

* Software Testing And Quality Assurance.

1. explain software testing life cycle (STLC).
2. What are Seven testing principles ? explain it in detail.
3. What is V-model and W-model ?
4. What is unit and integration testing ?
5. Differentiate between smoke and sanity testing.
6. Differentiate between Localization and Internationalization of testing.
7. Differentiate between Resting and regression testing.

1 explain software testing life cycle (STLC).

Ans → STLC is a sequence of different activities performed during the SW testing process.

* characteristics →

* STLC is a fundamental part of SW development life cycle (SDLC) but STLC consists of only the testing phase.

* STLC starts as soon as requirements are defined or software requirement document is shared by stakeholders.

* STLC yields a step-by-step process to ensure quality software.

In the initial stages of STLC, while the software product or the application is being developed, the testing team analyzes and defines the scope of testing, entry and exit criteria and also the test cases. It helps to reduce the test cycle time and also enhance the product quality.

As soon as the testing development phase is over, testing team is ready.

with test cases and start the execution. This helps in finding bugs in the early phase.

* phases of STLC. →

Requirement analysis



Test planning



Test case development



Test environment setup



Test execution



Test closure

① Requirement Analysis →

Requirement Analysis is the first step of software testing life cycle (STLC). In this phase, quality assurance team understands the requirements like what is to be tested. If anything is missing or not understandable then quality assurance team meets with the stakeholders to better understand the detail knowledge of requirement.

② Testing planning →

Test planning is most efficient phase of software testing life cycle where all testing plans are defined. In this phase manager of the testing team calculates estimated effort and cost for the testing work. This phase gets started once the requirement gathering phase is completed.

③ Test case development →

The test case development phase gets started once the test planning phase is completed. In this phase testing team note down the detailed test cases. Testing team also prepare the required test data for the testing. When the test cases are prepared then they are reviewed by quality assurance team.

④ Test environment setup →

Test environment setup is the vital part of the STLC. Basically test environment decides the conditions on which software is tested. This is independent activity and can be started along with test case development. In this process testing team is not involved. either the

developer or the customer creates the testing environment.

⑤ Test execution →

After the test case development and test environment setup test execution phase gets started. In this phase testing team start executing test cases based on prepared test cases. In the earlier step.

⑥ Test closure →

This is the last stage of STLC. in which the process of testing is analyzed.

2. what are seven testing principle ? explain it in detail.

Ans → software testing is the process of executing a program with the aim of finding the error. To make our sw. program well it should be error-free. If testing is done successfully it will remove all the errors from the sw.

* There are seven principles in sw. testing →

① Testing shows the presence of defects.

② exhaustive testing is not possible.

③ early testing.

④ defect clustering

⑤ pesticide paradox.

⑥ Test is context-dependent.

⑦ absence of errors fallacy.

* Testing shows the presence of defects →

The goal of sw. testing is to

make the SW fail. SW testing talks about the presence of defects and doesn't talk about the presence of defects and doesn't talk about the absence of defects. SW testing can ensure that defects ~~that~~ are present but it can not prove that software is defect-free. Even multiple testing can never ensure that SW is 100% bug-free. Testing can reduce the number of defects but not removes all defects.

* exhaustive testing is not possible →

It is the process of testing the functionality of the SW in all possible inputs (valid or invalid) and pre-conditions is known as exhaustive testing. exhaustive testing is impossible means the SW can never test at every test case. It can test only some test cases and assume that the software is correct and it will produce the correct output in every test case then it will take more cost, effort, etc. and which is impractical.

* Early testing →.

To find the defect in the SW, early test activity shall be started. The

defect detected in the early phases of SDLC will be very less expensive. For better performance of SW, software testing will start at the initial phase i.e. testing will perform at the requirement analysis phase.

* defect clustering →

In a project, a small number of the module can contain most of the defects. pareto principle to SW testing state that 80% of SW defect comes from 20% of modules.

* pesticide paradox →

Repeating the same test cases again and again, will not find new bugs so it is necessary to review the test cases and add or update test cases to find new bugs.

* Testing is context-dependent →

The testing approach depends on the context of the SW developed. different types of SW need to perform different types of testing. For example, the testing of the e-commerce site is different from the testing of the

android application.

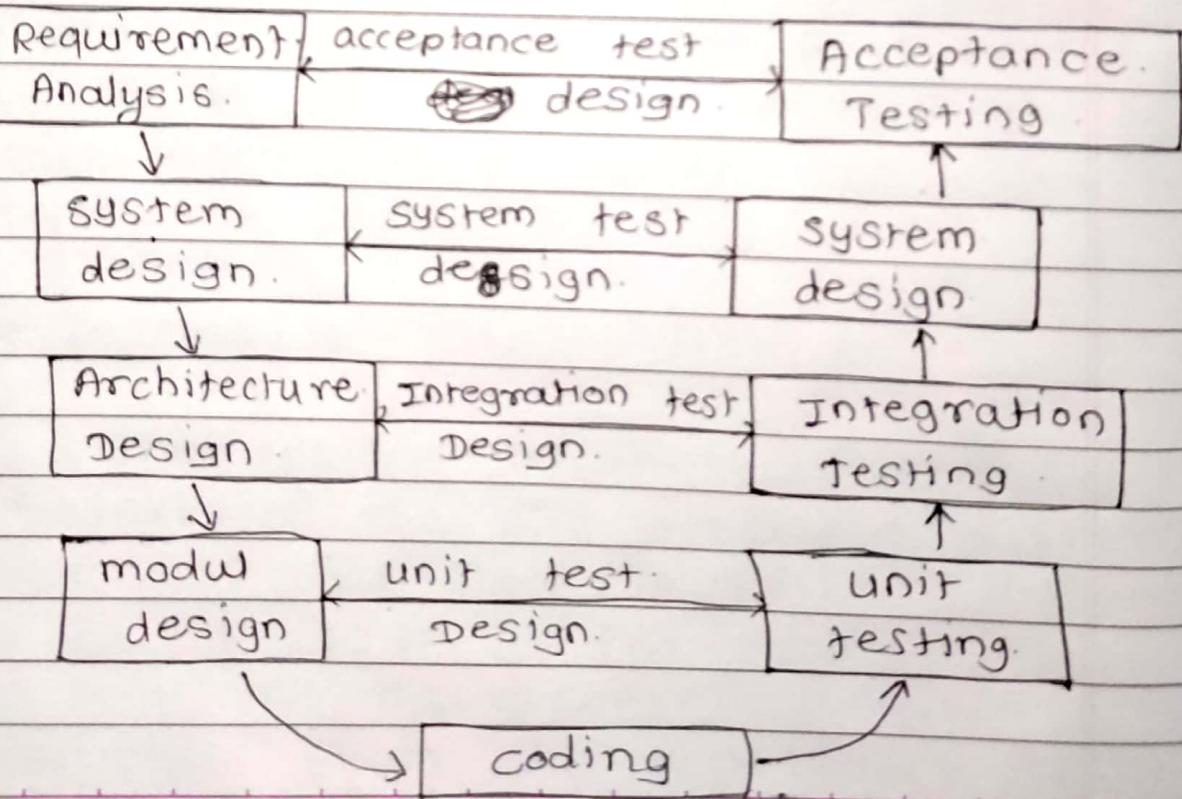
* absence of errors Fallacy →

In a build software is 99% bug free but it does not follow the user requirement then it is unusable. It is not only necessary that software is 99% bug free but it is also mandatory to fulfill all the customer requirements.

3. what is v-model and w-model ?

Ans → * SDLC V-Model →

The v-model is a type of SDLC model where process executes in a sequential manner in v-shape. It is also known as verification and validation model. It is based on the association of a testing phase. The next for each corresponding development stage development of each step directly associated with the testing phase. The next phase starts only after completion of the previous phase. i.e. For each development activity, there is a testing activity corresponding to it.



Verification →

It involves static analysis technique (review) done without executing code. It is the process of evaluation of the product development phase to find whether specified requirement meet.

Validation →

It involves dynamic analysis technique (Functional, non-Functional), testing done by executing code. Validation is the process to evaluate the SW after the completion of the development phase to determine whether SW meets the customer expectations and requirements.

So V-model contain verification phase on one side of the validation phases on the other side. Verification and validation phases are joined by coding phase in V-shape. Thus it is called V-model.

* design phase →

① Requirement Analysis.

② System design.

③ Architectural design.

④ module design.

* Testing phases →

① unit testing.

② Integration testing.

③ system testing.

④ user acceptance testing (UAT)

* principles of V-model →

① large to small

② data & process integrity.

③ scalability.

④ cross referencing.

⑤ Tangible documentation.

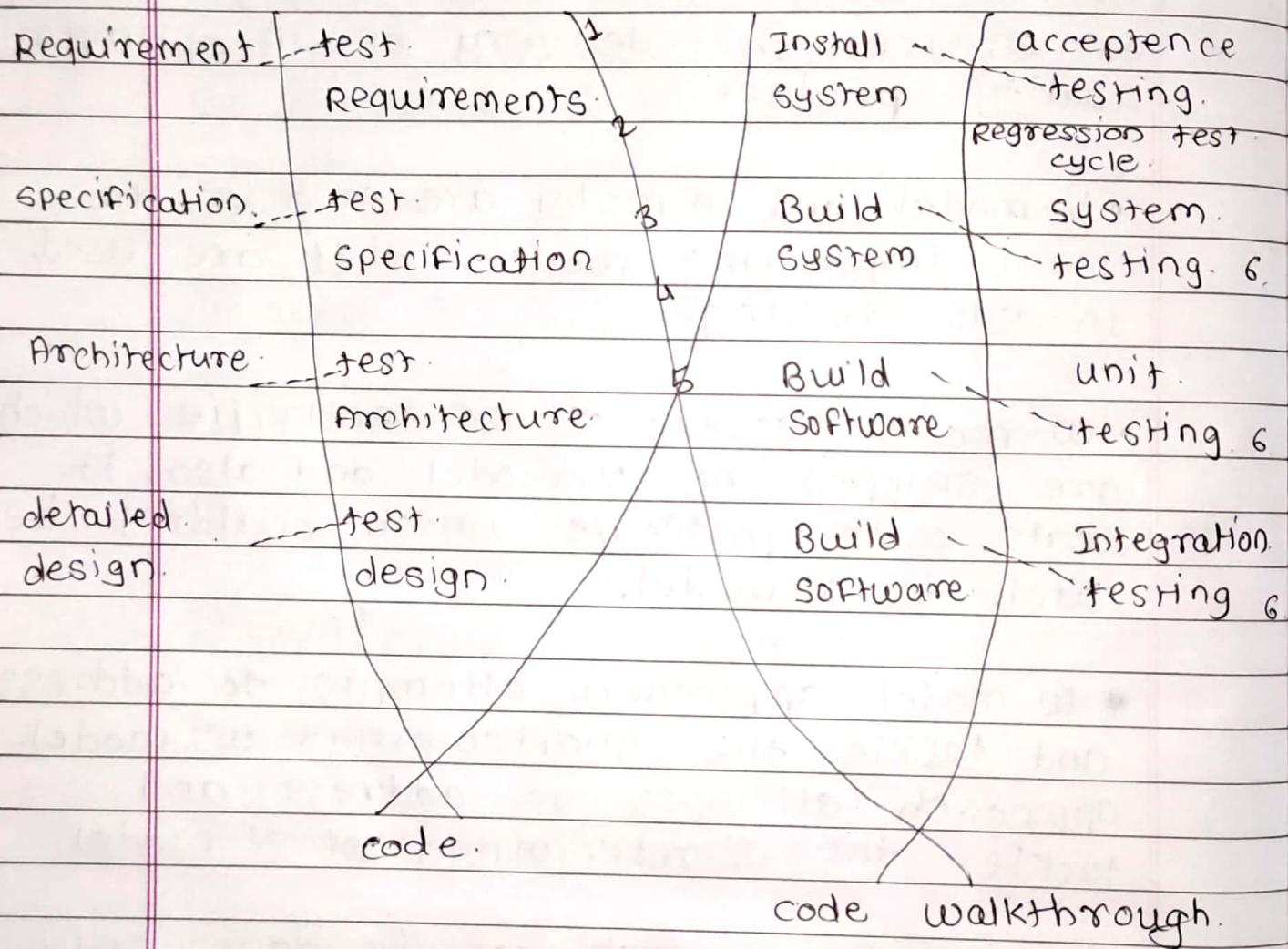
* SDLC. W-model →

W-model is the most recent software development model where we start real testing activity simultaneously.

development process starts, whereas software development process is a method, in which a SW or product is made through various stages of planning, development and testing before the final SW or product is delivered, testing is such a stage that is extremely crucial to ensure the delivery of an optimum quality product.

- V-model and W-model are two of the most important models that are used in SW testing.
- W-model covers those activities which are skipped by V-model and also, it deals with problems which couldn't be catch by V-model.
- W-model approach attempts to address and tackle the shortcomings W-model approach attempts to address and tackle the shortcomings of V-model.
- W-model approach can be done only once the development of the product is complete with no modifications required to be done in between. This type of testing is most suitable for short projects.

- with help of W-model, we ensure that the testing of the product starts from the very first day of the inception of product and each phase of the product development is verified and validated.



4 what is unit and Integration testing?

Ans → ~~* 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 is often automated but it can also be done manually.

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. This testing methodology is done during the development process by the software developers and sometimes QA staff. The main objective of unit testing is to isolate written code to test and determine if it works as intended.

unit testing is an important step in the development process, because if done correctly, it can help detect early flaws in code which may be more difficult to find in later testing stages.

unit testing is a component of,

test-driven development (TDD), a pragmatic methodology that takes a meticulous approach to building a product by means of continual testing and revision. This testing method is also the first level of SW testing, which is performed before other testing. Methods such as Integration testing. Unit tests are typically isolated to ensure a unit does not rely on any external code or functions. Testing can be done manually but is often automated.

* Integration testing →

Integration testing is the process of testing the interface between two SW units or module. Its focus on determining the correctness of the interface. The purpose of the integration between Integration testing is to expose faults in the interaction ~~testing~~ between integrated units. Once all the modules have been unit tested, integration testing is performed.

5 Differentiate between smoke and sanity testing.

Ans → * Smoke Testing →

- ① smoke testing is done to assure that the acute functionalities of program is working fine.
- ② smoke testing is also called subset of acceptance testing.
- ③ smoke testing is performed by either developers or testers.
- ④ smoke testing is documented.
- ⑤ smoke testing is scripted.
- ⑥ smoke testing may be stable or unstable.
- ⑦ smoke testing is done to measures the stability of the system/product by performing testing.
- ⑧ smoke testing is used to test all over function. of the system/product.
- ⑨ smoke testing can be performed either manually or by using automation.

tools:

- ⑩ Smoke testing is performed when new product is built.

* Sanity Testing →

- ① sanity testing is done to check the bugs have been fixed after the build.
- ② sanity testing ~~isn't document~~ is also called subset of regression testing.
- ③ sanity testing is normally performed by testers.
- ④ sanity testing isn't documented.
- ⑤ sanity testing is usually not scripted
- ⑥ sanity testing is stable.
- ⑦ sanity testing is done to measures the rationality of the system / product by performing testing.
- ⑧ sanity testing is used in the case of only modified or defect functions of system / products.

- ⑨ sanity testing is commonly executed manually, not by using any automation approach.
- ⑩ sanity testing is conducted after the completion of regression testing.

6. Differentiate between Localization and Internationalization of testing.

Ans → * Localization testing →

- ① Localization is defined as making a product, application, or document content adaptable to meet the cultural, lingual and other requirements of a specific region, or a locale.
- ② Localization is referred as I10n.
- ③ Localization focuses on online helps, GUI context, dialog, boxes, error message, read me tutorials, user manuals, release notes, Installation guide etc.
- ④ Localization itself means a specific local language for any given region.
- ⑤ Localization is not at user interface level.

* Internationalization. ~~is the pro~~ testing →

- ① Internationalization is the process of designing and developing a product, application, or document content such that it enables

Localization.

- ② Internationalization is referred as i18n
- ③ Internationalization focuses on compatibility testing, functionality testing, interoperability testing, usability testing, installation testing, user interface, user validation testing.
- ④ Application code is independent of language.
- ⑤ Internationalization is at design level.

7 Differentiate between Resting and Regression testing.

Ans → * Regression testing →

- ① Regression testing is known as a generic testing.
- ② Regression testing is to ensure that changes have not affected the unchanged part of product.
- ③ Regression testing is used for passed test cases.
- ④ defect verification is not coming under regression testing.
- ⑤ Regression testing can be done either in automation or manual testing.
- ⑥ Regression testing has lower priority than retesting testing, but in some cases it can be done in parallel with retesting.
- ⑦ passed test cases can be executed during regression testing.
- ⑧ during regression testing test cases can be automated.

* Retesting →

- ① Retesting is known as planned testing.
- ② Retesting is used to ensure the test cases which failed in last execution are fixed.
- ③ Retesting is used only for failed test cases.
- ④ Defect verification is coming under retesting.
- ⑤ Retesting can not be automated.
- ⑥ Retesting has higher priority than regression testing.
- ⑦ Only failed test cases are re-executed during retesting.
- ⑧ During retesting, test cases can't be automated.