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

Software development life cycle (SDLC) is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

Different phases like:

* Planning and Requirement Analysis

Requirement analysis is the most important and fundamental stage in SDLC. It is performed by the senior members of the team with inputs from the customer, the sales department, market surveys and domain experts in the industry. This information is then used to plan the basic project approach and to conduct product feasibility study in the economical, operational, and technical areas.

Planning for the quality assurance requirements and identification of the risks associated with the project is also done in the planning stage. The outcome of the technical feasibility study is to define the various technical approaches that can be followed to implement the project successfully with minimum risks.

* Defining Requirements

Once the requirement analysis is done the next step is to clearly define and document the product requirements and get them approved from the customer or the market analysts. This is done through SRS (Software Requirement Specification) document which consists of all the product requirements to be designed and developed during the project life cycle.

* Designing the product architecture

SRS is the reference for product architects to come out with the best architecture for the product to be developed. Based on the requirements specified in SRS, usually more than one design approach for the product architecture is proposed and documented in a DDS - Design Document Specification.

A design approach clearly defines all the architectural modules of the product along with its communication, data flow representation with the external and third party modules (if any). The internal design of all the modules of the proposed architecture should be clearly defined with the minutest of the details in DDS.

* Building or Developing the Product

In this stage of SDLC the actual development starts and the product is built. The programming code is generated as per DDS during this stage. If the design is performed in a detailed and organized manner, code generation can be accomplished without much hassle. In general, developers generate the code and build the functionality.

* Testing the Product

This stage is usually a subset of all the stages as in the modern SDLC models, the testing activities are mostly involved in all the stages of SDLC. However this stage refers to the testing only stage of the product where products defects are reported, tracked, fixed and retested, until the product reaches the quality standards defined in the SRS.

* Deployment in the Market and Maintenance

Once the product is tested and ready to be deployed it is released formally in the appropriate market. Sometime product deployment happens in stages as per the organizations business strategy. The product may first be released in a limited segment and tested in the real business environment (UAT- User acceptance testing).

Then based on the feedback, the product may be released as it is or with suggested enhancements in the targeting market segment. After the product is released in the market, its maintenance is done for the existing customer base.

**2) What is waterfall in SDLC?**

Waterfall model is traditional SDLC model and is of sequential type. Sequential means that the next phase can start only after the completion of first phase. The phases like requirement analysis, designing, coding, testing, deployment and maintenance.

**3. What is the process in agile model?**

Agile is the most popular model used in the industry. Agile introduces the concept of fast delivery to customers using prototype approach. Agile divides the project into small iterations with specific deliverable features. Customer interaction is the backbone of Agile methodology, and open communication with minimum documentation are the typical features of Agile development environment. All phases are executed parallely so that the defects at early stage can be detected. Modules are tested once coding is completed.

**4. What is scrum methodology?**

It is an agile framework and they follow work in sprints and they have daily stand-up meetings. Scrum sprints used to be 30 days long, but today we advise one-week or two-week sprints. Scrum is iterative and incremental.

**5. What is daily standup meeting and what we discuss?**

Daily standup meetings is to have quick status updates of a team for 10-15minutes.

**6. What is user story and tasks in user story?**

User story involves requirements specified by the user or client.

**7. What is sprint planning and sprint retro?**

The sprint planning meeting is a review of the product backlog items the Product Owner will ask the team to forecast and deliver. In the next part of the sprint planning, the team decides how the work will be built.

Sprint retro is a meeting at which the team discusses the just concluded sprint and determines the changes that make the next sprint more productive.

**8. 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. The velocity report tracks the story points completed in a specified timeframe. It enables team members to determine how well they are keeping to schedule and to measure their performance over time.

**9. What is product backlog item and sprint backlog items?**

* Product Backlog: A product backlog is a prioritized list of work for the development team that is derived from the roadmap and its requirements. The most important items are shown at the top of the product backlog so the team knows what to deliver first. The development team doesn't work through the backlog at the [product owner's](https://www.atlassian.com/agile/product-management) pace and the product owner isn't pushing work to the development team. Instead, the development team pulls work from the product backlog.
* Sprint Backlog: The sprint backlog is a list of tasks identified by the Scrum team to be completed during the [Scrum](https://www.mountaingoatsoftware.com/agile/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.

**10. What is V model, STLC and defect?**

**V-Model:** The process is followed in V- shape and it is also called as verification and validation process. As the development team works on verification parallely testing team does the validation.

**STLC**: Software Testing Life Cycle is a testing process which is executed in a sequence, in order to meet the quality goals. It is not a single activity but it consists of many different activities which are executed to achieve a good quality product.

**Defect:** The defect is raised by tester when the functionality or behavior of the product developed does not meet the user requirement and specifications.

**11. How to arise a defect and what we specify while logging defect? What is defect lifecycle?**

To raise a bug we use some tools like for example HP quality center, if we need to raise it manually we can specify in an excel sheet by providing ID, severity, priority, bug description and summary.

**Defect life cycle/ bug life cycle**

Once a defect is raised, the life cycle of defect starts.

New- when defect is raised by tester

Assign- when it is assigned to developer

Active - when defect is addressed by any developer

Deferred- defect can be deferred means the bug will be fixed in next phase release.

Rejected- bug is duplicate or it is not a defect

Fixed- defect got fixed

**12. When do we use regression testing? When do we use integration testing?**

Regression testing is nothing but re-testing, it is performed when any changes are made in the developed code. The modified code and the connected modules are tested again to check if any functionalities are affected by these changes or not.

Integration testing is performed after unit testing, all the modules developed are combined and tested to check the functionality behavior of the complete developed software.

**13. What is UAT?**

UAT is user acceptance testing, actual software users/clients uses developed software and test whether it is performed according to the user requirements or not.

**14. What is alpha and beta testing?**

Alpha testing of an application when development is nearing completion, minor design changes may still be made as a result of such testing. Typically done by end-users or others, not by programmers or testers.

Beta testing when development and testing are essentially completed and final bugs and problems need to be found before final release. Typically done by end-users or others, not by programmers or testers

**15. When do we use white box testing and block box testing?**

Black-Box testing is not based on any knowledge of internal design or code.Tests are based on requirements and functionality.

White box testing is based on knowledge of the internal program design and code. Tests are based on coverage of code statements, branches, paths, conditions.

**16. What we will do if come across any severity issue before release day?**

Few options to deal with severity issues before release day are

1. Explain the situation and ask the client to give some time to fix the bug
2. If client is not ready to give time than mention the limitations in the release notes as known bugs or limitations. This is a way to overcome defect effect. Normally known issues will be fixed in the next release of the software.

**17. When do we use automation testing?**

To increase the effectiveness, efficiency and coverage of the software testing, we use Automation testing.

**18. What tester will do in each phase of SDLC?**

* Inception Phase

In this phase, a test engineer will get an opportunity to indentify 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 indentify 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 document.

* 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 have to 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 on the basis of all probable references of the newest modernized data and also signed-off.

* Transition Phase: In this phase if any fault or errors are originate 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.

**19. Difference between load and performance testing? Different types of non­functional testing types?**

Performance testing:

Performance Testing is performed to evaluate application performance under some load and stress condition.

Load testing:

Testing application under heavy loads, such as

1) Testing of websites under range of loads.

2) To determine the threshold point where the system’s response time deviates or fails from the normal behavior.

Non-functional testing:

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

**20. 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.

**21. What is test plan/test strategy document?**

Test plan document contains different section like Types of testing:

The test strategy document is a living document created with project requirements as reference. It describes scope, approach, resources and schedule for testing activities of the project.

**22. What is Exit and Entry criteria?**

Entry criteria:

* It ensures that the proper environment is in place to support the entire system test process. Item must meet in entry criteria:
* All test hardware platforms must have been successfully installed, configured and functioning properly.

Exit Criteria:

* It ensures that the project application has been satisfactorily completed before exiting the system test stage and clarifying the application as complete.
* Ensure all application documentation has been completed and is up to date.

**23. What is TDD and BDD (cucumber framework)?**

In software engineering, behavior-driven development (abbreviated BDD) is a software development process based on test-driven development (TDD). Behavior-driven development combines the general techniques and principles of TDD with ideas from domain-driven design and object-oriented analysis and design to provide software development and management teams with shared tools and a shared process to collaborate on software development.

Test-driven development (TDD) is a software development process that relies on the repetition of a very short development cycle: first the developer writes an (initially failing) automated test case that defines a desired improvement or new function, then produces the minimum amount of code to pass that test, and finally refactors the new code to acceptable standards.

**24. How do we write test cases in BDD format?**

Traditionally, structured test, especially long scenarios are commonly written in the step, expected, actual format. Behaviour Driven Design replaces this with the style of "Given", "When" "Then", for example:

Given a working ATM

And invalid credit card

When the card is inserted

And a cash withdrawal is requested

Then no cash should be provided

And the card should be retained

**25. 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**.**

**26. How to estimate test cases?**

We will estimate user stories which contains story points and review test cases, acceptance criteria and incorporate feedback.

**27. What is most challenge defect you came across?**

The most common problem that I have faced in my previous projects are the Java script errors, data connectivity, error, HTTP 500 error (This error occurs when server is down), HTTP 400 error (when file is not found) and so on.

Another problem was that a user would search for branch location information of a bank. The user logs in by using User Name and Password. After the log in, on the “Search Location” page, the user enters and zip code of the location he wants to find, then clicks Find button. After that the system (application) gives a number of branch locations. The user now clicks “Request Information” for one of the branches. As soon as the user clicks “Request Information” button, the application breaks (displays “Page cannot be found” error). I logged this defect as a critical defect.

**28. How to deal the production defects?**

Normally end user will report this issue, we need to talk to them and reproduce the issue with test logins and Create defect in defect tool under the production release version developers will fix the issue. QA test the issue on production version code and release the fix to production 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.

**29. Test design review steps**

Get the test cases review and feedback from the developer assigned to that particular user story.

**30. If we don’t have time to test all test cases what we will do?**

Move it to next sprint.

**31. How we learn the functionality of system?**

By running the application we get to know the functionality and incase if we are not sure of a particular feature and what is exact output to be expected, communicating with the particular developer assigned to that feature or theme will help us**.**

**32. What are the tools to manage defects/stories?**

HP ALM, Jira, Bugzilla, HP agile manager to manage user stories.

**33. Who will assign the work?**

Test lead or Quality Analyst

**34. 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.

**35. What is traceability matrix?**

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.

**36. What are typical environments we have in projects?**

Analysts, development, testing, production and maintenance.

**37. What is development environment?**

A Development environment is where you configure, customize, and use source control to build an application to be promoted to another environment. You also write an upgrade procedure in this environment that you follow in each target environment**.**

**38. 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.

**39. 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**.**

**40. What are different defect metrics and measurements we prepare?**

Urgent, High, Medium, Low.

**41) Why testing is required?**

Testing is required for an effective performance of software application or product and to ensure quality. 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.

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

Web application, mobile application, client-server and windows based applications

**43) What is agile method?**

Agile is a time boxed, iterative approach to software delivery that builds software incrementally from the start of the project, instead of trying to deliver it all at once near the end.

**44) What is sprint review meeting?**

 At the end of each sprint, a sprint review meeting is held. During this meeting, the Scrum team shows what they accomplished during the sprint. Typically this takes the form of a demo of the new features.

**45) What is sprint grooming?**

Backlog grooming is when the product owner and some, or all, of the rest of the team review items on the [backlog](https://www.agilealliance.org/glossary/backlog/) to ensure the backlog contains the appropriate items, that they are prioritized, and that the items at the top of the backlog are ready for delivery

**46) What is user acceptance criteria test cases?**

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.

**47) What is unit testing?**

**Unit testing** is a [level of software testing](http://softwaretestingfundamentals.com/software-testing-levels/) where individual units/ components of a software are tested. The purpose is to validate that each unit of the software performs as designed. It is performed by developers.

**48) What is Integration Testing?**

Integration testing is individual software modules are combined and tested as a group. It occurs after unit testing and before validation testing.

**49) When do we use smoke testing and sanity 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.

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.

**50) When do we use white box testing and block 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. As long as functionality of interfaces remains unchanged, tests should pass even if internals are changed.

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

**51) 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.