Top 50 Manual Testing Interview Questions & Answers

### Q-1. What Is Requirement Traceability Matrix?

**Answer.**

Requirement Traceability Matrix (RTM) is a document which records the mapping between the high-level requirements and the test cases in the form of a table.

That’s how it ensures that the Test Plan covers all the requirements and links to their latest version.

### Q-2. Explain The Difference Between Pilot And Beta Testing?

**Answer.** Read the following points to know the difference between Pilot and Beta testing.

**1.** We do the beta test when the product is about to release to the customer whereas pilot testing takes place in the earlier phase of the development cycle.  
**2.** In the beta test, testing application is given to few users to make sure that application meet the customer requirements and does not contain any showstopper bug. Whereas, in the pilot test, few members of the testing team work at the Customer site to set up the product. They give their feedback also to improve the quality of the end product.

### Q-3. Describe How To Perform Risk Analysis During Software Testing?

**Answer.** Risk analysis is the process of identifying the hidden issues that may derail the successful delivery of the application. It also prioritizes the sequence of resolving the identified risks for testing purpose.

Following are some of the risks that are of concern to the QA.

**1.** New Hardware.  
**2.** New Technology.  
**3.** New Automation Tool.  
**4.** The sequence of code delivery.  
**5.** Availability of test resources for the application.

We prioritize them into three categories which are as follows.

**1.** High magnitude: Impact of the bug on the other functionality of the application.  
**2.** Medium: it is tolerable in the application but not desirable.  
**3.** Low: it is tolerable. This type of risk has no impact on the company business.

### Q-4. What Is Silk Test And Why Should You Use It?

**Answer.** Here are some facts about the Silk tool.

**1.** It’s a tool developed for performing the regression and functionality testing of the application.  
**2.** It benefits when we are testing Window based, Java, the web, and the traditional client/server applications.  
**3.** Silk Test help in preparing the test plan and managing them to provide the direct accessing of the database and validation of the field.

### Q-5. What Is The Difference Between Master Test Plan And Test Plan?

**Answer.** The difference between Master Plan and Test Plan can be described using following points.

**1.** Master Test Plan contains all the test scenarios and risks prone areas of the application. Whereas, Test Plan document contains test cases corresponding to test scenarios.  
**2.** Master Test Plan captures each and every test to be run during the overall development of application whereas test plan describes the scope, approach, resources and schedule of performing the test.  
**3.** MTP includes test scenarios to be executed in all the phases of testing that run during the complete life cycle of the application development. Whereas, a separate Test Plan exists for each phase of testing like Unit, Functional, and System which contains the test cases related to that type only.  
**4.** Only for big projects, we need a Master Test Plan which requires execution in all phases of testing. However, preparing a basic Test Plan is enough for small projects.

### Q-6. How Do You Handle A Non-Reproducible Bug?

**Answer.** Following bugs lie under the non-reproducible category.

**1.** Defects observed due to low memory issue.  
**2.** Bugs raised due to address pointing to a memory location that does not exist.  
**3.** The race condition is an error scenario which occurs when the timing of one event impacts another executing in a sequence.

A tester can take the following actions to handle the non-reproducible bugs.

**1.** Execute test steps that are close to the error description.  
**2.** Evaluate the test environment.  
**3.** Examine and evaluate test execution results.  
**4.** Keep the resources & time constraints under check.

### Q-7. How Do You Perform Automated Testing In Your Environment?

**Answer.** Automation Testing is a process of executing tests automatically. It reduces the human intervention to a great extent. We use different test automation tools like QTP, Selenium, and WinRunner. These tools help in speeding up the testing tasks.

Using the above tools we can create test scripts to verify the application automatically. After completing the test execution, these tools also generate the test reports.

### Q-8. What Are The Factors That You’ll Consider To Choose Automated Testing Over Manual Testing?

**Answer.**

The choice of automated testing over manual testing depends on the following factors.

**1.** Tests require periodic execution.  
**2.** Tests include repetitive steps.  
**3.** Tests execute in a standard runtime environment.  
**4.** Automation is expected to take less time.  
**5.** Automation is increasing reusability.  
**6.** Automation reports are available for every execution.  
**7.** Small releases like service packs which include a minor bug fix. In such cases, regression type of cases is sufficient for validation.

### Q-9. What Is The Difference Between A Test Driver And Test Stub?

**Answer.**

The **test driver** is a piece of code that calls a software component under test. It is useful in testing that follows the bottom-up approach.

**Test stub** is a dummy program that integrates with an application to complete its functionality. These are relevant for testing that uses the top-down approach.

Let’s take an example.

**1.** Let’s say there is a scenario to test the interface between modules A and B. We have developed only module-A. Then we can test module-A only if we have real module-B or a dummy module for it. In this case, we call module-B as the Test Stub.  
**2.** Now, module-B can’t send or receive data directly from module-A. In such scenario, we’ve to move data from one module to another using some external features called Test Driver.

### Q-10. What Are The Essential Qualities Of An Experienced QA Or Test Lead?

**Answer.** Every QA or Test Lead should have the following qualities.

**1.** Well-versed in Software testing processes.  
**2.** Ability to accelerate teamwork to increase productivity.  
**3.** Improve coordination between QA and Dev engineers.  
**4.** Provide ideas to refine the QA processes.  
**5.** Ability to conduct RCA meetings and draw conclusions.  
**6.** Excellent written and interpersonal communication skills.  
**7.**Quick learner and able to groom the team members.

### Q-11. What Are The Different Types Of Software Testing?

**Answer.** Following is the list of various testing types used by manual testers.

* Unit testing
* Integration testing
* Regression testing
* Shakeout testing
* Smoke testing
* Functional testing
* Performance testing
  + Load testing
  + stress testing
  + Endurance testing
* White box and Black box testing
* Alpha and Beta testing
* System testing

### Q-12. What Are The Key Elements Of A Test Plan?

**Answer.** A test plan contains the following main points.

* Testing objectives.
* Test scope.
* Testing the frame.
* The environment
* Reason for testing
* The criteria for entrance and exit
* Deliverables
* Risk factors

### Q-13. What Is A Test Case?

**Answer.**

A test case is a sequence of actions and observations that are used to verify the desired functionality. A good test case helps to identify problems in the requirements or design of an application.

### Q-14. What Is Agile Testing And Why Is It Important?

**Answer.** Agile testing is a software testing process which evaluates a software from the customer point of view. And it is important because this does not require Dev to complete coding for starting QA. Instead, the coding and testing both goes hand in hand. However, it may require continuous customer interaction.

### Q-15. How Do You Test A Product If The Requirements Are Yet To Freeze?

**Answer.** If the requirement spec is not available for a product, then a test plan can be created based on the assumptions made about the product. But we should get all assumptions well documented in the test plan.

### Q-16. How Will You Tell If Enough Test Cases Have Been Created To Test A Product?

**Answer.** First of all, we’ll check if every requirement has at least one test case covered. If yes, then we can say that there are enough test cases to test the product.

### Q-17. What Will You Do When A Bug Turns Up During Testing?

**Answer.** When a bug shows up, we can follow the below steps.

* Run more tests to make sure that the problem has a clear description.
* Run a few more tests to ensure that the same problem doesn’t exist with different inputs.
* Once we are sure of the full scope of the bug, then we can add details and report it.

### Q-18. If A Product Is In Production And One Of Its Modules Gets Updated, Then Is It Necessary To Retest?

**Answer.** It is advisable to perform regression testing and run tests for all of the other modules as well. Finally, the QA should carry out the System testing.

### Q-19. What Is The Difference Between Functional Requirement And Non-Functional Requirement?

**Answer.** The functional requirement specifies how a product should run whereas a non-functional requirement represents how it should be.

#### Functional Requirements.

* Authentication
* Business rules
* Historical Data
* Legal and Regulatory Requirements
* External Interfaces

#### Non-Functional Requirements.

* Performance
* Reliability
* Security
* Recovery
* Data Integrity
* Usability

### Q-20. How Comes The Severity And Priority Relate To Each Other?

**Answer.**

* **Severity –** Represents the gravity/depth of the bug.
* **Priority –** Specifies which bug should get fixed first.
* **Severity –** Describes the application point of view.
* **Priority –** Defines the user’s point of view.

### Q-21. What Are Different Types Of Severity?

**Answer.**The severity of a bug can be low, medium or high depending on the context.

* User Interface Defect – Low
* Boundary Related Defects – Medium
* Error Handling Defects – Medium
* Calculation Defects – High
* Misinterpreted Data – High
* Hardware Failures – High
* Compatibility Issues – High
* Control Flow Defects – High
* Load Conditions (Memory leakages under load testing) – High

### Q-22. What Is Entry And Exit Criteria In Software Testing?

**Answer.**

**Entry criteria –** It is a process that should run when a system begins. It includes the following artifacts.

* SRS (Software Requirement Specification)
* FRS (Functional Requirement Specification)
* Use case
* Test-Case
* Test-plan

**Exit Criteria –** It signals when the testing should complete and when should the product be ready to release. It includes the following artifacts.

* Test Summary Report
* Metrics
* Defect Analysis report

### Q-23. What Is Test Strategy?

**Answer.** Test strategy is an approach to carry out the testing activity. It covers the following.

* Roles and responsibilities for each member.
* Testing scope.
* Test tools.
* Test environment.
* Testing schedule.
* Associated risks.

### Q-24. What Is Smoke Testing And What Is Sanity?

**Answer.**

**Smoke testing** confirms the basic functionality works for a product. It requires you to identify the most basic test cases for execution.

**Sanity testing,** on the other hand, ensures that the product runs without any logical errors. For example, if we are testing a calculator app; we may multiply a number by 3 and check whether the sum of the digits of the answer is divisible by 3.

### Q-25. What Is The Difference Between A Bug, Defect, And Error?

**Answer.** A bug is usually same as the defect. Both of them represents an unexpected behavior of the software.

However, an error would also fall in the same category. But in some cases, errors are fixed values. For example – 404/405 errors in HTML pages.

### Q-26. What Is The Difference Between High Level And Low-Level Test Case?

**Answer.**

* High-level test cases cover the core functionality of a product like standard business flows.
* Low-level test cases are those related to user interface (UI) in the application.

### Q-27. What Is The Difference Between Static Testing And Dynamic Testing?

**Answer.**

#### Static Testing.

* It is a white box testing technique which directs the developers to verify their code with the help of checklist to find errors in it.
* Developers can start it done without actually finalizing the application or program.
* Static testing is more cost effective than Dynamic testing.
* It covers more areas than Dynamic testing in a shorter time.

#### Dynamic Testing.

* Dynamic testing involves the execution of an actual application with valid inputs and checking of the expected output.
* Examples of Dynamic testing are Unit Testing, Integration Testing, System Testing and Acceptance Testing.
* Dynamic testing happens after the code deployment.
* It starts during the validation stage.

### Q-28. What Is Test Harness?

**Answer.**

Test Harness requires configuring a set of tools and input data to test an application under various conditions. It involves monitoring the actual output with expected output for correctness.

Its benefits are as follows.

* Upward push in productivity due to process automation.
* Improve the overall product Quality.

### Q-29. What Is Defect Leakage?

**Answer.**

Defect leakage occurs at the Customer or the End-user side after the product delivery. If the end user sees any issue in the application, then such bugs lead to Defect leakage. And this process of finding bugs is also called as Bug Leakage.

### Q-30. What Kind Of Document Will You Need To Begin Functional Testing?

**Answer.**

* It is none other than the Functional specification document. It defines the full functionality of a product.
* Other documents are also useful in testing like user manual and BRS.
* Gap analysis is another document which can help in understanding the expected and existing system.

### Q-31. Beside Test Case & Test Plan, What Documents A Tester Should Produce?

**Answer.** Here are a few other documents to prepare.

* Testing metrics
* Test design specs
* End-to-end scenarios
* Test summary reports
* Bug reports

### Q-32. What Is Business Requirements Document (BRD)?

**Answer.** BRD provides a detailed business solution for a project including the documentation of customer needs and expectations.

BRD fulfills the following objectives.

* Gain agreement with stakeholders.
* Provide clarity on the business requirements.
* Describe the solution that meets the customer/business needs.
* Determine the input for the next phase of the project.

### Q-33. What Is Risk Analysis?

**Answer.** Risk analysis is a technique to identify the things that can go wrong in a software development project. They can negatively impact the scope, quality, timeliness, and cost of a project.

However, everyone involved in the project has a part in minimizing the risk. But it’s the lead who ensures that whole team understands the individual role in managing the risk.

### Q-34. What Is Exploratory Testing?

**Answer.**

Exploratory testing is a process which lets a tester to concentrate more on execution and less on planning.

* It requires formulating a test charter, a short declaration of the scope, set of objectives and possible approaches to be used.
* The test design and test execution activities may run in parallel without formally documenting the test conditions, test cases or test scripts.
* Testers can use boundary value analysis to concentrate the testing effort on error-prone areas by accurately pinpointing the boundaries.
* Notes should be recorded for the Exploratory Testing sessions as it would help to create a final report of its execution.

### Q-35. Can We Do System Testing At Any Stage?

**Answer.** No. The system testing should start only if all modules arc in place and work correctly. However, it should happen before the UAT (User Acceptance testing).

### Q-36. Why Is It Impossible To Test A Program Completely?

**Answer.**

Here are the two principal reasons that make it impossible to test a program entirely.

* Software specifications can be subjective and can lead to different interpretations.
* A software program may require too many inputs, too many outputs, and too many path combinations to test.

### Q-37. What Is The Primary Difference Between Debugging & Testing?

**Answer.**

* Testing is to find out defects while using a product whereas debugging is to reach the part of the code causing failure.
* Debugging is isolating the problem area in the code done by a developer whereas Testing is identifying the bug in an application and done by a tester.

### Q-38. What Are The Roles Of Glass-Box And Black-Box Testing Tools?

**Answer.**

#### Black-Box Testing.

It doesn’t require the knowledge of internal design or code. So the tests are based on requirements and functionality. Black box testing focuses on finding the following errors.

* Interface errors
* Performance errors
* Initialization errors
* Incorrect or missing functionality
* Errors in accessing external database

#### Glass-Box Testing Or White-Box Testing.

It requires familiarity with the internal design and application code. So the tests concentrate on path coverage, branch coverage, and statement coverage. It is expected to cover the following.

* All possible code flows of a module.
* Execute all loops.
* Exercise all logical decisions.
* Verify internal data structure to ensure their validity.

### Q-39. What Is GAP Analysis?

**Answer.** Gap analysis reveals any deviation between the features available for testing and how the customer perceives them to be.

Traceability matrix is a testing tool which testers can use to track down the gaps.

### Q-40. How Do We Know The Code Has Met Specifications?

**Answer.** Traceability matrix is an intuitive tool which ensures the requirements mapped to the test cases. And when the execution of all test cases finishes with success, it indicates that the code has met the requirements.

### Q-41. Is It Possible To Achieve 100% Coverage Of Testing? How Would You Ensure It?

**Answer.** No, it’s not possible to perform 100% testing of any product. But you can follow the below steps to come closer.

* Set a hard limit on the following factors.
  + Percentage of test cases to be passed.
  + The no. of bug found.
* Set a red flag if,
  + Test budget depleted.
  + Deadlines breached.
* Set a green flag if,
  + The entire functionality gets covered in test cases.
  + All critical & high bugs must have a status of CLOSED.

### Q-42. What Are Error Guessing And Error Seeding?

**Answer.**

#### Error Guessing.

It is a test case design technique in which testers have to guess the defects that might occur and write test cases to represent them.

#### Error Seeding.

It is the process of adding known bugs in a program for the tracking the rate of detection & removal. It also helps to estimate the number of faults remaining in the program.

### Q-43. What Is The Difference Between Coupling And Cohesion?

**Answer.** The difference between coupling and cohesion is as follows.

* Cohesion is the degree which measures the dependency of the software component that combines related functionality into a single unit whereas coupling represents the binding of related functionality into a different unit.
* Cohesion deals with the functionality that relates to different process within a single module whereas coupling deals with how much one module is dependent on the other modules within the product.
* It is a good practice to increase the cohesion between the software whereas coupling is discouraged.

### Q-44. What Is CMM?

**Answer.** The Capability Maturity Model for Software (CMM or SW-CMM) is a model for assessing the maturity of the software processes of an organization and for identifying the key practices that increase the maturity of these processes.

### Q-45. What Is Cause Effect Graph?

**Answer.** It is a graphical representation of inputs and the associated outputs effects which assist in designing test cases.

### Q-46. Why Does Software Have Bugs?

**Answer.**

* Miscommunication.
* Programming errors.
* Timeline pressures.
* Change in requirements.
* Software complexity.

### Q-47. What Is Ramp Testing?

**Answer.** It is a testing method which proposes to raise an input signal until the system breaks down.

### Q-48. What Is Recovery Testing?

**Answer.** It ensures that the program must recover from any expected or unexpected events without loss of data or functionality.

Events could be like shortage of disk space, unexpected loss of communication, or power out conditions.

### Q-49. What Is Inspection?

**Answer.** It’s a group review quality improvement process for the product documents. It focuses on the following two aspects.

* Product document improvement.
* Process improvement (of both document production and inspection).

### Q-50. What Is Globalization Testing?

**Answer.** Globalization testing concentrates on detecting the potential problems in the product design that could spoil globalization. It certifies that the code can handle the desired international support without breaking any functionality. And also, it ensures that there would be no data loss and display problems.