**What is difference between QA, QC and Software Testing?**

**Quality Assurance (QA): QA refers to the planned and systematic way of monitoring the quality of process which is followed to produce a quality product. QA tracks the outcomes and adjusts the process to meet the expectation.**

**Quality Control (QC): Concern with the quality of the product. QC finds the defects and suggests improvements. The process set by QA is implemented by QC. The QC is the responsibility of the tester.**

**Software Testing: is the process of ensuring that product which is developed by the developer meets the user requirement. The motive to perform testing is to find the bugs and make sure that they get fixed.**

**When to start QA in a project?**

**A good time to start the QA is from the beginning of the project startup. This will lead to plan the process which will make sure that product coming out meets the customer quality expectation. QA also plays a major role in the communication between teams. It gives time to step up the testing environment. The testing phase starts after the test plans are written, reviewed and approved.**

**What is destructive testing, and what are its benefits?**

**Destructive testing includes methods where material is broken down to evaluate the mechanical properties, such as strength, toughness and hardness.**

**Benefits of Destructive Testing (DT)**

**- Verifies properties of a material**

**- Determines quality of welds**

**- Helps you to reduce failures, accidents and costs**

**- Ensures compliance with regulations**

**Difference between Verification and Validation:**

**- Verification is Static testing whereas Validations is Dynamic Testing.**

**- Verification takes place before validation.**

**- Verification evaluates plans, documents, requirements and specifications, where as Validation evaluates product.**

**- Verification inputs are checklist, issues list, walkthroughs and inspection, where as in Validation testing of actual product.**

**- Verification output is set of documents, plans, specifications and requirement documents where as in Validation actual product is output.**

**What is the difference between**[**Load and Stress testing**](http://www.softwaretestinghelp.com/what-is-performance-testing-load-testing-stress-testing/)**?**

Stress Testing is a technique which validates the behavior of the system when it executes under stress. To explain, we reduce the resources and check the behavior of the system.

We first understand the upper limit of the system and gradually reduce the resources and check the system behavior.

In Load testing we validate the system behavior under the expected load. The load can be of concurrent user or resources accessing the system at the same time.

Below are the phases of STLC:

1. Requirements phase
2. Planning Phase
3. Analysis phase
4. Design Phase
5. Implementation Phase
6. Execution Phase
7. Conclusion Phase
8. Closure Phase

**What is the difference between build and release?**

Build: It is a number given to Installable software that is given to the testing team by the development team.

Release: It is a number given to Installable software that is handed over to the customer by the tester or developer

**What is branch testing, what is boundary testing and what is volume testing?**

The testing of all the branches of the code, which is tested once, is known as branch testing. While the testing, which is focused on the limit conditions of the software is known as boundary testing. Volume Testing: The process of checking the system, whether the system can handle the required amounts of data, user requests, etc. is known as Volume Testing.

**What is Agile testing and what is the importance of Agile testing?**

Agile testing is software testing, is testing using Agile Methodology. The importance of this testing is that, unlike normal testing process, this testing does not wait for the development team to complete the coding first and then doing testing. The coding and testing both goes simultaneously. It requires continuous customer interaction.

**Explain what are test driver and test stub and why it is required?**

* The stub is called from the software component to be tested. It is used in top down approach
* The driver calls a component to be tested. It is used in bottom up approach
* It is required when we need to test the interface between modules X and Y and we have developed only module X. So we cannot just test module X but if there is any dummy module we can use that dummy module to test module X.

**Explain what is the difference between Regression testing and Retesting?**

* Retesting is carried out to check the defects fixes, while regression testing is performed to check whether the defect fix have any impact on other functionality.

A mistake in coding is called error, error found by tester is called defect, defect accepted by development team then it is called bug, build does not meet the requirements then it Is failure.

**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.

**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.

**Performance Testing**: Performance Testing is performed to determine response time of the some components of the system perform under a particular workload. It is generally measured in terms of response time for the user activity. It is designed to test the overall performance of the system at high load and stress condition

**What is difference between test effectiveness and test efficiency?**

- Test Efficiency is the ratio of number of test cases executed by unit of time (generally per hour).  
  
- It is the amount of code and testing resources required by a program to perform a particular function. Test Effectiveness evaluates the effect of the test environment on the application.  
  
- Test Effectiveness is a measure by the customer response on meeting product requirements whereas Test Efficiency is a measure of optimum utilization of resources to create the Software product.

**What is Exploratory Testing and Specification-Driven testing?**

Exploratory testing: means testing an application without a test plan and test script. In exploring testing test explore the application on the basis on his knowledge. The tester has no knowledge about the application previously. He explores the application like an end user and try to use it. While using the application his main motive is to find the bugs which are in the application.

Specification-driven testing: means to test the functionality of software according to the user requirements. In this, tester inputs multiple data and monitors the outputs from, the test object. In this testing tester evaluate the showstopper bugs which break the major functionality of the application. This type of testing requires test plan and test.

**Explain CMM.**

Capability Maturity Model (CMM) is divided in five levels:

1. Initial: The organization is characterized by an adhoc set of activities. The processes aren't defined and success depends on individual effort and heroics.

2. Repeatable: In this level some processes are repeatable, possibly with consistent results.

3. Defined: In this level, we define all processes are documented for both management and engineering activities, and standards.

4. Managed: Detailed measures of each process are defined and product quality data is routinely collected. Both process and products are quantitatively understood and controlled.

5. Optimizing: In this we optimize the application by following improvement process.

**What is Negative Testing?**

Negative Testing is performed to find the situation when the software crashed. It is a negative approach, in this tester try to put efforts to find the negative aspects of the application. Negative testing ensures that application can handle the invalid input, incorrect data and incorrect user response. For example, when user enters the alphabetical data in a numeric field, then error message should be display saying “Incorrect data type, please enter a number”.

**What is Path Testing?**

Path testing is a testing in which tester ensure that every path of the application should be executed at least once. In this testing, all paths in the program source code are tested at least once. Tester can use the control flow graph to perform this type of testing.

**Why we need Localization Testing?**

Localization testing mainly deals with the functionality of application and GUI of the application. The purposes of using Localization testing are following:

- Mainly deal with internationalization and localization aspects of software.

- Evaluate how successfully the language is interpreted into a specific language.

- Translate GUI of application so that it can adapt to a particular region language and interface.

**What is Endurance Testing?**

Endurance testing: in this testing we test application behavior against the load and stress applies over application for a long duration of time. The goal of this testing are:

- To determine the how the application is going to responds for high load and stress conditions in the real scenario.

- To ensure that the response times in highly load and stress conditions are within the user’s requirement of response time.

- Checks for memory leaks or other problems that may occur with prolonged execution.

**What is Gorilla Testing?**

A test technique that involves testing with various ranges of valid and invalid inputs a particular module or component functionality extensively. In Gorilla testing test case and test data are not required. It uses random data and test cases to perform testing of application. The purpose of Gorilla testing is to examine the capability of single module functionality by applying heavy load and stress to it. And determine how much load and stress it can tolerate without getting crashed.

**What are different types of verifications?**

Verification is static type of software testing which is started in earlier phase of development of software. In this approach we don’t execute the software that the reason it comes in static testing. The product is evaluated by going through the code. Types of verification are:

- Walkthrough: Walkthroughs are informal technique. Where the Developer leader organizing a meeting with team member to take feedback regarding the software. This can be used for the improvement of the software quality. Walkthrough are unplanned in the SDLC cycle.

- Inspection: Inspection is a done by checking of a software product thoroughly with the intention to find out defect and ensuring that software is meeting the user requirements.

**What is severity and priority of bug? Give some example.**

Priority: concern with application from the business point of view.

It answers: How quickly we need to fix the bug? Or how soon the bug should get fixed?

Severity: concern with functionality of application. It deals with the impact of the bug on the application.

**What is Testware?**

The subset of software which helps in performing the testing of application. Testware are required to plan, design, and execute tests. It contains documents, scripts, inputs, expected results, set-up and additional software or utilities used in testing. Testware is term given to combination of all utilities and application software that required for testing a software package.

Testware is special because it has:

1. Different purpose

2. Different metrics for quality and

3. Different users

**What is the basis for choosing the SDLC model for development of software?**

The choice of SDLC depends on the various factors and how stable are the requirements:

- When the requirements are very clearly known, documented and not subject to change then we can follow the waterfall model.

- Most of the companies follow the V mode for the development because this model includes both verification and validation activities and testing is involved in earlier phase.

- Iterative model can be used to build application where requirement changes after a period of times or application features or added on with smaller release. When the client is ready for the delivery of the product in parts or phases.

**Explain bug leakage and bug release.**

Bug Leakage: When customer or end user discovered a bug which can be detected by the testing team. Or when a bug is detected which can be detected in pervious build then this is called as Bug Leakage.

Bug release: is when a build is handed to testing team with knowing that defect is present in the release. The priority and severity of bug is low. It is done when customer want the application on the time. Customer can tolerate the bug in the released then the delay in getting the application and the cost involved in removing that bug. These bugs are mentioned in the Release Notes handed to client for the future improvement chances.

**What is Monkey testing?**

Monkey testing is a type of Black Box Testing used mostly at the Unit Level. In this tester enter the data in any format and check the software is not crashing. In this testing we use Smart monkey and Dumb monkey.

- Smart monkeys are used for load and stress testing, they will help in finding the bugs. They are very expensive to develop.

- Dumb monkey, are important for basic testing. They help in finding those bugs which are having high severity. Dumb monkey are less expensive as compare to Smart monkeys.

**What methodologies do you used to develop test cases?**

For developing the test cases we use following strategies:

- Error Guessing: The tester has to guess what fault might occur and to design the tests to represent them.

- Equivalence Class Partitioning: The input domain data is divided into different equivalence data classes; take few valid values with 2 invalid values. This is used to reduce the total number of test cases to a finite set of testable test cases.

- Boundary value analysis: Boundary value analysis testing technique is used to identify errors at boundaries rather than finding those exist in center of input domain. Boundary value analysis is a next part of Equivalence.

**What is Traceability Matrix?**

* A test matrix is used to map test scripts to requirements.

Traceability Matrix is a method used to validate the compliance of product with requirements for that product. The requirement is written in a row of the matrix and the columns of the matrix. Now they are used to identify how and where each requirement has been addressed.

It is in the form of table that correlates two base lined documents that require a many-to-many relationship. It is used with high level requirement and detailed requirement of the software product to the matching parts of high level design, detailed design, test plan, and test cases. The relationship to the source documents is required for both backward traceability and forward traceability.

**What is random testing?**

When test inputs are selected randomly from the input domain of the system, this is Random Testing. Random testing involves following procedure:

- The input domain is selected.

- Test inputs are selected independently from the domain.

-The system under test is executed on these inputs. The inputs constitute a random test set.

- The results are compared to the system specification. The test is a failure if any input leads to incorrect results, otherwise it is a success.

**How do you perform integration testing?**

Integration testing is black box testing. Integration testing focuses on the interfaces between units, to make sure the units work together. For integration testing we ensure that all units testing of the each component is performed earlier. Integration testing begins only after the unit testing. The purpose of integration testing is to ensure different components of the application interact with each other. So that, components work as per the customer requirements. Test cases are developed with the purpose of exercising the interfaces between the components. Integration testing is considered complete, when actual results and expected results are same

**What is alpha and beta testing?**

Alpha testing: is performed by the IN-House developers. After alpha testing the software is handed over to software QA team, for additional testing in an environment that is similar to the client environment.

Beta testing: It is performed by end user. So that they can make sure that the product is bug free or working as per the requirement. IN-house developers and software QA team perform alpha testing. The public, a few select prospective customers or the general public performs beta testing.

**What is data driven testing?**

Data Driven is an automation testing part in which test input or output values, these values are read from data files. It is performed when the values are changing by the time. The different data files may include data pools, csv files, Excel files. The data is then loaded into variables in recorded or manually coded scripts. For data driven testing we use Parameterzing and Regular expression Technique.

**How much the bug is affecting the functionality of the application?**

**Ex.**

High Priority and Low Severity:

Company logo is not properly displayed on their website.

High Priority and High Severity:

Suppose you are doing online shopping and filled payment information, but after submitting the form, you get a message like "Order has been cancelled."

Low Priority and High Severity:

If we have a typical scenario in which the application get crashed, but that scenario exists rarely.

Low Priority and Low Severity:

There is a mistake like "You have registered success" instead of successfully, success is written.

**What are the common problems with software automation?**

Software problem are listed below:

1. Purchasing the license of tool (QTP, selenium, QC, LR)

2. Lack of skilled Tester to run the tool

3. Expectation that automated tests will find a lot of new defects

4. Maintenance of automated tests

5. Technical problems of tools

**Explain bug life cycle.**

**Bug Life Cycle:**

- When a tester finds a bug .The bug is assigned with NEW or OPEN status.

- The bug is assigned to development project manager who will analyze the bug .He will check whether it is a valid defect. If it is not valid bug is rejected, now status is REJECTED.

- If not, next the defect is checked whether it is in scope. When bug is not part of the current release .Such defects are POSTPONED

- Now, Tester checks whether similar defect was raised earlier. If yes defect is assigned a status DUPLICATE

- When bug is assigned to developer. During this stage bug is assigned a status IN-PROGRESS

- Once code is fixed. Defect is assigned with FIXED status.

- Next the tester will re-test the code. In case the test case passes the defect is CLOSED

- If the test case fails again the bug is RE-OPENED and assigned to the developer. That’s all to Bug Life Cycle.

**Terms:** V-model, waterfall, System Test, User Acceptance (UAT), Functional Testing, Non-Functional Testing, White-BOX Testing, Security Testing, Factor Acceptance Test (FAT), Site Acceptance Test (SAT), Usability Testing, Compatibility Testing, and Black-BOX Testing

* Queries
* Indexing
* Normalization

Systems:

Types of testing. Difference between qa and qc Bug life cycle Sdlc Basic know how of agile Verification and validation Db queries (select update and inner join preferably)