**MANUAL TESTING**

**1.Why testing is required?**

Ans: **Testing** is required for an effective performance of software application or product. It's **important** to ensure that the application should not result into any failures because it can be very expensive in the future or in the later stages of the development.

**2) What types of application we test**

Ans: web applications,

       desktop/windows applications

       Mobile applications

       ETL jobs

      Back end/batch programs/windows services

**3) what is SDLC and different phases in SDLC?**

Ans: Software development life cycle (SDLC) is a process to develop the application

**Different phases like:**

**Requirement Analysis and planning :** Senior team members analyze the requirements/input given by customers/business users. They will check whether the requirement is feasible or not (can be done or not). They also identify the risks associated with project.

Note: this high level requirements will be written in BRD (Business Requirement document) by Business Analyst

**Define/Design** :- In the define stage Business Analyst define more details about requirements (which are in BRD) in the form of SRS (software requirement specification) or Use Case diagram.

As part of design,

Senior Developers write High Level Design Document (HLD)

Developers write Low Level Design Document (LLD)

Seniors Tester write Test Planning document

Implementation/Development: Developers write the code for the requirements

Testers write test cases as per SRS

Testing : Execute the test cases what we prepared in previous stage

Deployment : Release the tested code to production

Maintenance : Support team monitoring the system that is running in production

**4) what is waterfal in SDLC?**

Ans :- The **waterfall** model is a sequential (non-iterative) design process, used in software development processes, in which progress is seen as flowing steadily downwards (like a **waterfall**) through the phases of conception, initiation, analysis, design, construction, testing, production/implementation and maintenance.

(Product req.doc) (software architecture)

| |

REQUIREMENTS--->DESIGN---->IMPLEMENTATION🡪VERIFICATION🡪MAINTENANCE

|

(software)

**5).What is the process in agile model**

Ans:- **Agile** SDLC **model** is a combination of iterative and incremental **process models** with focus on **process**adaptability and customer satisfaction by rapid delivery of working software product. **Agile** Methods break the product into small incremental builds. These builds are provided in iterations.



**6).what is scrum methodology**

**Ans:- Scrum** is an agile way to manage a project, usually **software**development. Agile **software** development with **Scrum** is often perceived as a **methodology**; but rather than viewing **Scrum** as **methodology**, think of it as a framework for managing a **process**.

**7).what is daily standup meeting and what we discuss**

**Ans:-** A **daily stand-up meeting** is a short organizational **meeting** that is held each day. The **meeting**, generally limited to between five and fifteen minutes long, is sometimes referred to as a **stand-up**, a morning roll-call or a **daily** scrum.

**8).what is user story/feature/sprint back log items and tasks in user story**

**9).what is sprint planning and spring retro**

**Ans:- Sprint planning** is a collaborative effort involving a ScrumMaster, who facilitates the meeting, a Product Owner, who clarifies the details of the product backlog items and their respective acceptance criteria, and the Entire Agile Team, who define the work and effort necessary to meet their **sprint**commitment.

**spring retro:-**

**10).what is burndown chart and velocity**

**Ans:-** Its purpose is to enable that the project is on the track to deliver the expected solution within the desired schedule. Simple **Burndown Chart**. The rate of progress of a Scrum Team is called "**velocity**". It expresses the amount of e.g. story points completed per iteration.

**11).what is product backlog item and sprint backlog items**

**Ans:-** In Scrum, a **product backlog item** ("PBI", "**backlog item**", or "**item**") is a unit of work small enough to be completed by a team in one Sprint iteration. **Backlog items** are decomposed into one or more tasks. See also **backlog** effort estimation unit.

**sprint backlog items:-** The **sprint backlog** is a list of tasks identified by the Scrum team to be completed during the Scrum **sprint**. During the **sprint** planning meeting, the team selects some number of **product backlog items**, usually in the form of user stories, and identifies the tasks necessary to complete each user story.

**12).what is user acceptance criteria test cases**

**Ans:- User acceptance test** (**UAT**) **criteria** (in agile software development) are usually created by business customers and expressed in a business domain language. These are high-level **tests** to verify the completeness of a **user** story or stories 'played' during any sprint/iteration.

**13).what is v model?**

**Ans:-** The **V** - **model** is SDLC **model** where execution of processes happens in a sequential manner in **V**-shape. It is also known as Verification and Validation **model**. **V** - **Model** is an extension of the waterfall **model** and is based on association of a **testing** phase for each corresponding development stage.

**14).what is STLC?**

**Ans:- Software Testing Life Cycle** (**STLC**) is the testing process which is executed in systematic and planned manner. In **STLC** process, different activities are carried out to improve the quality of the product. Let's quickly see what all stages are involved in typical **Software Testing Life Cycle** (**STLC**).

**15).what is defect?**

**Ans:-** A Software Defect / Bug is a condition in a software product which does not meet a software requirement (as stated in the requirement specifications) or end-user expectations (which may not be specified but are reasonable). In other words, a defect is an error in coding or logic that causes a program to malfunction or to produce incorrect/unexpected results.

* A program that contains a large number of bugs is said to be *buggy*.
* Reports detailing bugs in software are known as*bug reports*. (See [Defect Report](http://softwaretestingfundamentals.com/defect-report/))
* Applications for tracking bugs are known as *bug tracking tools*.
* The process of finding the cause of bugs is known as *debugging*.
* The process of intentionally injecting bugs in a software program, to estimate test coverage by monitoring the detection of those bugs, is known as *bebugging*.

**16).How to arise a defect and what we specify while logging defect?**

**Ans:-** Click on the drop down icon and select a User. You may want to link to an associated **Defect**. Click on the drop down icon on the Associated **Defect** field and select a **defect** to associate with. Switch to 'OS / Browser' tab to select the Web Browser(s) / Operating System(s) against which this **defect** is being **raised**.

**we specify while logging defect:-**

**17).How can you defect lifecycle**

**Ans:-** Defect life cycle is a cycle which a defect goes through during its lifetime. It starts when defect is found and ends when a defect is closed, after ensuring it’s not reproduced. [**Defect life cycle**](http://istqbexamcertification.com/what-is-a-defect-life-cycle/) is related to the bug found during testing

**Different types of testing:-**

**18).What is unit testing?**

**Ans:- Unit testing** is a software development process in which the smallest testable parts of an application, called units, are individually and independently scrutinized for proper operation. **Unit testing** is often automated but it can also be done manually.

**19). when do we use regression testing?**

**Ans:-** **Regression testing:-**

The purpose of regression testing is to confirm that a  recent program or code change has not adversely affected existing features.

Regression testing is nothing but full or partial selection of already executed test cases which are re-executed to ensure existing functionalities work fine.

This testing is done to make sure that new code changes should not have side effects on the existing functionalities. It  ensures that old code still works once  the new code changes are done.

**when do we use regression testing:-**

**20).What is integration testing?**

**Ans: Integration testing** (sometimes called **integration** and **testing**, abbreviated I&T) is the phase in software **testing** in which individual software modules are combined and **tested** as a group. It occurs after unit **testing** and before validation **testing**.

**when do we use integration testing:-**

**21).when do we use smoke testing and sanity testing?**

**Ans:- Smoke Testing**, also known as “Build Verification **Testing**”, is a type of software **testing** that comprises of a non-exhaustive set of **tests** that aim at ensuring that the most important functions work. The results of this **testing** is used to decide if a build is stable enough to proceed with further **testing**.

**sanity testing:- Sanity Testing** is the subset of Regression **Testing** and it is performed when we do not have enough time for doing **testing**. **Sanity testing** is the surface level **testing**where QA engineer verifies that all the menus, functions, commands available in the product and project are working fine.

**22).what is unit testing?**

**Ans:- Unit testing** is a software development process in which the smallest testable parts of an application, called units, are individually and independently scrutinized for proper operation. **Unit testing** is often automated but it can also be done manually.

**23).what is UAT?**

**Ans:-** In software development, **user acceptance testing** (**UAT**) - also called beta testing, application testing, and end user testing - is a phase of software development in which the software is tested in the "real world" by the intended audience.

**24).what is alpha and beta testing?**

**Ans:- Alpha testing** is simulated or actual operational **testing** by potential users/customers or an independent **test** team at the developers' site. **Alpha testing**is often employed for off-the-shelf software as a form of internal acceptance **testing**, before the software goes to **beta testing**.

**25).when do we use white box testing and block box testing?**

**Ans:- White**-**box testing** (also known as clear **box testing**, glass **box testing**, transparent **box testing**, and structural **testing**) is a method of **testing** software that **tests**internal structures or workings of an application, as opposed to its functionality (i.e. black-**box testing**).

**What is the purpose of white box testing:- White box testing** is a **testing** technique, that examines the program structure and derives **test** data from the program logic/code. The other names of glass **box testing**are clear **box testing**, open **box testing**, logic driven **testing** or path driven **testing**or structural **testing**.

**What is the purpose of blackbox testing?**

**block box testing:-** Black-box testing checks that the user interface and user inputs and outputs all work correctly. Part of this is that error handling must work correctly. It's used in functional and system testing.

**26).what we will do if we don’t have a time to test all stories?**

**27).what we will do if come across any severity issue before release day?**

**28).when do we use automation testing?**

**Ans:-** New Function – test case – manual test – works / passes – release – create automated script for regression • Evaluating functions for automation is the responsibility of the entire project team and needs to happen during the entire SDLC. • Includes automated unit tests, nightly builds and scripts with or without Functional Testing for Web Applications (formerly eTester) that can validate build files, DB, configurations and GUI

**29).what tester will do in each phase of SDLC?**

**30).difference between load and performance testing?**

**Ans:- *Performance Testing = how fast is the system?Load Testing = how much volume can the system process?***

Performance testing seems to me to be much more broad than load testing. Consider:

* A web developer can test the speed at which a page renders in a browser, and that is testing performance. Yet, that test would have nothing to do with load.
* I might analyze the efficiency at which my database processes a single specific SQL query, and the resulting speed of delivery of the records can be the slowest component of the whole page building process. Measuring that speed is about performance, but only one transaction is involved (small load).
* Load testing is usually focused on metrics like requests per second and concurrent users (the cause); whereas performance testing is more concerned with response times (the effect).

**31).different types of non-functional testing types?**

## Ans:- Types of Non Functional Testing

1. Performance Testing

2. Load Testing

3. Stress Testing

4. Volume Testing

5. Failover Testing

6. Security Testing

7. Compatibility Testing

8. Usability Testing

9. Scalability Testing

**Let's understand all the types of non functional testing in detail:**

**Performance Testing :**

First and foremost type of non functional testing is performance testing. In order to ensure that the response time of a system is acceptable, performance testing is carried out. By setting up a considerable load and a production-sized database, the system is tested for response times of several business critical processes.

**Load Testing :**

Types of non functional testing in [software testing](http://www.testing-whiz.com/) also includes load testing. To check whether the system can sustain the pressure or load of many users accessing the system at one time, load testing needs to be carried out. The production load is replicated in the test environment in this case after which the application/system is tested.

**Stress Testing :**

This testing is done to pull the system far beyond its capabilities and see how it reacts. Contrary to load testing in which the maximum allowable load is generated, in stress testing, the load generated is more than what the system is expected to handle.

**Volume Testing :**

When storage requirements and capabilities of the system are to be tested, volume testing is done. When a huge database size is encountered, system’s performance and its ability to exchange data and information are tested in this case.

**Failover Testing :**

To test how well the redundancy mechanism works when the system encounters heavy load or unexpected failure is what failover testing is about. Also, when the specific failed system is back again, it must begin to function as per requirements – this is fail-back testing.

**Security Testing :**

While performing non functional testing, to test how well the system can preserve itself and the data it holds in situation of malicious attacks is called security testing. Confidentiality, integrity, availability, authentication and authorization are the main areas that are tested when security testing is considered. Also, network security, system security and application security are other areas that will be tested in this case.

**Compatibility Testing :**

An application is tested for its coordination with different hardware and software that it is expected to work on. Another testing that can be done is working of the application with different versions or releases of the same hardware or software.

**Usability Testing :**

To verify the ease of usage of an interface within an application is what usability testing is about. Learnability and memorability of the application are main factors in this case. This testing is particularly important when [testing GUI](http://www.testing-whiz.com/web-ui-comparison-and-functional-test-automation).

**Scalability Testing :**

When an application is tested for it ability to increase and scale up on any of its non-functionality requirements such as load, number of transactions, number of servers, volume of data etc., it is known as scalability testing.

The above list is not exhaustive since there are more than 150 testing types as of today. The different types of testing, including [automated testing](http://www.testing-whiz.com/), that needs to be done depending upon the scope of the project and maturity of the application under test.

- See more at: http://www.testing-whiz.com/blog/types-of-non-functional-software-tests#sthash.n2UwWuWF.dpuf

**32).what is test case?**

**Ans:-** A test case is a document, which has a set of test data, preconditions, expected results and postconditions, developed for a particular test scenario in order to verify compliance against a specific requirement.

**33).what is test plan/test strategy document**

**Ans: Test plan document contains different section like**

**Types of testing :**

**Exit and Entry criteria :**

A **test plan** is a document detailing the objectives, target market, internal beta team, and processes for a specific beta **test** for a software or hardware product. The **plan**typically contains a detailed understanding of the eventual workflow. Software **Testing** portal.

A **Test** Plan **Documents** the **strategy** that will be used to verify and ensure that a product or system meets its design specifications and other requirements. ... The **Test Strategy document** describes the scope, approach, resources and schedule for the **testing** activities of the project.

**34).what is TDD and BDD (cucumber framework)**

**Ans:- Cucumber** is a testing **framework** which supports Behavior Driven Development (**BDD**). ... **Cucumber** itself is written in Ruby, but it can be used to “test” code written in Ruby or other languages including but not limited to Java, C# and Python.

**35).what is priority and severity in defect?**

**Ans:- Priority** of a **defect** is related to how quickly a **bug** should be fixed and deployed to live servers. When a **defect** is of high severity, most likely it will also have a high **priority.**

**severity in defect** :-In software testing, **defect severity** can be defined as the degree of impact a **defect** has on the development or operation of a component application being tested. Higher effect on the system functionality will lead to the assignment of higher **severity** to the bug

**36).how to estimate test cases?**

1. **Ans:-** 3-Point Software Testing Estimation Technique.
2. Use – Case Point Method:
3. Work Breakdown Structure.
4. Wideband Delphi technique.
5. Function Point/Testing Point Analysis.
6. Percentage of development effort method.
7. Percentage distribution.
8. Best Guess.

**37).what is most challenge defect u came across?**

**38).how to deal the production defects?**

**Ans:** normally end user will report this issue.

       we need to talk to them (end users) and reproduce the issue with in staging environment

  Create defect in defect tool under the production release version

  developers will fix the issue

  we (QA) test the issue on production version code (stageing) and release the fix to proudction after we verify

 we have to create a defect on current sprint/release so that developer will add this code to the current sprint/release

**39).test design techniques:**

**Ans:- Test Design** is creating a set of inputs for given software that will provide a set of expected outputs. The idea is to ensure that the system is working good enough and it can be released with as few problems as possible for the average user. Broadly speaking there are two main categories of **Test Design Techniques**.

**40).if we dont have time to test call test cases what we will do**

**Ans:-** Go to manager - it is his duty to resolve thit problem - providing you are just poor tester.

Faking test results can backfire badly. Or go unnoticed

**41).how we learn the functionality of system?**

**Ans:-** A function is described as a set of inputs, the behavior, and outputs. Functional requirements may be calculations, technical details, data manipulation and processing and other specific **functionality** that define what a **system** is supposed to accomplish

**42).what are the tools to manage defects/stories?**

**43).who will assign the work?**

**44).types of test metrics we use normally**

In software testing, **Metric** is a quantitative measure of the degree to which a **system, system component, or process**possesses a given attribute

**45).what is traceability matrix?**

**Ans:-** A **traceability matrix** is a document, usually in the form of a table, used to assist in determining the completeness of a relationship by correlating any two baselined documents using a many-to-many relationship comparison

**46).what are typical environments we have in projects**

**47).what are different defect metrics and measurements we prepare**

**48).What is staging environment**

**Ans:-** A stage or **staging environment** is an **environment** for testing that exactly resembles the production **environment**. In other words, it's a complete but independent copy of the production **environment**, including the database. **Staging**provides a true basis for QA testing because it precisely reproduces what is in production.

**49).what is development environment**

**Ans:-** In computer program and software product **development**, the **development environment** is the set of processes and programming tools used to create the program or software product. The term may sometimes also imply the physical **environment**.

**50).what is QA environment**

**Ans:-** A **QA environment** is where you test your upgrade procedure against data, hardware, and software that closely simulate the Production **environment** and where you allow intended users to test the resulting Waveset application. A Production **environment** is where the Waveset application is actually available for business use.

**51).what is production environment**

**Ans:-** A **production environment** is where the real-time staging of programs that run an organization are executed, and includes the personnel, processes, data, hardware, and software needed to perform day-to-day operations.