1. **What is software testing?**

“Software engineering is the art of developing quality software on time and within budget.”

Software testing is a method to check whether the actual software product mathes expected requirements and to ensure that software product is Defect free.

1. **What is SDLC?**

Software Development Life Cycle (SDLC) is a structure imposed on the development of a software product that defines the process for planning, implementation, testing, documentation, deployment, and ongoing maintenance and support.

1. **What is SRS?**

A software requirements specification(SRS) is a complete description of the behaviour of the system to be developed.

It includes a set of use cases(functional) and non-functional requinments that describe all of the interactions that the users will have with the software.

* Types of Requirements in SRS
  + Customer Requirements

Characteristics or specifications that should be present in a product.

Examples:

Service Requirements-

On-time delivery

Service with a smile

Easy-payment

Output Requinments-

Product specifications like the loudness and clarity of a pair of speakers

* + Functional Requirements:

Describe system services or functions

A requirement that specifies input/output behaviour of the system

* + Non-Functional Requirements-

Describes constraints on the system or the development process. A requirement that specifies system PROPERTIES, such as environmental and implementation constraints, performance, dependencies, maintainability, extensibility and reliability.

1. **What is agile methodology?**

Agile model believes that every project needs to be handled differently and existing methods need to be tailored to best suit the project requirements. In agile the tasks are divided to time boxes (small time frames) to deliver specific features for a release. Iterative approach is taken and working software build is delivered after each iteration. Each build is incremental in terms of features; the final build holds all the features required by the customer. Agile thought process had started early in the software development and started becoming popular with time due to its flexibility and adaptability.

Pros

* Is a very realistic approach to software development
* Promotes teamwork and cross training
* Functionality can be developed rapidly and demonstrated
* Resource requirements are minimum.
* Suitable for fixed or changing requirements
* Delivers early partial working solutions
* Good model for environments that change steadily
* Planned context
* Little or no planning required
* Easy to manage
* Gives flexibility to developers

Cons

* Not suitable for handling complex dependencies.
* More risk of sustainability, maintainability and extensibility
* An overall plan, an agile leader and agile PM practice is a must without which it will not work.
* Strict delivery management dictates the scope, functionality to be delivered, and adjustments to meet the deadlines.
* Depends heavily on customer interaction, so if customer is not clear, team can be driven in the wrong direction.
* Transfer of technology to new team members may be quite challenging due to lack of documentation.

1. **Write SDLC phases with basic introduction?**

Software Development Life Cycle (SDLC) is a structure imposed on the development of a software product that defines the process for planning, implementation, testing, documentation, deployment, and ongoing maintenance and support.

* Requirement Gathering

The requirement is the first stage in the SDLC process. It is conducted by the senior team members with inputs from all the industry. This stage gives a clearer picture of the scope of the entire project and the anticipated issues, opportunities, and directives which triggered the project. Requirements Gathering stage need teams to get detailed and precise requirements. This helps companies to finalize the necessary timeline to finish the work of that system.

The requirements are collected using a number of practices as given

* Studying the existing or obsolete system and software.
* Conducting interviews of users and developers.
* Referring to the database
* Collecting answers from the questionnaires.

Types of Requirements:

* Functional Requirements: describe system services or functions.
* Non-Functional Requirements: describes constraints on the system or the development process.
* Analysis

It defines the requirements of the system, independent of how these requirements will be accomplished. The deliverable result at the end of this phase is a requirement document (SRS-Software Requinments Specification) which describes clear description of all requinments. Ideally, this document describes what is to be built. The requirement documentaries to capture the requirements from the customer’s perspective by defining goals. Details on computer programming languages and environments, machines, packages, application architecture, distributed architecture layering, memory size, platform, algorithms, data structures, global type definitions, interfaces, and many other engineering details are established. This SRS needs to share it with the customer or with the business analyst for the approval.

* Design Phase

In this third phase, software design documents are prepared as per the requirement specification document.

This helps define overall system architecture.

Two kinds of design documents developed in this phase:

High-Level Design(HLD)

* + Brief description and name of each module
  + An outline about the functionality of every module
  + Interface relationship and dependencies between modules
  + Database table identified along with their key elements
  + Complete architecture diagrams along with technology details

Low-Level Design(LLD)

* + Functional logic of the modules
  + Database tables, which include type and size
  + Complete details of the interface
  + Addresses all types of dependency issues
  + Listing of error messages
  + Complete input and outputs for every module
* Implementation/Coding Phase

 In this phase, developers start build the entire system by writing code using the chosen programming language. In the coding phase, tasks are divided into units or modules and assigned to the various developers. It is the longest phase of the Software Development Life Cycle process. In this phase, Developer needs to follow certain predefined coding guidelines. They also need to use [programming tools](https://www.guru99.com/software-development-tools.html) like compiler, interpreters, debugger to generate and implement the code.

* Testing Phase

Once the software is complete, and it is deployed in the testing environment. The testing team starts testing the functionality of the entire system. This is done to verify that the entire application works according to the customer requirement. During this phase, QA and testing team may find some bugs/defects which they communicate to developers. The development team fixes the bug and send back to QA for a re-test. This process continues until the software is bug-free, stable, and working according to the business needs of that system.

* Deployment Phase

Once the software testing phase is over and no bugs or errors left in the system then the final deployment process starts. Based on the feedback given by the project manager, the final software is released and checked for deployment issues if any.

* Maintenance Phase

After the software clears all the SDLC phases without any issues, then it goes into the maintenance stage.

Maintenance is the process of changing a system after it has been deployed.

* + Corrective maintenance: identifying and repairing defects
  + Adaptive maintenance: adapting the existing solution to the new platforms.
  + Perfective maintenance: implementing the new requirements.

following 3 activities occur

* + Bug fixing – bugs are reported because of some scenarios which are not tested at all
  + Upgrade – Upgrading the application to the newer versions of the Software
  + Enhancement – Adding some new features into the existing software

1. **Explain Phases of the waterfall model?**

* Requirements

Potential requirements, deadlines and guidelines for the project are analyzed and placed into a formal requirements document, also called a functional specification. This stage of development defines and plans the project without mentioning specific processes.

* Design

A design specification document is created to outline technical design requirements, such as the programming language, [hardware](https://www.techtarget.com/searchnetworking/definition/hardware), data sources, architecture and services.

* Coding

The [source code](https://www.techtarget.com/searchapparchitecture/definition/source-code) is developed using the models, logic and requirement specifications designated in the prior phases. Typically, the system is coded in smaller components, or units, before being put together.

* Testing

This is when [quality assurance](https://www.techtarget.com/searchsoftwarequality/definition/quality-assurance), [unit](https://www.techtarget.com/searchsoftwarequality/definition/unit-testing), [system](https://www.techtarget.com/searchsoftwarequality/definition/system-testing) and [beta](https://www.techtarget.com/whatis/definition/beta-test) tests identify issues that must be resolved. This may cause a forced repeat of the coding stage for [debugging](https://www.techtarget.com/searchsoftwarequality/definition/debugging). If the system passes integration and testing, the waterfall continues forward.

* Maintenance

Corrective, adaptive and perfective maintenance is carried out indefinitely to improve, update and enhance the product and its functionality. This could include releasing [patch](https://www.techtarget.com/searchenterprisedesktop/definition/patch) updates and new versions.

1. **Write phases of spiral model?**

The spiral model has four phases. A software project repeatedly passes through these phases in iterations called Spirals.

* Identification

This phase starts with gathering the business requirements in the baseline spiral. In the subsequent spirals as the product matures, identification of system requirements, subsystem requirements and unit requirements are all done in this phase. This phase also includes understanding the system requirements by continuous communication between the customer and the system analyst. At the end of the spiral, the product is deployed in the identified market.

* Design

The Design phase starts with the conceptual design in the baseline spiral and involves architectural design, logical design of modules, physical product design and the final design in the subsequent spirals.

* Construct or Build

The Construct phase refers to production of the actual software product at every spiral. In the baseline spiral, when the product is just thought of and the design is being developed a POC (Proof of Concept) is developed in this phase to get customer feedback. Then in the subsequent spirals with higher clarity on requirements and design details a working model of the software called build is produced with a version number. These builds are sent to the customer for feedback.

### Evaluation and Risk Analysis

Risk Analysis includes identifying, estimating and monitoring the technical feasibility and management risks, such as schedule slippage and cost overrun. After testing the build, at the end of first iteration, the customer evaluates the software and provides feedback.

