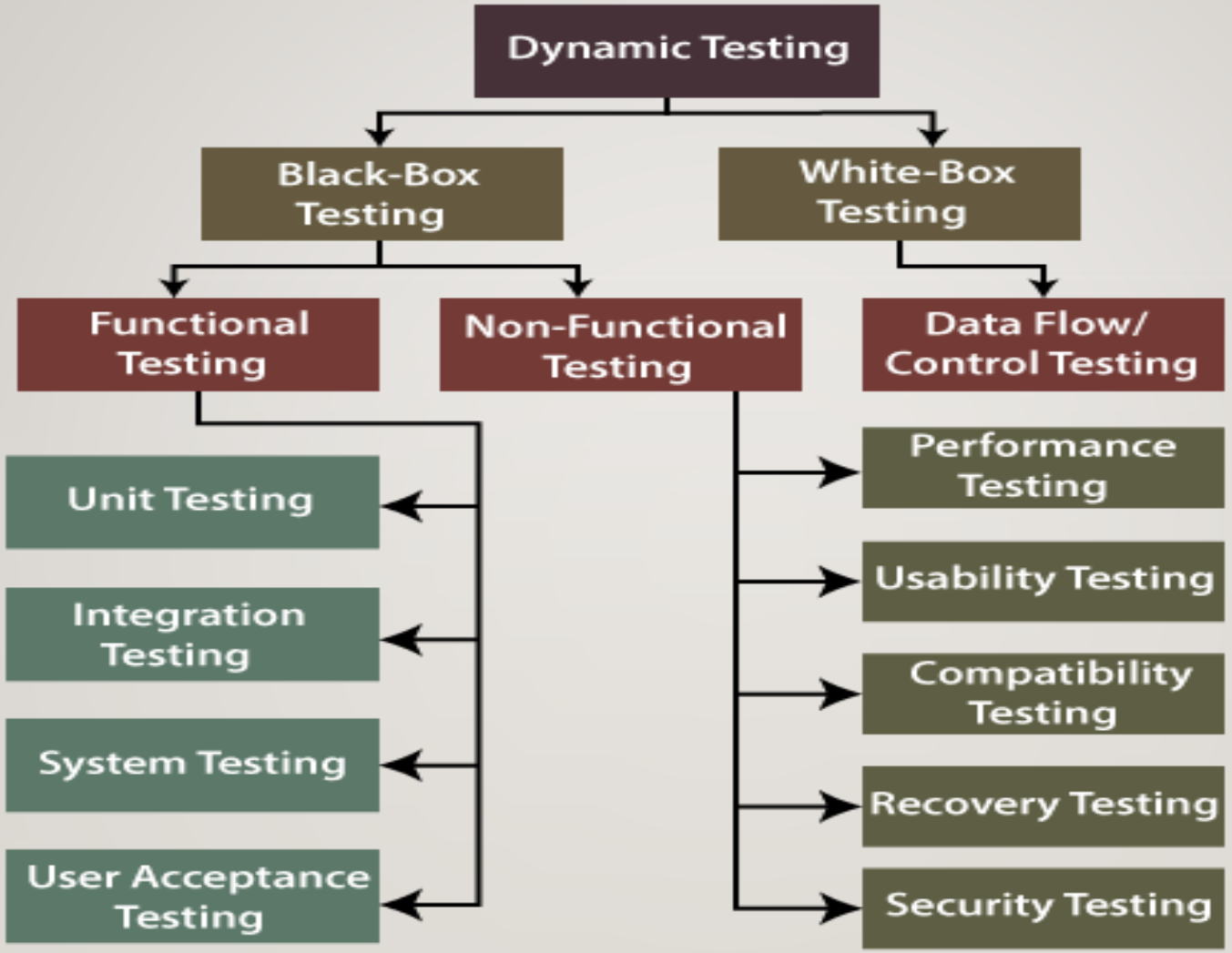
  Inspection is basically the verification of document the higher authority like the verification of software requirement specifications (SRS).

ADVANTAGES OF STATIC TESTING

- **Improved Product quality**
- **Improved the efficiency of Dynamic testing**
- **Reduced SDLC cost(software development life cycle)**
- **Immediate evaluation & feedback**
- **Exact location of bug is traced**

DYNAMIC TESTING

- Dynamic testing is one of the most important parts of Software testing, which is used to analyse the code's dynamic behavior.
- Why do we need to perform Dynamic Testing?
- We will perform dynamic testing to check whether the application or software is working fine during and after installing the application without any error.
- We can perform dynamic testing to verify the efficient behavior of the software.
- The software should be compiled and run if we want to perform dynamic testing.
- Dynamic testing are used in manual testing.


Dynamic Testing
Black-Box Testing
White-Box Testing
Functional Testing
Non-Functional Testing
Data Flow/ Control Testing
Unit Testing
Integration Testing
System Testing
User Acceptance Testing
Performance Testing
Usability Testing
Compatibility Testing
Recovery Testing
Security Testing

TYPES OF DYNAMIC TESTING

- Dynamic testing divided into two different testing approach,
- White-box testing : When the developers perform the White-box testing and then send the software application to the testing team, the testing team will do the black box testing, validate the application as well as the requirements.
- Black-box testing : is a testing technique where the test engineer selects a module and gives an input value to observe its functionality and analysis of whether the function is giving the expected output or not. If the function produced the correct output, then the particular function will be marked as pass.

- Black-box testing is further classified into two types, which are as follows:

- Functional testing:
- Non-function testing

FUNCTIONAL TESTING

- One of the most important parts of black-box testing. It mainly focuses on application specification rather than the actual code, and the test engineer will test the program rather than the system.
- The functional testing is used to validate the software application's functionality, whether the function is working as per the requirement specification.
- In functional testing, each module has been tested by giving the value, determining the output, and verifying the actual output with the expected value.

TYPES OF FUNCTIONAL TESTING

- The functional testing is classified into four different type of testing, which are as follows:
    - Unit testing
    - Integration testing
    - System testing
    - User acceptance testing

UNIT TESTING

- The unit testing is the first level of functional testing to perform any testing on the software application.
- We will start checking every component of the module or application independently one by one. And this process is known as **components testing**.
- The primary objective to perform unit testing is to test the correctness of remote code and validate the unit components with their performance.

INTEGRATION TESTING

- When we have successfully done the unit testing on the specific software, we will go for the integration testing. The integration testing will help us to combined individual units and tested as a group. And it is **the second level** of functional testing.
- When all the components or modules are working independently, we will check the data flow between the dependent modules, which is known as integration testing.
- The developers and the test engineer perform the integration testing. And the main purpose of the integration is to identify the faults in the interaction between the integrated units.

SYSTEM TESTING

- System testing is used to check the end-to-end flow of an application or the software as a user.
- System testing is also known as end-to-end testing as the testing environment is similar to the production environment.
- In the third level (system testing) of functional testing, we go through all the necessary modules of an application and check if the end features or the end business works fine, and test the product as a whole system.

USER ACCEPTANCE TESTING

- The user acceptance testing is performed to certify the system according to requirements. The customer or client does it before accepting the final product.
- In other words, we can say that the UAT is done by the customer (domain expert) for their satisfaction and check whether the application is working according to given business scenarios and real-time scenarios.
- It is the last level of functional testing, which is execute before releasing the software to the market or production environment where two or more end-users will involve.

NON- FUNCTIONAL TESTING

- **Non-functional testing** plays a vital role in customer satisfaction while testing the software or the application.
- It reduces the risk of production and related costs of the software, and it provides a thorough knowledge of product behavior and used technologies.
- **Performance testing**
- **Usability testing**
- **Compatibility testing**
- **Recovery testing**
- **Security testing**

PERFORMANCE TESTING

- The performance testing is the most importantly used type of **non-functional**
- Once the software is stable and moved to the production, and it may be accessed by multiple users concurrently, we will do **performance testing**.
- The **performance testing** is testing where we check the *behavior of an application by applying some load.*

USABILITY TESTING

- In usability testing, we will check the user-friendliness, efficiency, and accuracy of the software application.
- If we are using usability testing, it ensures that the developed software is easy to test while using the system without facing any problem and makes end-user life easier.

COMPATIBILITY TESTING

- The next type of **non-functional testing** is **compatibility testing**, which is used to check the functionality of an application on different **software, hardware platforms, network, and browsers.**
- The compatibility testing is not performed for all the applications; we will use the compatibility testing only for those applications where we don't have control over the platform used by users.

RECOVERY TESTING

- In **recovery testing**, we can verify how well a system can recover from hardware failures and crashes.
- It reproduced the failure modes or essential producing failures in a controlled environment.
- The recovery testing is performed to confirm that a system is fault-tolerant and can improve well from failure

SECURITY TESTING

- The security testing is used to discover the weaknesses, risks, or threats in the software application and help us stop the nasty attack from outsiders and ensure our software applications' security.
- The main purpose of security testing is to identify all the possible uncertainties and vulnerabilities of the application so that the software does not stop working.

THE ADVANTAGES OF DYNAMIC TESTING:

- It helps the testing team to identify the weak areas of the run-time environment.
- The most important benefit of using **dynamic testing** over static testing is the relatively higher number of bugs can be found.
- As compared to **static testing, dynamic testing** requires a smaller number of meetings at the planning level of testing.
- It implements the software, end to end, and delivers Bug-free software.
- It becomes an essential tool for identifying any security threats.

DISADVANTAGES

- It is **a time-consuming** process as it implements the software application or code, which needs a massive resource.
- The dynamic testing process is **a bit costlier** as it increases the budget of the software.
- The dynamic testing needs more human resources to complete the task, which makes its implementation costlier.
- Generally, dynamic testing is executed after the coding phase is completed, and therefore, the bugs are identified later in the life cycle.

AUTOMATION TESTING

- **Automation Testing** is a software testing technique that performs using special automated testing software tools to execute a test case suite.
- Why Test Automation?
- Manual Testing of all workflows, all fields, all negative scenarios is time and money consuming
- It is difficult to test for multilingual sites manually
- Test Automation in software testing does not require Human intervention. You can run automated test unattended (overnight)
- Test Automation increases the speed of test execution
- Automation helps increase Test Coverage

TEST AUTOMATION PROCESS

- Test Tool Selection:

There will be some criteria for Selection of tool. Majority of the criteria includes : Do we have skilled resource to allocate for automation tasks, Budget constraints, Do the tool satisfy our needs ?

- Define Scope of Automation:

This includes few basic points such as : Framework should support Automation Scripts, Less Maintenance must be there, High Return on Investment, Not much complex Test Cases

- Planning, Design and Development:

For this we need to Install particular frameworks or libraries, start designing and developing the test cases such as NUnit, JUnit, QUnit or required Software Automation Tools

- Test Execution:

Final Execution of test cases will take place in this phase and it depends on Language to Language such as for .NET, we'll be using NUnit, for Java, we'll be using JUnit, for JavaScript, we'll be using QUnit or Jasmine, etc.

- Maintenance:

Creation of Reports generated after Tests and that should be documented so as to refer that in future for next iterations

POPULAR AUTOMATION TOOLS:

- Selenium
- QTP
- Sikuli
- Appium
- Jmeter
- TestNG
- Zephyr
- UFT

ADVANTAGES OF AUTOMATION TESTING:

- Simplifies Test Case Execution
- Improves Reliability of Tests
- Increases amount of test coverage
- Minimizing Human Interaction
- Saves Time and Money
- Test Results are made public

SOME OTHER TYPES OF SOFTWARE TESTING

- In software testing, we also have some other types of testing that are not part of any above discussed testing, but those testing are required while testing any software or an application.
- **Smoke Testing**
- **Sanity Testing**
- **Regression Testing**
- **Adhoc Testing**

- **smoke testing**

we will test an application's basic and critical features before doing one round of deep and rigorous testing.

Analyzing the workflow of the application's core and main functions is the main objective of performing the smoke testing.

- **Sanity Testing**

Sanity testing was performed when we are receiving software build (with minor code changes) from the development team. It is a checkpoint to assess if testing for the build can proceed or not.

- **REGRESSION TESTING**

Whenever a bug is fixed by the developers and then testing the other features of the applications that might be simulated because of the bug fixing is known as regression testing.

we can say that whenever there is a new release for some project, then we can perform Regression Testing, and due to a new feature may affect the old features in the earlier releases.

Regression testing is the most suitable testing for automation tools. As per the project type and accessibility of resources, regression testing can be similar to Retesting.

- **Adhoc Testing**

Testing the application randomly as soon as the build is in the checked sequence is known as Adhoc testing.

CONCLUSION

- We have discussed the most commonly used types of Software Testing like black-box testing, white box testing, functional testing, non-functional testing, regression testing, Adhoc testing, etc.

THANK YOU