**Module–2(Manual Testing**

**What is software testing?**

Software testing is a process to find any defects from the application or products.

Or

Software testing is a process of correctness, completeness and quality of development of computer software

**• What is Exploratory Testing?**

Exploratory Testing is a type of software testing where Test cases are not created in advance but testers check system on the fly.

**• What is traceability matrix?**

It is used to track the requirement and to check the current project requirement are meeting or not this we call traceability matrix.

**• What is Boundary value testing?**

Boundary Value Testing is used is Black Box Testing. Boundary Value testing is the process of testing between extreme ends or boundaries between partitions of the input values.

**• What is Equivalence partitioning testing?**

Equivalence partitioning or equivalence class partitioning (ECP) is a software testing technique that divides the input data of a software unit into partitions of equivalent data from which test cases can be derived.

**• What is Integration testing?**

Integration testing -- also known as integration and testing (I &T) -- is a type of software testing in which the different units, modules or components of a software application are tested as a combined entity.

**• What determines the level of risk?**

The level of risk is determine by it’s a combination of Probability and Severity.

**What is alpha testing?**

**Alpha Testing** is a type of software testing performed to identify bugs before releasing the product to real users or to the public. Alpha Testing is one of the **user acceptance testing**.

**• What is beta testing?**

Beta testing is an opportunity for real users to use a product in a production environment to uncover any bugs or issues before a general release.

**• What is component testing?**

Component Testing is also referred to **as Unit Testing, Program Testing or Module Testing**.

Component testing is defined as a software testing type, in which the testing is performed on each individual component separately without integrating with other components.

**• What is functional system testing?**

Functional testing is a type of testing that seeks to establish whether each application feature works as per the software requirements.

**• What is Non-Functional Testing?**

**Non-Functional Testing** is defined as a type of Software testing to check non-functional aspects (performance, usability, reliability, etc.) of a software application.

**• What is GUI Testing?**

GUI stands for Graphical User Interface where you interact with the computer using images rather than text.

**• What is Ad hoc testing?**

**Ad hoc Testing** is an informal or unstructured software testing type that aims to break the testing process in order to find possible defects or errors at an early possible stage.

**• What is white box testing and list the types of white box testing?**

**White Box Testing** is software testing technique in which internal structure, design and coding of software are tested to verify flow of input-output and to improve design, usability and security.

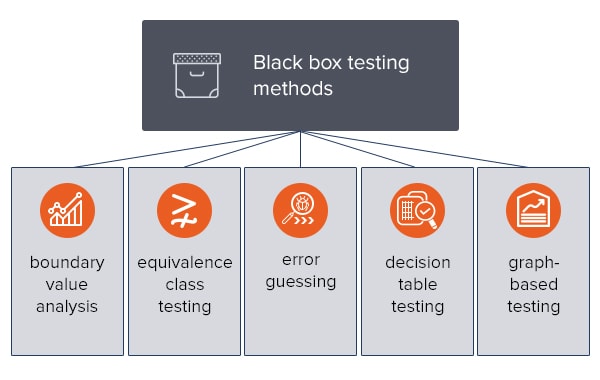
**Types of white box testing:**

* Statement Coverage.
* Branch Coverage.
* Path Coverage.

**• What is 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.

**What are the different black box testing techniques?**



**• Mention what are the categories of defects?**

Following are the common types of defects that occur during development:

* Arithmetic Defects
* Logical Defects
* Syntax Defects
* Multithreading Defects
* Interface Defects
* Performance Defects

**• Mention what big bang testing is?**

Big Bang Integration Testing is an integration testing strategy wherein all units are linked at once, resulting in a complete system.

**• What is the purpose of exit criteria?**

Exit criterion is used to determine whether a given test activity has been completed or NOT.

**• When should "Regression Testing" be performed**?

Regression testing is performed to find out whether the updates or changes had caused new defects in the existing functions.

**• What is 7 key principles? Explain in detail?**

1. Testing shows presence of Defects

2. Exhaustive Testing is Impossible.

3. Early Testing

4. Defect Clustering

5. The Pesticide Paradox

6. Testing is Context Dependent

7. Absence of Errors Fallacy

**Testing shows presence of defects:**

Testing can show that defects are present, but cannot prove that there are no defects. Testing reduces the probability of undiscovered defects remaining in the software but, even if no defects are found, it is not a proof of correctness. We test to find Faults As we find more defects, the probability of undiscovered defects remaining in a system reduces.

**Exhaustive testing is impossible:**

Testing everything including all combinations of inputs and preconditions is not possible. So, instead of doing the exhaustive testing we can use risks and priorities to focus testing efforts.

**Early testing:**

Testing activities should start as early as possible in the development life cycle .These activities should be focused on defined objectives – outlined in the Test Strategy.

**Defect clustering:**

A small number of modules contain most of the defects discovered during pre-release testing, or are responsible for the most operational failures. Defects are not evenly spread in a system they are ‘clustered’.

**The pesticide paradox:**

If the same tests are repeated over and over again, eventually the same set of test cases will no longer find any new defects. To overcome this “pesticide paradox”, the test cases need to be regularly reviewed and revised, and new and different tests need to be written to exercise different parts of the software or system to potentially find more defects.

**Testing is context dependent:**

Testing is basically context dependent. Testing is done differently in different contexts. Different kinds of sites are tested differently.

**Absence of error fallacy:**

If the system built is unusable and does not fulfill the user’s needs and expectations then finding and fixing defects does not help.

**• Difference between QA v/s QC v/s Tester**

**QA (quality assurance)**

* Subset of SDLC.
* Process oriented.
* Prevent defects.
* Whole team approach.
* Proactive process.
* Focus on process rather than conducting the actual process.

**QC (control quality)**

* Subset of QA.
* Product oriented.
* Prevent defects.
* Testing team.
* Reactive process.
* Focus on actual process.

**Tester**

* Subset of QC.
* Product oriented.
* Find and fix defects.
* Testing team.
* Reactive process.
* Focus on actual testing the product.

**• Difference between Smoke and Sanity?**

**Smoke testing:**

* This testing is perform by developers as well as by tester.
* It is usually scripted or documented.
* It’s a subset of acceptance testing.
* It check the entire processes from end to end.
* It perform in critical functionalities.

**Sanity testing:**

* This testing is perform b tester.
* It is not documented or scripted.
* It is a subset of regression testing.
* It check only some particular components.
* It perform in new functionalities.

**• Difference between verification and Validation**

**Verification:**

It is a static testing.

It comes before validation.

It consists of documentation.

Method use in this process is reviews, walkthrough, and inspection.

It does not execution of code.

**Validation:**

It is a dynamic testing.

It comes after verification.

It consists of programs.

Method use in this process is functional, non-functional and also use cases.

It execute code.

**• Explain types of Performance testing.**

**Load testing**: Basically, load testing determines the behavior of the application when multiple users use it at the same time. It is the response of the system measured under varying load conditions. The load testing is carried out for normal and extreme load conditions.

**Stress testing**:  Stress testing is defined as a type of software testing that verifies the stability and reliability of the system.

**Endurance testing**: Endurance testing refers to tests typically done to find out whether an application can withstand the processing load it is expected to have to endure for a long period.

**Spike testing**: It determine the behavior of a software application when it receives extreme variations in traffic.

**Volume testing**: Volume Testing is a type of software testing that is performed to test the performance or behavior of the system or application under the huge amount of data.

**Scalability testing:** **It** is a type of non-functional testing in which the performance of a software application, system, network or process is tested in terms of its capability to scale up or scale down the number of user request load or other such performance attributes.

**• What is Error, Defect, Bug and failure?**

**Error:** A discrepancy between a computed, observed, or measured value and condition and the true, specified, or theoretically correct value or condition. This can be a misunderstanding of the internal state of the software, an oversight in terms of memory management, confusion about the proper way to calculate a value, etc.

**Defect:** Commonly refers to several troubles with the software products, with its external behavior or with its internal features.

**Bug**: A fault in a program which causes the program to perform in an unintended or unanticipated manner. See: anomaly, defect, error, exception, and fault. Bug is terminology of Tester.

**Failure**: The inability of a system or component to perform its required functions within specified performance requirements. See: bug, crash, exception, and fault.

**• Difference between Priority and Severity**

**Priority:**

Defect Priority has defined the order in which the developer should resolve a defect.

Priority is associated with scheduling.

Priority indicates how soon the bug should be fixed.

Priority is driven by business value.

Priority status is based on customer requirements.

Priority is categorized into three types

* + Low
  + Medium
  + High

**Severity:**

Defect Severity is defined as the degree of impact that a defect has on the operation of the product.

Severity is associated with functionality or standards.

Severity indicates the seriousness of the defect on the product functionality.

Severity is driven by functionality.

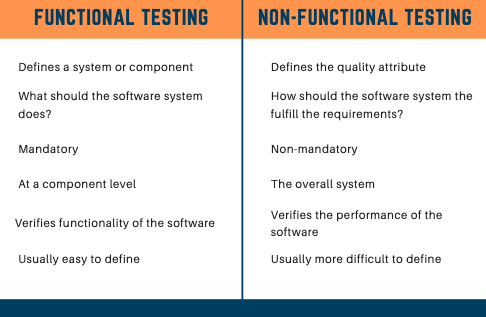
Severity status is based on the technical aspect of the product.

* Severity is categorized into five types
  + Critical
  + Major
  + Moderate
  + Minor
  + Cosmetic.

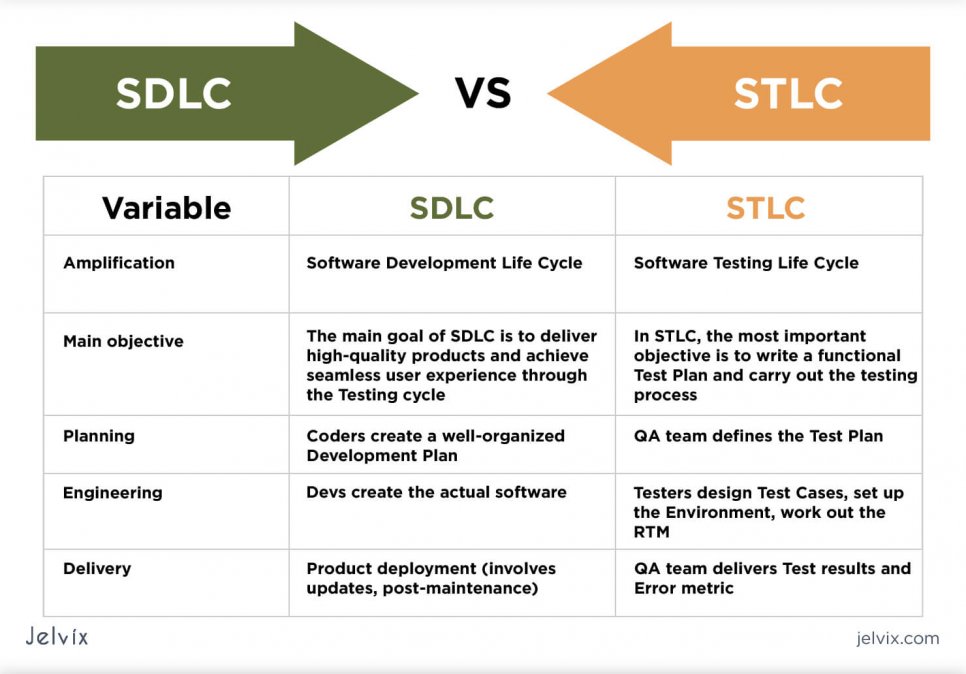
• **What is Bug Life Cycle?**

“A computer bug is an error, flaw, mistake, failure, or fault in a computer program that prevents it from working correctly or produces an incorrect result. Bugs arise from mistakes and errors, made by people, in either a program’s source code or its design.”

**• Explain the difference between Functional testing and Nonfunctional testing.**



**What is the difference between the STLC (Software Testing Life Cycle) and SDLC (Software Development life cycle)?**



**What is the difference between test scenarios, test cases, and test script?**

**Test scenarios:**

A Test Scenario is **a statement describing the functionality of the application to be tested**. It is used for end-to-end testing of a feature and is generally derived from the use cases. Test scenarios can serve as the basis for lower-level test case creation. A single test scenario can cover one or more test cases.

**Test case:**

A test case is a defined format for software testing required to check if a particular application/software is working or not. A test case consists of a certain set of conditions that need to be checked to test an application or software.

**Test script:**

A test script in software testing is **a set of instructions that will be performed on the system under test to test that the system functions as expected**. There are various means for executing test scripts. Manual testing. These are more commonly called test cases.

**Explain what Test Plan is?**

 “Test Plan is a document describing the scope, approach, resources, and schedule of intended test activities.”

A **Test Plan** is a detailed document that describes the test strategy, objectives, schedule, estimation, deliverables, and resources required to perform testing for a software product. Test Plan helps us determine the effort needed to validate the quality of the application under test. The test plan serves as a blueprint to conduct software testing activities as a defined process, which is minutely monitored and controlled by the test manager.

**What is the information that should be covered?**

The information that should be covered in Test Plan are:

Analyze the product.

Design the Test Strategy.

Define the Test Objectives.

Define Test Criteria.

Resource Planning.

Plan Test Environment.

Schedule & Estimation.

Determine Test Deliverables.

**What are the different Methodologies in Agile Development Model?**

**The different Methodologies in Agile Development Model are:**

Scrum.

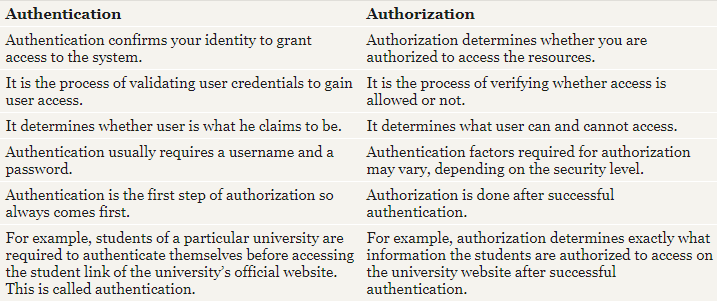
Kanban.

Extreme Programming (XP).

Lean Development.

Crystal.

**Explain the difference between Authorization and Authentication in Web testing.**



**What are the common problems faced in Web testing?**

Generally five type of common problems are faced in web testing, they are:

Integration.

Interoperability.

Security.

Performance.

Usability.