* **What is software testing?**

Software testing is a process used to identify the correctness completeness and quality of developed computer software.

* **What is exploratory testing?**
* **What is traceability matrix?**

To protect against changes you should be able to trace back from every system component to the original requirement that caused it’s presence.

* **What is boundary value testing?**

Boundary value analysis is a methodology for designing test cases the concentrates software testing efforts on cases near the limits of valid ranges.

* **What is equivalence partitioning testing?**

Equivalence partitioning is the process of defining the optimum number of test by reviewing the documents such as the functional design specification, detailed design specification, and identifying each input condition.

* **What is integration testing?**

Integration testing is associate with the architectural design phase. System tests check the entire system functionality and the communication of the system under development with external systems.

* **What determines the level of risk?**
* **What is Alpha testing?**
* **What is beta testing?**
* **What is component testing?**

A minimal software item that can be tested in isolation is known as component or unit testing.

* **What is functional system testing?**

Functional testing based on the analysis of the specification of the functionality of a component or a system. Specification –E.g Requirement specification, Use case, Functional specification or maybe undocumented.

* **What is non-functional system testing?**

Testing the attributes of a componentor system that do not relate to functionality, e.g. reliability, efficiency, usability, interoperability, maintainability and portability.

* **What is GUI testing?**
* **What is Adhoc testing?**

Adhoc testing is an informal testing type with an aim to break the system. Main aim of this testing is to find defects by random checking. Adhoc testing can be achieved with the testing technique called Error Guessing

* **What is white box testing and list the types of white box testing?**

Testing based on an analysis of the internal structure of the component or system. White box testing techniques.

* **Based on code and the design of the system.**

1. **Statement coverage**
2. **Brach coverage**
3. **Decision coverage**

* **Based on the knowledge of tester.**

1. **Grey Box**
2. **Error Guessing**
3. **Exploratory testing.**

* **What is block box testing? What are different block box testing techniques?**

Testing, either functional or non-functional, without reference to the internal structure of the component or system. The testers have no knowledge of how the system or component is structured inside the box. In black-box testing the tester is concentrating on what the software does, not how it does it.

Black box testing Techniques based on the requirements.

1. **Equivalence Partitioning.**
2. **Boundary value analysis**
3. **Decision table.**
4. **State transition testing**
5. **Use case testing.**

* **Mention what are the categories of defects?**

1. Data quality/Data base defect
2. Critical Functionality defect
3. Functionality defect
4. Security defect
5. User interface defect.

* **What big bang testing is?**

In Big Bang integration testing all components or modules is integrated simultaneously, after which everything is tested as a whole.

* **What is purpose of exit criteria?**
* **When should “Regression testing” is performed?**

Regression testing means testing software application when It under goes a code change to ensure that the new code has not affected other parts of the software.

* What is 7 key principles? Explain it in detail.

1. **Testing shows presence of error.**
2. **Exhaustive testing is impossible.**
3. **Early testing.**
4. **Defect clustering.**
5. **The pesticide paradox**
6. **Testing is context depended.**
7. **Absence of error fallacy.**

* **Testing shows presence of error.**

Testing can show that defects are present, but cannot prove that there are no defects. Testing reduces the probability of undiscovered defects remaining in the software but, even if no defects are found, it is not a proof of correctness.

* **Exhaustive testing is impossible**

Testing everything including all combinations of inputs and preconditions is not possible.

* **Early testing**

Testing activities should start as early as possible in the software or system development life cycle, and should be focused on defined objectives. Testing activities should start as early as possible in the development lifecycle.

* **Defect clustering**

A small number of modules contain most of the defects discovered during pre- release testing, or are responsible for the most operational failures. Defects are not evenly spread in a system they are clustered.

In other words, most defects found during testing are usually confined to a small number of modules similarly, most operational failures of a system are usually confined.

* **Pesticide paradox**

If the same tests are repeated overland over again eventually the same set of test cases will no longer find any new defects. To overcome this “pesticide paradox”, the test cases need to be regularly reviewed and revised, and new and different tests need to be written to exercise different parts of the software or system to potentially find more defects.

* **Testing is context depended**

Testing is basically context dependent. Testing is done differently in different contexts. Different kinds of sites are tested differently. For example Safety critical software is tested differently from an E-commerce site.

* **Absence of error fallacy**

If the system built is unusable and does not fulfill the user’s needs and expectations then finding and fixing defects does not help.

* **Difference between verification and validation.**

|  |  |
| --- | --- |
| **Verification** | **Validation** |
| The process of evaluating work-products of a development phase to determine whether they meet the specified requirements for that phase. | The process of evaluating software during or at the end of the development process to determine whether it satisfies specified business requirements. |
| To ensure that the product is being built according to the requirements and design specifications. In other words, to ensure that work products meet their specified requirements. | To ensure that the product actually meets the user’s needs, and that the specifications were correct in the first place. In other words, to demonstrate that the product fulfills its intended use when placed in its intended environment. |
| It is done before coding | It is done after coding |
| Activity of verification is  Reviews  Walkthroughs  Inspections | Activity of validation  Testing |

* **Different between smoke and sanity.**

|  |  |
| --- | --- |
| **Smoke** | **Sanity** |
| Smoke testing is performed to ascertain that critical functionality of the program works fine. | Sanity testing is performed to check new functionality is working well or not. |
| This testing is performed by developer or by tester | This activity is performed by the tester. |
| Smoke testing is subset of regression testing. | Sanity testing is a subset of acceptance testing. |
| Smoke testing exercises the entire system from end to end. | Sanity testing exercises only the particular component of the system |
| Smoke testing is usually documented or scripted. | Sanity testing is not documented and is unscripted. |

* **Different between QA Vs QC Vs Tester.**

|  |  |  |
| --- | --- | --- |
| **QA** | **QC** | **TESTER** |
| Activity which ensure the implementation of process, procedures and standers in context to verification of developed software intended requirements. | Activity which ensure the verification of the development with respect to documented | Activity which ensure the identification of error bug and defect in the development. |
| Process oriented activity | Product oriented activity | Product oriented activity |
| Preventive activity | It’s a corrective activity | Preventive activity |
| it is subset of STLC | It is subset of QA | It’s subset of QC |