**Testing: -**

Testing is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not. In simple words, testing is executing a system in order to identify any gaps, errors, or missing requirements in contrary to the actual requirements.

According to ANSI/IEEE 1059 standard, Testing can be defined as - A process of analyzing a software item to detect the differences between existing and required conditions (that is defects/errors/bugs) and to evaluate the features of the software item.

An early start to testing reduces the cost and time to rework and produce error-free software that is delivered to the client. However in Software Development Life Cycle (SDLC), testing can be started from the Requirements Gathering phase and continued till the deployment of the software.

It also depends on the development model that is being used. For example, in the Waterfall model, formal testing is conducted in the testing phase; but in the incremental model, testing is performed at the end of every increment/iteration and the whole application is tested at the end.

Testing is done in different forms at every phase of SDLC −

* During the requirement gathering phase, the analysis and verification of requirements are also considered as testing.
* Reviewing the design in the design phase with the intent to improve the design is also considered as testing.
* Testing performed by a developer on completion of the code is also categorized as testing.

**Unit Testing: -**

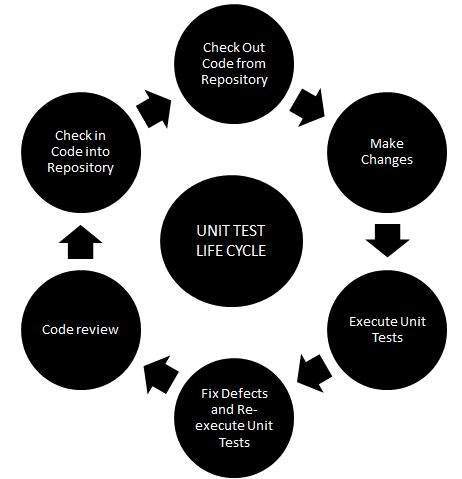
Unit testing, a testing technique using which individual modules are tested to determine if there are any issues by the developer himself. It is concerned with functional correctness of the standalone modules.

The main aim is to isolate each unit of the system to identify, analyze and fix the defects.

Unit Testing - Advantages:

* Reduces Defects in the Newly developed features or reduces bugs when changing the existing functionality.
* Reduces Cost of Testing as defects are captured in very early phase.
* Improves design and allows better refactoring of code.
* Unit Tests, when integrated with build gives the quality of the build as well.

## **Unit Testing LifeCyle: -**



**Unit Testing Techniques: -**

* Black Box Testing - Using which the user interface, input and output are tested.
* White Box Testing - used to test each one of those functions behaviour is tested.
* Gray Box Testing - Used to execute tests, risks and assessment methods.

**Integration Testing: -**

Upon completion of unit testing, the units or modules are to be integrated which gives raise to integration testing. The purpose of integration testing is to verify the functional, performance, and reliability between the modules that are integrated.

**Integration Strategies:**

* Big-Bang Integration
* Top Down Integration
* Bottom Up Integration
* Hybrid Integration

**Interface Testing: -**

Interface Testing is performed to evaluate whether systems or components pass data and control correctly to one another. It is to verify if all the interactions between these modules are working properly and errors are handled properly.

**Interface Testing – Checklist**

* Verify that communication between the systems are done correctly
* Verify if all supported hardware/software has been tested
* Verify if all linked documents be supported/opened on all platforms
* Verify the security requirements or encryption while communication happens between systems
* Check if a Solution can handle network failures between Web site and application server

**Internationalization Testing: -**

Internationalization testing is the process of verifying the application under test to work uniformly across multiple regions and cultures.

The main purpose of internationalization is to check if the code can handle all international support without breaking functionality that might cause data loss or data integrity issues. Globalization testing verifies if there is proper functionality of the product with any of the locale settings.

**Internationalization Checklists:**

* Testing to check if the product works across settings.
* Verifying the installation using various settings.
* Verify if the product works across language settings and currency settings.

## Isolation Testing: -

Isolation testing is the process of breaking down the system into various modules so that defects can be spotted easily in isolation. It happens especially when the bug is difficult to locate and resolve by development team.

**Characteristics of isolation Testing:**

* It is a time-consuming process and there should be stubs and drivers available to retest each one of them individually
* It is expensive as all the items are to be broken into several pieces to make it atomic in nature.
* It verifies the output of each one of those interfaces/subsystems precisely.

**Conclusion: -**

Above mentioned are the testing conducted and verified the application accordingly..