**What is software?**

Software is a collection of specialized programs which takes user input and generate desire o/p.

**Software** is a collection of instructions and data that tell a computer how to work.

is a set of instructions or programs instructing a computer to do specific tasks …

**What is testing?**

In general, testing **is finding out how well something works**

Testing is a process of checking whether the given software or application is generating the desired o/p.

**What is Software Testing ?**

**Software Testing** is a method to check whether the actual software product matches expected requirements and to ensure that software product is defect free.

Process of checking the completeness and correctness of an application with respect to the customer expectation/ customer requirement

Software testing is a validation process which confirms that a system works as per the business requirements. It qualifies a system on various aspects such as usability, accuracy, completeness, efficiency, etc.

Software testing is widely used technology because it is compulsory to test each and every software before deployment.

### What is Manual Testing?

Manual testing, as the term suggests, refers to a test process in which a QA manually tests the software application in order to identify bugs. To do so, QAs follow a written test plan that describes a set of unique test scenarios. The QA is required to analyze the performance of the web or mobile application from an end user’s perspective.

**Resources involved in the Software Development:**

* **Customer**
* **Business Analyst** (Gather Requirement)
* **Developer** ( Develop the application according to the requirement)
* **Tester** ( Check the correctness and completeness of the functionality)
* Final Product to the customer
* Customer want some product
* BA will gather the requirement from the customer
* BA then make the document and send it to the Developer
* Developer will then develop the application according to the customer requirement
* Then developed application send to the testing team for testing
* Tester then test the application positively & Negatively

* **Positive testing** means suppose there is a name field, tester will test the field by entering credentials (eg. Name:- Prasad)
* This +ve test is done in the correct manner.
* Another example, checking the system for a user name and password combination to login is a way of positive testing.
* Checks for valid set of values
* **Negative testing** is done by entering wrong credentials in the field. For name field tester enter the number / special characters. This number should not be accepted , this is the –ve testing.
* negative testing is done to break the project with respect to its different test conditions.
* Checks for invalid sets of values

**SQA( Software Quality Assurance)**

* is simply a way to assure quality in the software. It is the set of activities which ensure processes, procedures as well as standards suitable for the project and implemented correctly
* SQA is done for monitoring and measuring the software development factors.

**Factors involved in the SQA**

* **To meet the Customer Requirement**

Which Type of application customer want

i.e. Banking domain, Telecom domain

For which purpose customer want the software

* **To meet the customer expectation**

Privacy:- includes security, any software like Banking application, gathered customers data which is very sensitive, so client wants privacy to all the customers data

**Performance:-** Software should balance the load properly. It should have to work properly under heavy load

**Costing of Project:**

Project costing for MNCs are per hour cost. Customers have to pay for it.

This payment depend upon resources utilization as well as time to complete for project

**Timing Delivery:**

At the time of resources gathering and documentation time to complete for a project get decided

If a company exceeds the delivery time then the company have to pay a penalty that penalty is called escalation.

**Maintenance:**

Maintenance is the part of services provided by the company after delivery of the project

If any problem occurs after delivering the project, then the Company- dev team has to fix it.

**Projects:**

**Project have 2 categories ( this is just example)**

**Project**

**Critical Project** **Normal Project**

Eg. Banking domain project Eg. ERP System

Resources Requirement Resources Requirement

2 developer: 1 tester 3 developer: 1 tester

Like 100 dev: 50 tester Like 300 dev: 100 tester

**Service – based companies and Product – based companies**

**Service – based companies: -** They provide service and develop software for other companies They provide software which is and specified as per the client company’s requirement and never keep the code of the developed product and does not provide the software to any other company other than the client company. Ex – Wipro, Infosys, TCS, Accenture

**Product – based companies :-** The develop software products and sell it to many companies which may need the software and make profits for themselves They are the sole owners of the product they develop and the code used and sell it to other companies which may need the software. Ex – Oracle, Microsoft

Product based companies create or design their products or applications in advance even before clients approach them. Once the product is made or application is developed it is opened to the market. Service based companies work only when a client approaches them with specific needs or requirements

**Seven Principles of software testing**

1. Testing shows presence of defects
2. Exhaustive testing is not possible
3. Early testing
4. Defect clustering
5. Pesticide paradox
6. Testing is context dependent
7. Absence of errors fallacy

**Testing shows presence of defects**

The test engineer will test the application to make sure that the application is bug or defects free. While doing testing, we can only identify that the application or software has any errors. The primary purpose of doing testing is to identify the numbers of unknown bugs with the help of various methods and testing techniques because the entire test should be traceable to the customer requirement, which means that to find any defects that might cause the product failure to meet the client's needs.

By doing testing on any application, we can decrease the number of bugs, which does not mean that the application is defect-free because sometimes the software seems to be bug-free while performing multiple types of testing on it. But at the time of deployment in the production server, if the end-user encounters those bugs which are not found in the testing process.

**Exhaustive testing is not possible**

Sometimes it seems to be very hard to test all the modules and their features with effective and non- effective combinations of the inputs data throughout the actual testing process.

Hence, instead of performing the exhaustive testing as it takes boundless determinations and most of the hard work is unsuccessful. So we can complete this type of variations according to the importance of the modules because the product timelines will not permit us to perform such type of testing scenarios.

**Early testing**

Here early testing means that all the testing activities should start in the early stages of the software development life cycle's requirement analysis stage to identify the defects because if we find the bugs at an early stage, it will be fixed in the initial stage itself, which may cost us very less as compared to those which are identified in the future phase of the testing process.

To perform testing, we will require the requirement specification documents; therefore, if the requirements are defined incorrectly, then it can be fixed directly rather than fixing them in another stage, which could be the development phase.

**Defect clustering**

the defect clustering defined that throughout the testing process, we can detect the numbers of bugs which are correlated to a small number of modules. We have various reasons for this, such as the modules could be complicated; the coding part may be complex, and so on.

These types of software or the application will follow the Pareto Principle, which states that we can identify that approx. Eighty percent of the complication is present in 20 percent of the modules. With the help of this, we can find the uncertain modules, but this method has its difficulties if the same tests are performing regularly, hence the same test will not be able to identify the new defects.

**Pesticide paradox**

This principle defines that if we are executing the same set of test cases again and again over a particular time, then these kinds of the test will not be able to find the new bugs in the software or the application. To get over these pesticide paradoxes, it is very significant to review all the test cases frequently. And the new and different tests are necessary to be written for the implementation of multiple parts of the application or the software, which helps us to find more bugs.

**Testing is context dependent**

Testing is a context-dependent principle states that we have multiple fields such as e-commerce websites, commercial websites, and so on are available in the market. There is a definite way to test the commercial site as well as the e-commerce websites because every application has its own needs, features, and functionality. To check this type of application, we will take the help of various kinds of testing, different technique, approaches, and multiple methods. Therefore, the testing depends on the context of the application.

**Absence of errors fallacy**

Once the application is completely tested and there are no bugs identified before the release, we can say that the application is 99% bug-free. But there is the chance when the application is tested beside the incorrect requirements, identified the flaws, and fixed them on a given period would not help as testing is done on the wrong specification, which does not apply to the client's requirements. The absence of error fallacy means identifying and fixing the bugs would not help if the application is impractical and not able to accomplish the client's requirements and needs.

## What is Error?

The Problem in code leads to errors, which means that a mistake can occur due to the developer's coding error as the developer misunderstood the requirement or the requirement was not defined correctly. The developers use the term error.

## What is Failure?

we can say that if an end-user detects an issue in the product, then that particular issue is called a failure.

**What is Defect?**

The Defect is the difference between the actual outcomes and expected outputs.

When the application is not working as per the requirement is known as defects. It is specified as the aberration from the actual and expected result of the application or software.

**What is a bug?**

If testers find any mismatch in the application/system in the testing phase then they call it a Bug.

## What are the common problems in the software development process?

* Bad requirements
* Unrealistic schedules
* Inadequate testing
* Adding new features
* Poor communication

**SDLC :- Software Development Life Cycle**

Any SDLC should result in a high quality system that meets or exceeds customer expectations, reaches completion within time and cost estimates, works effectively and efficiently and is inexpensive to maintain and cost effective to enhance.