**Manual Testing**

1. **Defect life cycle?**

Defect life cycle, also known as Bug Life cycle is the journey of a defect cycle, which a defect goes through during its lifetime. It varies from organization to organization and from project to project as it is governed by the software testing process and depends upon the tools used.

1. **difference between load and performance testing?**

**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).
* different types of non-functional testing types?
* Load/Performance testing.
* Compatibility testing.
* Localization testing.
* Security testing.
* Reliability testing.
* Stress testing.
* Usability testing.
* Compliance testing.

1. **how do we write test cases in BDD format?**

BDD is a process designed to aid the management and the delivery of software development projects by improving communication between engineers and business professionals. In so doing, BDD ensures all development projects remain focused on delivering what the business needs while meeting all requirements of the user.

1. **how to arise a defect and what we specify while logging defect?**

Because of the following reasons the software [defects](http://istqbexamcertification.com/what-is-defect-or-bugs-or-faults-in-software-testing/)arise:

– The person using the software application or product may not have enough knowledge of the product.

– Maybe the software is used in the wrong way which leads to the defects or [failures](http://istqbexamcertification.com/what-is-a-failure-in-software-testing/).

– The developers may have coded incorrectly and there can be defects present in the design.

– Incorrect setup of the testing environments.

5**) how to deal the production defects?**

The best thing you can do is learn from it and prevent the same in the future. I write an automated test-case for each defect found in production, since these are the brittle parts of the application.

6) **how to estimate test cases?**

1. 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.

7) **if we don’t have time to test call test cases what we will do?**

 First, we have the test cases (or test scripts) that are written based on the requirement document. This pretty much covers what functionalities to test. Therefore, looking at the test cases tells us what to test in the application.

8) **what are different defect metrics and measurements we prepare in testing?**

Software Metrics are used to measure the quality of the project. Simply, Metric is a unit used for describing an attribute. Metric is a scale for measurement.

Types of Manual Test Metrics:

Testing Metrics are mainly divided into 2 categories.

1. Base Metrics
2. Calculated Metrics

**Base Metrics:**Base Metrics are the Metrics which are derived from the data gathered by the Test Analyst during the test case development and execution.

This data will be tracked throughout the Test Life cycle. I.e. collecting the data like, Total no. of test cases developed for a project (or) no. of test cases need to be executed (or) no. of test cases passed/failed/blocked etc.

**Calculated Metrics:**  
Calculated Metrics are derived from the data gathered in Base Metrics. These Metrics are generally tracked by the test lead/manager for Test Reporting purpose.

1. ) **what are test design techniques?**

By design we mean to create a plan for how to implement an idea and technique is a method or way for performing a task. So, 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. They are:

1. [Static Techniques](http://istqbexamcertification.com/what-is-static-testing-technique/)
2. [Dynamic Techniques](http://istqbexamcertification.com/what-is-dynamic-testing-technique/)
3. **what are the tools to manage defects/stories?**

 Most Popular Bug Tracking Software. Here goes: ...

Stryka: Details: Stryka is a cutting-edge enterprise test management tool, built from the ground up using the latest web and mobile technologies. ...

Bugzilla: ...

Lean Testing. ...

JIRA: ...

Mantis: ...

Trac: ...

Redmine:

1. **what are typical environments we have in projects?**

Development

QA == Functional testing of the system

System Integration Testing == Tests the system from end to end

User Acceptance Testing = Allows the user to validate the functionality over time

Production == Production

Production Parallel == A parallel of production to replicate production issues

CCE = Client Certification Environment

1. **what is agile method?**

AGILE. Agile software development refers to a group of software development methodologies based on iterative development, where requirements and solutions evolve through collaboration between self-organizing cross-functional teams.

1. **what is alpha and beta testing?**

**Alpha testing:** In this type of testing, the users are invited at the development center where they use the application and the developers note every particular input or action carried out by the user. Any type of abnormal behavior of the system is noted and rectified by the developers.

**Beta testing:** In this type of testing, the software is distributed as a beta version to the users and users test the application at their sites. As the users explore the software, in case if any exception/defect occurs that is reported to the developers.

1. **what is burndown chart and velocity?**

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.

1. **what is daily standup meeting and what we discuss?**

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.

1. **what is defect?**

A programmer while designing and building the software can make mistakes or error. These mistakes or errors mean that there are flaws in the software. These are called defects. When actual result deviates from the expected result while testing software application or product then it results into a defect.

1. **what is development environment?**

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.

1. **what is Exit and Entry criteria?**

Entry criterion is used to determine when a given test activity should start. It also includes the beginning of a level of testing, when test design or when test execution is ready to start.Every company has entry and exit criteria. When we test applications, we refer to exit criteria. When we are about to finish testing, then the QA Team (QA Manager) refers to the exit criteria (exit criteria tells the level of defect that you can be comfortable with before it goes to production. For example, there should be ZERO critical defect, ZERO high level defect, ZERO medium defect, 1 Low level defect, all the test cases must be 100% executed etc). Once the exit criteria meet the requirements, then the software is sufficiently tested.

1. **What is integration testing?**

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.

1. **what is most challenge defect u came across?**

Challenging in terms of finding them are couple of them which are bound to be missed out or difficult to figure out while testing manually.

* Load and Performance testing of an app or website or just a small feature within that app. A very common scenario where a tester might find a feature working fine while testing but see it responding slowly or crashing when released in production. Reason being that feature is not able to handle to much of users at the same time or is just not ready for public environment. To overcome this companies go for using tools like Jmeter, Silk Performer etc. Obviously, this depends on what you are testing. Standalone apps which does not require any interaction with server won’t need this type of testing at all
* Regression testing. The biggest challenge for a manual tester is to find bugs in features which are already stable. Question is if the feature is already stable then why will it have any bugs? Yes, it might. New features often tend to break older features and as a manual tester things which are already working fine since ages are neglected or missed. Most obvious solution to this is automate your regression testing to avoid missing out such bugs. Again, this depends on your product and company budget/resources.

1. **what is priority and severity in defect?**

The priority status is set based on the customer requirements. While Severity is the extent to which the defect can affect the software. In other words, it defines the impact that a given defect has on the system. It is totally related to the quality standard or devotion to standard.

1. **what is product back log items?**

Product Backlog Item. 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.

1. **what is production environment?**

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.

1. **what is QA environment?**

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 Wave set application. A Production environments where the Wave set application is available for business use.

1. **what is requirement traceability matrix?**

The Requirements Traceability Matrix (RTM) is a document that links requirements throughout the validation process. The purpose of the Requirements Traceability Matrix is to ensure that all requirements defined for a system are tested in the test protocols.

1. **what is scrum methodology?**

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.

1. **what is SDLC and different phases in SDLC?**

 The systems (or software) development life cycle (SDLC) is a conceptual model used in project management that describes the stages involved in an information system development project, from an initial feasibility study through maintenance of the completed application.

It includes the following different stages:

1.  Requirement phase

2.  Design phase

3.  Coding (programming)

4.  Testing

5.  Release (Production)

6.  Maintenance (Support)

30) **what is sprint grooming?**

Product backlog refinement—sometimes called product backlog grooming in reference to keeping the backlog clean and orderly—is a meeting that is held near the end of one sprint to ensure the backlog is ready for the next sprint

30) **what is sprint planning meeting?**

Sprint planning is a collaborative effort involving a Scrum Master, 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.

31) **what is sprint retrospective?**

The sprint retrospective is a meeting facilitated by the Scrum Master at which the team discusses the just-concluded sprint and determines what could be changed that might make the next sprint more productive.

32) **what is sprint review meeting?**

In Scrum, each sprint is required to deliver a potentially shippable product increment. This means that at the end of each sprint, the team has produced a coded, tested and usable piece of software. So, at the end of each sprint, a sprint review meetings held.

33) **What is staging environment?**

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.

34) **what is STLC?**

Software Testing Life Cycle (STLC) is defined as a sequence of activities conducted to perform Software Testing. It consists of series of activities carried out methodologically to help certify your software product. Diagram - Different stages in Software Test Life Cycle.

35) **what is TDD and BDD (cucumber framework)?**

Behavior-driven Development (BDD) is an agile software development practice that enhances the paradigm of Test Driven Development (TDD) and acceptance tests, and encourages the collaboration between developers, quality assurance, domain experts, and stakeholders.

36) **what is test case?**

A test case is a document, which has a set of test data, preconditions, expected results and postconditions, developed for a test scenario to verify compliance against a specific requirement.

37) **what is test planning/test strategy document?**

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.

38) **what is the process in agile model?**

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.

39) **what is UAT?**

User acceptance testing (UAT) is the last phase of the software testing process. During UAT, actual software users test the software to make sure it can handle required tasks in real-world scenarios, according to specifications.

40) **what is unit testing?**

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 can be done manually but is often automated.

41) **what is unit testing?**

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 can be done manually but is often automated.

42) **what is user acceptance criteria test cases?**

Firstly, the criteria by which the software is “working” needs to be assembled. These are likely to be collated from the system requirements, and user stories. Next, a set of UAT test cases must be created. Centric defines a UAT test case as: ... Each case covers a specific usage scenario of the software.

44) **what is v model?**

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.

45) **what is waterfall method?**

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

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

**Inception Phase**

In this phase, a test engineer will get an opportunity to identify the necessities of project. Normally the data are recorded by the architecture team in the architectural reference document. Data design, information design, system design are the main issues in this phase.

**Elaboration Phase**

In this phase, a test engineer will get an opportunity to identify how the project is planned. This is a major phase, where the entire design of the project is documented in the JAD phase in the System requirement document, business requirement document, product requirement document commercial use cases. Planner, Commercial reviewer, project organization, execution, testing, maintenance of project teams etc are attended the JAD phase to give sign-off on these completed documents.

**Construction Phase**

In this phase, programmers play an important role of building the application depends on the plan acknowledged during the JAD stage. Here tester group must follow the programming group to identify several adjustments taken by the system. There may be any kind of fault which are overlooked by programmer, misapprehend the planed records, in that time, a tester can always rise the issue to the regarding programmer to solve the issue. A testing group requires developing the high-level scenarios (HLS) on basis of the elaboration phase. High level scenarios may have more than one test case. A tester should ensure that all the necessities are discovered to a test case by a quality affirmation standard. It is mandatory to record test cases based on all probable references of the newest modernized data and signed-off.

**Transition Phase**

In this phase if any fault or errors are originating then these are test again and it goes though the regression testing. With the help of regression testing, consistent systems develop. By the helping of these testing methods, any fundamental result can be converted into a tough and consistent system.

47) **What types of application we test?**

**Exploratory Testing:** Black box testing technique performed without planning and documentation. It is usually performed by manual testers.

**Keyword-driven Testing:** Also, known as table-driven testing or action-word testing, is a software testing methodology for automated testing that separates the test creation process into two distinct stages:  a Planning Stage and an Implementation Stage. It can be used by either manual or automation testing teams.

**Manual Scripted Testing:** Testing method in which the test cases are designed and reviewed by the team before executing it. It is done by manual testing teams.

**Manual-Support Testing:** Testing technique that involves testing of all the functions performed by the people while preparing the data and using these data from automated system. it is conducted by testing teams.

**Non-functional Testing:** Testing technique which focuses on testing of a software application for its non-functional requirements. Can be conducted by the performance engineers or by manual testing teams.

**Negative Testing:** Also, known as "test to fail" - testing method where the tests' aim is showing that a component or system does not work. It is performed by manual or automation testers.

50) **when do we use automation testing?**

Test engineers strive to catch them before the product is released but they always creep in and they often reappear, even with the best manual testing processes. Test Automation software is the best way to increase the effectiveness, efficiency and coverage of your software testing.

51) **when do we use integration testing?**

Integration testing, also known as integration and testing (I&T), is a [software](http://searchsoa.techtarget.com/definition/software) development process which program units are combined and tested as groups in multiple ways. In this context, a unit is defined as the smallest testable part of an [application](http://searchsoftwarequality.techtarget.com/definition/application). Integration testing can expose problems with the [interface](http://searchcio-midmarket.techtarget.com/definition/interface)s among program components before trouble occurs in real-world program execution. Integration testing is a component of [Extreme Programming](http://searchsoftwarequality.techtarget.com/definition/Extreme-Programming) (XP), a pragmatic method of software development that takes a meticulous approach to building a product by means of continual testing and revision.

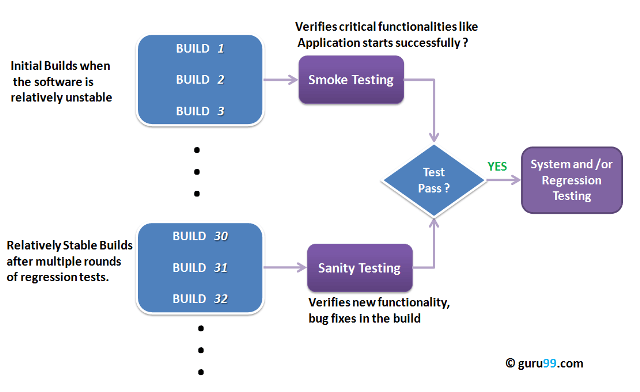
52) **when do we use regression testing?**

Regression testing is the process of testing changes to computer programs to make sure that the older programming still works with the new changes. Regression testing is a normal part of the program development process and, in larger companies, is done by code testing specialists.

53**) when do we use smoke testing and sanity testing?**

Smoke and Sanity testing are the most misunderstood topics in Software Testing. There is enormous amount of literature on the subject, but most of them are confusing. The following article tries to address the confusion.

The key differences between Smoke and Sanity Testing can be learned with the help of following diagram -



**What is a Software Build?**

If you are developing a simple computer program which consists of only one source code file, you merely need to compile and link this one file, to produce an executable file. This process is very simple.  
Usually this is not the case. A typical Software Project consists of hundreds or even thousands of source code files. Creating an executable program from these source files is a complicated and time-consuming task.  
You need to use "build" software to create an executable program and the process is called " Software Build"

**What is Smoke Testing?**

Smoke Testing is a kind of Software Testing performed after software build to ascertain that the critical functionalities of the program is working fine. It is executed "before" any detailed functional or regression tests are executed on the software build. The purpose is to reject a badly broken application, so that the QA team does not waste time installing and testing the software application.

In Smoke Testing, the test cases chosen cover the most important functionality or component of the system. The objective is not to perform exhaustive testing, but to verify that the critical functionalities of the system is working fine.  
For Example a typical smoke test would be - Verify that the application launches successfully, check that the GUI is responsive ... etc.

**What is Sanity Testing?**

Sanity testing is a kind of Software Testing performed after receiving a software build, with minor changes in code, or functionality, to ascertain that the bugs have been fixed and no further issues are introduced due to these changes. The goal is to determine that the proposed functionality works roughly as expected. If sanity test fails, the build is rejected to save the time and costs involved in a more rigorous testing.

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

**Black-box testing:** Black-box testing (also known as functional testing) treats software under test as a black-box without knowing its internals. Tests are using software interfaces and trying to ensure that they work as expected. If functionality of interfaces remains unchanged, tests should pass even if internals are changed. Tester is aware of what the program should do but does not have the knowledge of how it does it. Black-box testing is most commonly used type of testing in traditional organizations that have testers as a separate department, especially when they are not proficient in coding and have difficulties to understand the code. It provides **external perspective** of the software under test.

**White-box testing**

White-box testing (also known as clear box testing, glass box testing, transparent box testing, and structural testing) looks inside the software that is being tested and uses that knowledge as part of the testing process. If, for example, exception is thrown under certain conditions, test might want to reproduce those conditions. White-box testing requires internal knowledge of the system and programming skills. It provides **internal perspective** of the software under test.

56) **Why testing is required?**

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. It's required to stay in the business.