**MANUAL TESTING**

1. what is waterfal in SDLC?

The Waterfall approach, the whole process of software development is divided into separate phases like

1.Requirement gathering and analysis

2.system design

3. implementation

4. Integration and testing

5. Deployment of system

6. Maintenance

2. what is the process in \*agile \*model?

Agile methodology deals with different phases like requirements, design, implementation, verification and maintenance.

3. what is scrum methodology?

Scrum is the subset of agile. Scrum was formalized for software development projects. A [Scrum process](https://www.cprime.com/2015/03/5-tips-to-manage-scrum-processes-in-the-real-world/) is distinguished from other agile processes by specific concepts and practices, divided into the three categories of Roles, Artifacts, and Time Boxes.

4. what is daily standup meeting and what we discuss?

Daily standup meeting is the short organizational meeting which will be held each day. The meeting will be for five to ten minutes. Sometimes called as stand-up or a daily scrum meeting.

5. what is user story/feature/sprint back log items and tasks in user story?

<https://www.mountaingoatsoftware.com/blog/the-difference-between-a-story-and-a-task>

6. what is sprint planning and spring retro?

Sprint planning is a meeting

<http://istqbexamcertification.com/what-is-sprint-planning-in-agile-project-management/>

Spring retro means feedback about that meeting.

7. What is burndown chart and velocity?

The burndown is a chart that shows how quickly you and your team are burning through your customer’s user stories. It shows the total effort against the amount of work we deliver each iteration.

<http://www.agilenutshell.com/burndown>

Velocity means the rate of progress of the scrum team.

8. What is product backlog item and sprint backlog items?

<http://mplaza.pm/difference-between-product-backlog-and-sprint-backlog/>

9. what is user acceptance criteria test cases?

User acceptance is a type of testing performed by the Client to certify the system with respect to the requirements that was agreed upon. This testing happens in the final phase of testing before moving the software application to Market or Production environment.

the criteria by which the software is considered to be “working” needs to be assembled. These are likely to be collated from the system requirements, and userstories. 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.

10. what is v model?

V- model is the SDLC model where execution will take place in a sequential manner in V- shape. It is also called as verification and validation model. It is the extension of the waterfall model.

what is STLC?

Ans: STLC: Software Test Life Cycle. It consists of series of activities carried out methodologically to help certify your software product.



what is defect?

Ans: A defect is an error or a bug, in the application which is created. 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 a software application or product then it results into a defect. Hence, any deviation from the specification mentioned in the product functional specification document is a defect. In different organizations it’s called differently like bug, issue, incidents or problem.

When the result of the software application or product does not meet with the end user expectations or the software requirements then it results into a Bug or Defect. These defects or bugs occur because of an error in logic or in coding which results into the [failure](http://istqbexamcertification.com/what-is-a-failure-in-software-testing/)or unpredicted or unanticipated results.

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

Ans: 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.

<http://istqbexamcertification.com/when-do-defects-in-software-testing-arise/>

defect lifecycle

Ans:



<http://istqbexamcertification.com/what-is-a-defect-life-cycle/>

what tester will do in each phase of SDLC?

Ans: Based on the above SDLC: The different roles could be defined as..

Project Manager: This would include

* Tracking the project progress, address issues.
* Supporting PDD
* Estimation Schedule
* Researching Risk Mgnt, Cust. Relation Mgnt

Business Analyst: This would include

* Requirements Gathering
* Gap Analysis with existing system or process
* Identify Actors, Use cases
* Work flow, Sequence diagram
* Screen Design
* Facilitate UAT

Architect: includes

* Workflow analysis: focus on specification of the system
* Architectural analysis - objects (use case realization)

Requirement and Use case analysis (Analyze class model)

* Object diagram, Sequence diagram
* Class Diagram
* Moving analysis to implementation
* Database Design

Developer: includes

* Analyze Database Design
* Coding and unit testing on Local host

Tester/QA : includes

* Functional/Integrated Testing (not developer)

Release Manager: includes

* Code review, Build war file, deploy to various environments

Different types of testing:

What is unit testing?

Ans: It is a test to check the code whether it is properly working or not as per the requirement. It is done by the developers (Not testers).

When do we use regression testing?

Ans:

* Change in requirements and code is modified according to the requirement
* New feature is added to the software
* Defect fixing
* Performance issue fix

What is integration testing?

Ans: Integration testing is a software testing methodology used to test individual software components or units of code to verify interaction between various software components and detect interface defects. Components are tested as a single group or organized in an iterative manner. After the integration testing, has been performed on the components, they are readily available for system testing.

When do we use integration testing?

Ans:

* A Module in general is designed by an individual software developer whose understanding and programming logic may differ from other programmers. Integration testing becomes necessary to verify the software modules work in unity.
* At the time of module development, there are wide chances of change in requirements by the clients. These new requirements may not be unit tested and hence integration testing becomes necessary.
* Interfaces of the software modules with the database could be erroneous
* External Hardware interfaces, if any, could be erroneous
* Inadequate exception handling could cause issues.

When do we use smoke testing and sanity testing?

Ans: Smoke testing: Smoke Testing is performed after software build to ascertain that the critical functionalities of the program are 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.

Sanity testing: After receiving a software build, with minor changes in code, or functionality, Sanity testing is performed 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.

<http://www.guru99.com/smoke-sanity-testing.html>

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.

What is UAT?

Ans: 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.

UAT is one of the final and critical software project procedures that must occur before newly developed software is rolled out to the market.

<https://www.techopedia.com/definition/3887/user-acceptance-testing-uat>

What is alpha and beta testing?

Ans: Alpha testing: Alpha testing is a type of acceptance testing; performed to identify all possible issues/bugs before releasing the product to everyday users or public. The focus of this testing is to simulate real users by using blackbox and whitebox techniques. The aim is to carry out the tasks that a typical user might perform. Alpha testing is carried out in a lab environment and usually the testers are internal employees of the organization.

Beta testing: Beta Testing of a product is performed by "real users" of the software application in a "real environment" and can be considered as a form of external user acceptance testing.

Beta version of the software is released to a limited number of end-users of the product to obtain feedback on the product quality. Beta testing reduces product failure risks and provides increased quality of the product through customer validation.

When do we use, white box testing and block box testing?

Ans: White box: White Box Testing is the testing of a software solution's internal coding and infrastructure. It focuses primarily on strengthening security, the flow of inputs and outputs through the application, and improving design and usability.

Black box: Black box testing is a software testing techniques in which functionality of the software under test (SUT) is tested without looking at the internal code structure, implementation details and knowledge of internal paths of the software. This type of testing is based entirely on the software requirements and specifications.

what we will do if come across any severity issue before release day?

Ans: Qa needs to know what is the frequency of the bug even if it has high severity or high priority. You can check the frequency by visiting user traffic data such as : how many users visit this feature through the route described in the bug report; how many user actually visit this feature; etc.  
  
If this bug occurrence frequency is high then I would estimate the EXTRA time and budget needed to (a) fix the bug (b) run end-to-end business scenarios (c) perform some regression testing. Of course my wonderful QA team will help me with all the efforts estimation  
  
I will then hold a meeting with PM and BA and discuss about the impact and risk of going live with the bug, and how much money and time will be required to fix the bug should the client decide to go live with this major bug.  
  
I totally agree with Preetchana "Its better to send the release late, but it should be bug free. Whats the fun, if we send the release on time, but it has many defects, that the customer is not satisfied."

when do we use automation testing?

Ans: Automated software testing is a process in which software tools execute pre-scripted tests on a software application before it is released into production.  Automation Testing means using an automation tool to execute your test case suite.   The automation software can also enter test data into the System Under Test ,  compare  expected and actual  results and generate detailed test  reports. Goal of Automation is to reduce number of test cases to be run manually and not eliminate manual testing all together.

  Automated testing is important due to following reasons:

* Manual Testing of all work flows, all fields , all negative scenarios is time and cost consuming
* It is difficult to test for multi lingual sites manually
* Automation does not require Human intervention. You can run automated test unattended (overnight)
* Automation increases  speed of test execution
* Automation helps increase  Test Coverage
* Manual Testing can become boring and hence error prone.

difference between load and performance testing?

Load test: any test that involves to put a determined load on an application to verify how it behaves (i.e.: response time)

Performance test: it is a load test limited by the load defined by the especification of the application - the test is to verify or confirm that the application will work at the planned performance

different types of non-functional testing types?

* Load/Performance testing.
* Compatibility testing.
* Localization testing.
* Security testing.
* Reliability testing.
* Stress testing.
* Usability testing.
* Compliance testing.

what is test case?

A test case is a set of conditions or variables under which a tester will determine whether a system under test satisfies requirements or works correctly. The process of developing test cases can also help find problems in the requirements or design of an application.

what is test plan/test strategy document

Ans: Test plan document contains different section like

       Types of testing :

       Exit and Entry criteria :

what is TDD and BDD (cucumber framework)

Test-driven development (TDD) is a software development process that relies on the repetition of a very short development cycle: requirements are turned into very specific test cases, then the software is improved to pass the new tests, only.

Behavior-driven development (BDD) is a software development methodology in which an application is specified and designed by describing how its behavior should appear to an outside observer.

what is priority and severity in defect?

Severity of a defect is related to how severe a bug is. Usually the severity is defined in terms of financial loss, damage to environment, company's reputation and loss of life. Priority of a defect is related to how quickly a bug should be fixed and deployed to live servers

how to estimate test cases?

3-Point Software Testing Estimation Technique.

Use – Case Point Method:

Work Breakdown Structure.

Wideband Delphi technique.

Function Point/Testing Point Analysis.

Percentage of development effort method.

Percentage distribution.

what is most challenge defect u came across?

http://www.softwaretestinghelp.com/manual-and-automation-testing-challenges/

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

test design techniques:

Boundary Value Analysis (BVA)

Equivalence Partitioning (EP)

Decision Table Testing.

State Transition Diagrams.

Use case testing

if we dont have time to test call test cases what we will do?

how we learn the functionality of system?

The process of testing of an integrated hardware and software system to verify that the system meets its specified requirements.

Verification: Confirmation by examination and provisions of objective evidence that specified requirements have been fulfilled

what are the tools to manage defects/stories?

Features of a defect management tool:

* Provides a centralized repository for tracking defects across projects.
* Provides automated notifications of resource assignments.
* Ability to define defect resolution status in order to map back to your defect management process.
* Ability to provide management reporting, like the number of open defects grouped by various criteria such as open defects by project, severity, and priority

who will assign the work?

Business analyst and team lead

types of test metrics we use normally

Process Metrics: It can be used to improve the process efficiency of the SDLC ( Software Development Life Cycle)

Product Metrics: It deals with the quality of the software product

Project Metrics: It can be used to measure the efficiency of a project team or any tools being used by the team members

**what is traceability matrix?**

 A testing requirements traceability matrix is a document that traces and maps user requirements, usually requirement IDs from a requirement specification document, with the test case

**what are typical environments we have in projects**

Risk such as the late delivery of the test items to the test team or availability issues with the test environment.

There are also indirect risks such as excessive delays in repairing defects found in testing or problems with getting professional system administration support for the test environment

**what are different defect metrics and measurements we prepare**

<http://www.softwaretestinghelp.com/software-test-metrics-and-measurements/>

**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. Stagingprovides a true basis for QA testing because it precisely reproduces what is in production.

**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 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

**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 Waveset application. A Production environment is where the Waveset application is actually available for business use.

After the application was developed to an agreed stage it is released to the QA environment. This is where the testers ensures the quality of the application, open bugs and review bug fixes. This environment must resemble the production environment accurately, because this is the last safe place to find and fix environment-related bugs.

**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

A production environment can be a real-time setting where programs are run and hardware setups are installed and relied on for organization or commercial daily operations. In the production environment, the product has been delivered and needs to work flawlessly. Here the product wil be available to all users of the system.