1. What are the components of an SRS?

An SRS contains the following basic components:

Introduction

Overall Description

External Interface Requirements

System Requirements

System Features

2. What is the difference between a test plan and a QA plan?

A test plan lays out what is to be done to test the product and includes how

quality control will work to identify errors and defects. A QA plan on the

other hand is more concerned with prevention of errors and defects rather than

testing and fixing them.

3. How do you test an application if the requirements are not available?

If requirements documentation is not available for an application, a test plan

can be written based on assumptions made about the application. Assumptions

that are made should be well documented in the test plan.

4. What is a peer review?

Peer reviews are reviews conducted among people that work on the same team. For

example, a test case that was written by one QA engineer may be reviewed by a

developer and/or another QA engineer.

5. How can you tell when enough test cases have been created to adequately test

a system or module?

You can tell that enough test cases have been created when there is at least one

test case to cover every requirement. This ensures that all designed features

of the application are being tested.

6. Who approves test cases?

The approver of test cases varies from one organization to the next. In some

organizations, the QA lead may approve the test cases while another approves

them as part of peer reviews.

7. Give an example of what can be done when a bug is found.

When a bug is found, it is a good idea to run more tests to be sure that the

problem witnessed can be clearly detailed. For example, let say a test case

fails when Animal=Cat and. A tester should run more tests to be sure that the

same problem doesn’t exist with Animal=dog. Once the tester is sure of the full

scope of the bug can be documented and the bug adequately reported.

8. Who writes test plans and test cases?

Test plans are typically written by the quality assurance lead while testers

usually write test cases.

9. Is quality assurance and testing the same?

Quality assurance and testing is not the same. Testing is considered to be a

subset of QA. QA is should be incorporated throughout the software development

life cycle while testing is the phase that occurs after the coding phase.

Typical Manual Testing Interview Questions:

10. What is a negative test case?

Negative test cases are created based on the idea of testing in a destructive

manner. For example, testing what will happen if inappropriate inputs are

entered into the application.

11. If an application is in production, and one module of code is modified, is

it necessary to retest just that module or should all of the other modules be

tested as well?

It is a good idea to perform regression testing and to check all of the other

modules as well. At the least, system testing should be performed.

12. What should be included in a test strategy?

The test strategy includes a plan for how to test the application and exactly

what will be tested (user interface, modules, processes, etc.). It establishes

limits for testing and indicates whether manual or automated testing will be

used.

13. What can be done to develop a test for a system if there are no functional

specifications or any system and development documents?

When there are no functional specifications or system development documents, the

tester should familiarize themselves with the product and the code. It may also

be helpful to perform research to find similar products on the market.

14. What are the functional testing types?

The following are the types of functional testing:

Compatibility

Configuration

Error handling

Functionality

Input domain

Installation

Inter-systems

Recovery

15. What is the difference between sanity testing and smoke testing?

When sanity testing is conducted, the product is sent through a preliminary

round of testing with the test group in order to check the basic functionality

such as button functionality. Smoke testing, on the other hand is conducted by

developers based on the requirements of the client.

16. Explain random testing.

Random testing involves checking how the application handles input data that is

generated at random. Data types are typically ignored and a random sequence of

letter, numbers, and other characters are inputted into the data field.

17. Define smoke testing.

Smoke testing is a form of software testing that is not exhaustive and checks

only the most crucial components of the software but does not check in more

detail.

Advanced Manual Testing Interview Questions

18. What steps are involved in sanity testing?

Sanity testing is very similar to smoke testing. It is the initial testing of a

component or application that is done to make sure that it is functioning at the

most basic level and it is stable enough to continue more detailed testing.

19. What is the difference between WinRunner and Rational Robot?

WinRunner is a functional test tool but Rational Robot is capable of both

functional and performance testing. Also, WinRunner has 4 verification points

and Rational Robot has 13 verification points.

20. What is the purpose of the testing process?

The purpose of the testing process is to verifying that input data produces the

anticipated output.

21. What is the difference between QA and testing?

The goals of QA are very different from the goals of testing. The purpose of QA

is to prevent errors is the application while the purpose of testing is to find

errors.

22. What is the difference between Quality Control and Quality Assurance?

Quality control (QC) and quality assurance (QA) are closely linked but are very

different concepts. While QC evaluates a developed product, the purpose of QA is

to ensure that the development process is at a level that makes certain that the

system or application will meet the requirements.

23. What is the difference between regression testing and retesting?

Regression testing is performing tests to ensure that modifications to a module

or system do not have a negative effect on previous releases. Retesting is

merely running the same testing again.

24. Explain the difference between bug severity and bug priority.

Bug severity refers to the level of impact that the bug has on the application

or system while bug priority refers to the level of urgency in the need for a

fix.

25. What is the difference between system testing and integration testing?

For system testing, the entire system as a whole is checked, whereas for

integration testing, the interaction between the individual modules are tested.

26. Explain the term bug.

A bug is an error found while running a program. Bug fall into two categories:

logical and syntax.

Senior Tester Interview Questions

27. Explain the difference between functional and structural testing.

Functional testing is considered to be behavioral or black box testing in which

the tester verifies that the system or application functions according to

specification. Structural testing on the other hand is based on the code or

algorithms and is considered to be white box testing.

28. Define defect density.

Defect density is the total number of defects per lines of code.

29. When is a test considered to be successful?

The purpose of testing is to ensure that the application operates according to

the requirements and to discover as many errors and bugs as possible. This

means that tests that cover more functionality and expose more errors are

considered to be the most successful.

30. What good bug tracking systems have you used?

This is a simple interview question about your experience with bug tracking.

Provide the system/systems that you are most familiar with if any at all. It

would also be good to provide a comparison of the pros and cons of several if

you have experience.

31. In which phase should testing begin – requirements, planning, design, or

coding?

Testing should begin as early as the requirements phase.

32. Can you test a program and find 100% of the errors?

It is impossible to fine all errors in an application mostly because there is no

way to calculate how many errors exist. There are many factors involved in such

a calculation such as the complexity of the program, the experience of the

programmer, and so on.

33. What is the difference between debugging and testing?

The main difference between debugging and testing is that debugging is typically

conducted by a developer who also fixes errors during the debugging phase.

Testing on the other hand, finds errors rather than fixes them. When a tester

finds a bug, they usually report it so that a developer can fix it

34. How should testing be conducted?

Testing should be conducted based on the technical requirements of the

application.

35. What is considered to be a good test?

Testing that covers most of the functionality of an object or system is

considered to be a good test.

36. What is the difference between top-down and bottom-up testing?

Top-Down testing begins with the system and works its way down to the unit

level. Bottom-up testing checks in the opposite direction, unit level to

interface to overall system. Both have value but bottom-up testing usually aids

in discovering defects earlier in the development cycle, when the cost to fix

errors is lower.

37. Explain how to develop a test plan and a test case.

A test plan consists of a set of test cases. Test cases are developed based on

requirement and design documents for the application or system. Once these

documents are thoroughly reviewed, the test cases that will make up the test

plan can be created.

38. What is the role of quality assurance in a product development lifecycle?

Quality assurance should be involved very early on in the development life cycle

so that they can have a better understanding of the system and create sufficient

test cases. However, QA should be separated from the development team so that

the team is not able to build influence on the QA engineers.

39. What is the average size of executables that you have created?

This is a simple interview question about our experience with executables. If

you know the size of any that you’ve created, simply provide this info.

40. What version of the Oracle are you familiar with?

This is an interview question about experience. Simply provide the versions of

the software that you have experience with.

41. How is an SQL query executed in Oracle 8?

This is an interview question to check your experience with Oracle and you can

simply provide the answer “from the command prompt.” If you do not have Oracle

experience, do not pretend and simply state that you have not worked on an

Oracle database.

42. Have you performed tests on the front-end and the back-end?

This is an interview question in which you should explain whether you performed

testing on the GUI or the server portion of previous applications.

43. What is the most difficult problem you’ve found during testing?

This is a simple interview question in which you should provide an example

44. What were your testing responsibilities at your previous employer?

This interview question is very likely being asked to verify your knowledge of

your resume. Make sure that you know what is on your resume and that it is the

truth.

45. What do you like the most about testing?

There are several answers that you can give for this question. Here are a few

examples:

You enjoy the process of hunting down bugs

Your experience and background have been focused on enhancing testing techniques

You like being in the last phase of work before the product reaches the customer

You consider your contribution to the whole development process to be very

important.

1. What should be done if the actual result does not match with

expected result?

If actual test results do not match the expected results, then it must be logged

as a defect. Further testing should be performed to find as many details as

possible about this defect so it can be logged into the defect tracking system.

2. Why is requirements traceability important during product testing?

Requirements traceability is important during product testing because it allows

the testers to determine if adequate test coverage has been achieved.

3. When is it most appropriate to perform system testing?

When all of the components of a system have been integrated it is then

appropriate to perform testing on the system as a whole to verify that it meets

all requirements and quality standards.

4. What is a use case and how does it differ from a test case?

A use case describes the behavior of a system as it acts in response to

requests. Use cases describe which users are allowed to take which actions

whereas a test case is a set of variables and conditions that are used to

determine whether the application or system is working as designed.

5. Explain the difference between a test script and a test case.

A test case is a set of variables and conditions that are used to determine

whether the application or system is working as designed and a test script is

the actual set of instructions that are performed to determine the functionality

of a system. So a test case describes what needs to be done and a test script

tells exactly how to accomplish it.

6. What are the elements of a defect report?

The basic elements of a defect report are the following:

1. Defect ID

2. Test Case

3. Author

4. Date

5. Build (where defect was found)

6. Description of problem

7. Severity

8. Priority

9. Status

10. Assigned

7. What is the best way to proceed with testing if minimal or no product

documentation is available?

If little to know product documentation is available, assumptions must be made

about the product and these assumptions must be clearly stated in the testing

documents. Also, if this is necessary to perform tests without knowledge of the

product being tested, it may be good to research similar products to find

potential testing requirements.

8. How do you determine when and adequate amount of testing has been performed?

One can determine that an adequate amount of testing has been performed when all

product requirements has been covered under a test case.

9. How does automation fit into the overall testing process?

Automation is sometimes part of the testing process because automated tests are

often capable of performing actions at a rate and duration that humans are

incapable of. This may improve test coverage and increase the amount of testing

that can be performed in a shorter amount of time.

10. Give a description of the Software Development Life Cycle.

The Software Development Life Cycle (SDLC) is the process used to develop

software. Though it may vary from company to company, the SDLC typically has

the following phases:

1. Planning

2. Requirements Analysis

3. Design and Development

4. Integration and Test

5. Implementation

6. Maintenance

11. If a company’s primary product is software, what is the role of QA in that

company?

The role of QA is to work with the development team to ensure the quality of the

product. This includes verifying quality of deliverables throughout the

project. Also, QA is heavily involved in testing and typically works under a

process that is separate from the development process.

12. Explain white page testing.

White page-testing checks the dimensions, canvas size, window-resizing, and page

and line scrolling etc. match the requirements documents.

Q16: What is automated testing?

: In automated testing a test algorithm or program is written which runs on computer to automatically test any software for bugs or other similar things.

Q17: How do you view manual testing in comparison to automated testing?

Although automated testing is much faster way of testing but it also has certain disadvantages as compared to manual testing as a lot of hard work is required to         design the testing algorithm or program in case of automated testing but this struggle is not required in case of manual testing. Also in certain graphical user interface based applications where the interface changes continuously it is very difficult to use automated testing as automated testing procedure is based on certain predefined algorithm and this algorithm may not be able to test the application for such frequent changes.

Q18: What should be done after finding a bug in software testing procedure?

After finding a bug, the immediate step must be to inform the software development team about the bug with all the relevant information such as bug name, ID, location,         severity and causes, so that immediate action is possible to fix the bug.

Q19: How a client/server based application is tested?

Usually it is very difficult to test client/server based applications due to the complexities involved such as data communications, multiple clients, hardware and           multiple servers. So testing requirements may be very extensive in this case. Also a           lot of time is consumed in such testing.

Q20: What are the factors that affect the effectiveness of manual testing?

Like many other processes, there are several factors that affect the effectiveness of manual testing. These include documentation thoroughness and test case         comprehensiveness. The main objective of tester should be to optimize the         management’s comprehension of the manual testing by allocating the proper         resources in every area, of course, considering the overall resource limitations.

Q21: How to do manual testing if time span is short?

If the time span is short then the tester must first carefully test those specific parts of application that are much liable to get bugs. Also management is necessary in case of         short time span so a tester must try to carry out multiple functionality tests or the tests that best cover the risky areas in available time span.

Q22: When should the testing procedure be stopped?

Although several applications today are so complex that it is not possible to test them completely by manual testing procedures but still there are some factors on the basis of which it is decided when to stop testing. These factors include completion of test cases, completion of deadline, fall of bug rate below certain level, ending of budget and many more similar factors.

Q23: How hidden or unexpected functionalities must be dealt in testing?

It is very difficult to determine whether application has some unexpected or hidden functionality or not. But if some unexpected or hidden functionality is found during          testing procedure then it should be immediately removed from code if it is not linked to any other parts of the code. Because these unexpected or hidden functionalities make the testing procedure lengthy and difficult, so they must be removed immediately.

Q24: What do you mean by bug leak or defect leak?

When an application is tested & released testing team, the client will do Beta testing, and while doing this testing, if client finds any bug then that bug is called Bug leak or Defect leak.

Q25: What is the difference between QA and QC?

QC actually is related to product and it detects defects of product while QA is related to process and it is based on defect prevention.

Q11: Name several stages of manual testing?

There are several stages involved in manual testing. These are listed below:-

1. Unit testing

2. Integration testing

3. System testing

4. User acceptance testing

Q. 18: What is the difference between Statement Coverage, Branch Coverage and Path Coverage?

"Statement Coverage" is a type of "White-Box Testing" technique, involving execution of all statements at least once. Statement coverage is a simple metric to calculate & measure the number of statements in a method or class which have been executed. Its key benefit is its ability to identify which blocks of code have not been executed.

"Branch Coverage" is an outcome of a decision, and measures the number of decision outcomes or branches, which have been tested. This takes a more in-depth view of the source code rather than a simple "Statement Coverage". A branch is an outcome of a decision. For example Boolean decisions like an "If - Statement", has two outcomes or branches (i.e. True and False).

Whereas "Path Coverage" is a method of testing which satisfies the coverage criteria through which the program is tested across each logical path. Usually, paths through the program are grouped into a finite set of classes and one path out of every class is tested. In Path Coverage flow of execution takes place from the start of a method to its exit. Path Coverage ensures that we test all decision outcomes independently of one another.

Q. 19: What is the difference between Ad-hoc Testing, Monkey Testing and Exploratory Testing?

"Ad-hoc Testing" is performed without any planning of process and without any documentation like Test Case or Test Scenarios. It involves test design and simultaneous test execution. For Ad-hoc testing the testers possess significant understanding of the software before testing it.

"Monkey Testing" is done with no specific test in mind. Here the monkey is the producer of any input data (which can be either a file data or can be an input device data). It involves pressing some keys randomly and checking whether the software fails or not.

Whereas "Exploratory Testing" involves simultaneous learning, test design and test execution. It is a type of "Ad-hoc Testing", but only difference is that in this case, the tester does not have much idea about the application & he explores the system in an attempt to learn the application and simultaneously test it.

Q. 20: What is the difference between System Testing and End-to-End Testing or E2E Testing?

"System Testing" falls within the scope of Black-Box testing and the tester requires no knowledge of the inner design of the code or logic. It is conducted on a complete / combined part of a system to verify that all-functional, information, structural and quality requirements as per the specifications have been met.

"End-to-End Testing" or "E2E Testing" is also quite similar to "System Testing". It involves testing of the application in a environment that simulates the real-world use, such as interacting with a database, using network communications, or interacting with other hardware, applications, or systems.

Q. 39: What are the Testing Types & Techniques?

Black Box and White Box are the most popular types of software testing. These are not the stand-alone testing techniques.

Testing techniques falling under the Black-Box type are: 1) Equivalence Partitioning 2) Boundary Value Analysis 3) Cause-Effect Graphing 4) Error-Guessing etc.

Whereas testing techniques falling under the White-Box type are:

1) Statement coverage

2) Decision coverage

3) Condition coverage

4) Decision-condition coverage

5) Multiple condition coverage

6) Basis Path Testing

7) Loop testing

8) Data flow testing etc.

Q. 55: What do we lose with Automation compared to Manual Testing?

Creating an automated test is usually more time-consuming & costly than running it once manually. The cost differential varies, depending on the product and the automation style.

1) If the product is being tested through a GUI and your automation style is to write scripts that drive the GUI, an automated test may be several times as expensive as a manual test.

2) If you use a GUI capture / replay tool that tracks your interactions with the product and builds a script from them, automation is relatively cheaper. It is not as cheap as manual testing, though, when you consider the cost of recapturing a test from the beginning after you make a mistake, the time spent organizing and documenting all the files that make up the test suite, the aggravation of finding and working around bugs in the tool, and so forth. Those small "in the noise" costs can add up surprisingly quickly.

3) If you’re testing a compiler, automation might be only a little more expensive than manual testing, because most of the effort will go into writing test programs for the compiler to compile. Those programs have to be written whether or not they’re saved for reuse.

Q. 57: What is the difference between code coverange analysis & test coverage analysis?

Both these terms are similar. Code coverage analysis is sometimes called test coverage analysis. The academic world generally uses the term "test coverage" whereas the practitioners use the term "code coverage".

Q. 82: What is the role of QA in a software producing company?

QA is responsible for managing, implementing, maintaining and continuously improving the Processes in the Company and enable internal projects towards process maturity and facilitate process improvements and innovations in the organization.

Tester is responsible for carrying out the testing efforts in the company.

In many companies QA person is responsible both the roles of Testing as well as creating and improving the processes.

Q. 92: What is Measure of Completeness in software testing?

In software testing there are two measures of completeness, code coverage and path coverage.

Code coverage is a white box testing technique to determine how much of a program's source code has been tested. There are several fronts on which code coverage is measured. Code coverage provides a final layer of testing because it searches for the errors that were missed by the other test cases.

Whereas Path coverage establishes whether every potential route through a segment of code has been executed and tested.