75 Manual Testing Interview Questions and answers

Q 1: What are the 2 main types of testing?

Ans: The 2 main types of testing are:

1. Manual Testing: where the testing activities are performed by testers without the help of any tools.
2. Automation Testing: where the testing is carried out with the help of tools or stand-alone scripts that can replicate the manual operations performed.

Q 2: What is BVA?

Ans: BVA, Boundary Value Analysis, is a validation technique used for checking the edit boxes that take in numbers. For example, if a field takes the values 1 to 100, under BVA you would test the field with values +1 and -1 from the boundaries. So, in this case, the tester would check to see if the field with values 0,2,99, and 101.

Q 3: What is DOA and how is it used?

Ans: DOA, Dead on Arrival, is used to refer to applications build that are unsuitable for testing. There can be several situations where a build is declared as DOA. Some of these are:

1. Smoke Test fails
2. Application is unstable and crashes intermittently
3. The functionalities mentioned in the release notes are not included in the build or are not available for testing

Q 4: What is the Severity and Priority of a bug?

Ans: The severity and priority are important aspects of a bug. The Severity is decided by the tester based on the impact on the testing and the functionality of the application. For example, an app crash or user unable to login, back end server down, etc would all be high severity defects.

Priority on the other hand is decided by the developer based on how soon they can fix it or how much time would be needed to fix the defects. Some in most cases high severity and high priority bugs would be the same. But there can be some cases where the Severity is low but priority is high. Like a spelling mistake in the company name on the home page. Now, an example of high severity but a low priority can be unable to open a link in a web page, when the link is rarely used by the customer.

Q 5: What is a traceability matrix? Why is it important?

Ans: Traceability matrix is the link between a client requirement or use case and the test case. While writing a test case, a tester should ensure there is at least 1 test case for every use case of functional requirement. This mapping between the requirements and test cases is called traceability.

The traceability matrix helps to ensure that each requirement in the functional document is considered for testing, thus avoiding any misses.

Q 6: What are the different levels of testing?

Ans: Testing is done at different levels, they are

1. Unit testing or component level testing
2. Functional and module testing
3. Integration testing
4. System testing
5. Acceptance testing

Q 7: What is a test plan?

Ans: A test plan is a document that enlists all the aspects of the testing project. Some of the main headers under a test plan are:

1. In scope and out of scope features.
2. The hardware, software, and resource requirements
3. Entry and exit criteria
4. Test strategy
5. Test deliverables
6. Risks and contingencies
7. Roles and responsibilities
8. Escalation mechanism

Q 8: What is the difference between regression and retesting?

Ans: Regression testing refers to repeated testing of the functionalities already delivered and tested in previous testing cycles. Regression testing is done to ensure that the new code changes in the current build have not broken or adversely impacted any previous functionalities. In many companies, regression testing is automated to ensure more time can be dedicated to the thorough testing of the new features.

Retesting, on the other hand, refers to rechecking the defects after they are fixed, to ensure the issue is completed resolved. For retesting, a tester would perform the same steps as mentioned in the defect steps to reproduce to ensure the defect is fixed and then based on their discretion they would also check areas that might get affected by the defect fix.

Q 9: What is a defect?

Ans: Any deviation from the expected behavior of the application can be termed as a defect.

Q 10: What are the different stages in a defect life cycle?

Ans: The different stages in a defect life cycle are:

1. New
2. Assigned
3. Fixed
4. Moved to QA / Ready to test
5. Retested
6. Closed
7. Reopen
8. Duplicate
9. Redundant

10. Moved to next iteration / Deferred

Q 11: What is a latent defect?

Ans: There are some defects in the system that may not cause failures or impact the customer usage of the application because the preconditions for these defects are rarely met. Such defects that are silently present in the system are known as latent defects.

Q 12: What are the most common risks associated with project failure?

Ans: The most common risks are:

1. Shortage of time or squeezed timelines
2. Shortage of skilled resources to perform the testing activities
3. Limited Budget
4. Absence of a stable environment for testing

Q 13: What are stubs and drivers?

Ans: Stubs and drivers are used in incremental testing. Testing follows either a top-down or a bottom-up approach. In the top-down approach, a stub is used to help in the testing of other modules. Similarly, a driver is used in the bottom-up approach for facilitating the testing.

Q 14: What would you do if you find a defect?

Ans: Here is what you should do once you find a defect.

1. Recreate the defect to make sure it is consistent
2. Try to recreate the defect in a different device, browser, or OS to make sure if it occurring everywhere or is specific to any particular device, browser, or OS.
3. Take the screenshot of the error
4. Capture the required logs as applicable
5. Log the details in the bug tracking tool used by the team

Q 15: What is fuzz testing?

Ans: Fuzz testing is commonly used in security testing to find vulnerabilities and coding errors in the system. In fuzz testing, data is added to the system to make it crash. Sometimes, a fuzz tester is also used to find the causes of the vulnerability as well. It is mostly done in big projects and may not prove cost-effective for smaller projects.

Q 16: What are the test cases you can create for a simple login screen with username, password, and login buttons?

Ans: Some of the test cases that can be created for the login screen are:

1. Test cases for login edit box, what type of characters it can take.
2. Test cases for password edit box, what type of characters it can take.
3. Password should be masked and should not be visible
4. Valid credentials should allow users to log in and move to the next screen
5. Invalid credentials should block the user, the appropriate message should be displayed
6. Username blank – appropriate message should be displayed
7. Password blank – appropriate message should be displayed
8. The edit boxes should be aligned properly
9. A forgot password link should be available

10. Appropriate labels should be written for the login and password edit boxes

Q 17: What is the difference between a latent and masked defect?

Ans: Latent defects are those that are existing in the system but do not impact the functionality since the preconditions for the defects are never met. A masked defect, on the other hand, is those which are hidden behind another defect. And they can be found only once the other defect is found.

Q 18: What techniques can be used to understand the test coverage?

Ans: The 3 different techniques used for ensuring the test coverage are:

1. <u>Statement coverage</u>: testing ensures that each line of the code is executed at least once.
2. <u>Decision coverage</u>: testing is done to ensure that each path of the decision code loops and decision boxes are executed at least once.
3. <u>Path coverage:</u> testing is done to ensure every possible code path for a particular functionality is executed at least once.

Q 19: What is a code walk-through?

Ans: A code walkthrough is an informal analysis of any code to check if the logic is correct and if the proper coding standards are followed. It is an unmoderated review.

Q 20: What are the different parts of a defect report?

Ans: A defect report would ideally include:

1. Summary
2. Tested By
3. Assigned to
4. Tested environment
5. Build No.
6. Severity
7. Steps to reproduce
8. Expected and Actual Results
9. Screenshots
10. Logs
11. Defect status
12. Comments

Q 21: What is an exit criterion and why is it used?

Ans: Exit criteria list the conditions in which the testing would be stopped at any point in the project. It is very useful during the project execution as a guide to reject the testing build.

Some of the main exit criteria are:

1. Build Failure
2. Login failure
3. App Crash in critical functionalities
4. No data in the application
5. Failed smoke test

Q 22: What is alpha testing?

Ans: Alpha testing is the testing done by a subset of potential users or clients in the developer's site with the development infrastructure.

Q 23: What is beta testing?

Ans: Beta testing is the testing done by a subset of the potential users or clients at their locations which is generally different from the environment where the product was developed.

Q 24: What is crowdsourced testing?

Ans: Crowdsourced testing is a new and upcoming form of testing that involves people, not necessarily testers, from across the globe to test the product and provide their feedback. These people get paid to perform the task.

Q 25: What is a smoke test?

Ans: A smoke test is the first test that is done to ensure the shared build is ready for testing and the critical functionalities are working. In many companies, the build is accepted for further testing only if the smoke test is a pass.

Q 26: How does testing fit into a CI pipeline?

Ans: In a CI pipeline all the tasks are automated, hence testing is also automated. A trigger is set to start the testing suite every time there is a new code check-in and build. Once the testing is complete a pass/fail report is sent out to the stakeholders and based on the results the decision is taken to push the build to production or to roll it back.

Q 27: How is testing possible in the agile/scrum sprints?

Ans: In Agile and Scrum the testing would start early along with the development. As and when a block of code is ready, testing starts for that functionality with the help of stubs, drivers, and harness.

Q 28: What are positive and negative testing?

Ans: Positive testing refers to all the happy paths where the application performs as expected with the correct input. Negative tests, on the other hand, involves giving incorrect data to the system to ensure it responds appropriately.

Q 29: What is an exploratory test?

Ans: Random tests that are done without any pre-decided steps to just traverse through the application and find defects are called exploratory testing.

Q 30: What is DRE?

Ans: DRE, defect removal efficiency, a metrics commonly used for evaluating the effectiveness of the testing process. It is defined as the percentage of the defect found in testing by the sum of defects in testing and production.

Q 31: What is the career path for a tester?

Ans: Just like a developer there is a great career path for testers. It starts with a tester or junior tester, test analyst, test lead, test manager, and test consultant.

Q 32: What are the roles and responsibilities of a tester?

Ans: A tester is expected to

1. Take care of the day to day testing activities
2. Report bugs when found
3. Retest the bugs

Q 33: What are the roles and responsibilities of a test lead?

Ans: A test lead would

1. Assign the tasks to the team members
2. Monitor the defects raised by the team
3. Participate in defect triages when required.
4. Consolidate and share the daily or weekly testing status with the manager
5. Help the manager while creating a test plan
6. Inform the manager of any foreseen risks

Q 34: What are the roles and responsibilities of a test manager?

Ans: A test manager is responsible for

1. Getting the testing projects from clients and stakeholders
2. Preparing and presenting the strategy document
3. Preparing the test plan with the help of a lead
4. Help the team with the resources needed
5. Help the team with mitigating the risk
6. Prepare and present the testing metrics to the team
7. Help in escalations as and when needed

Q 35: What are the different types of testing. Name a few types in each.

Ans: Testing can be divided mainly into 2 types:

1. **Functional Testing**: It deals with testing the functionality of the application under test. Some of the testings under functional testing are:
   a. Unit Testing
   b. System testing
   c. User Acceptance testing
   d. Integration testing
2. **Non-functional testing**: it involves testing the non-functional features like performance, scalability, security reliability, etc.
   a. Load testing
   b. Stress testing
   c. User Acceptance testing
   d. Security testing
   e. Endurance testing
   f. Installation testing

Q 36: What are the stages in the software testing lifecycle?

Ans: The stages in the software testing lifecycle are:

1. Requirement Analysis
2. Test Planning
3. Test Case Development
4. Test Environment Setup
5. Testing

6. Test Closure and deliverables handover

Q 37: What are the qualities of a good tester?

Ans: Some qualities of a good tester are:

1. An eye for finding details and mistakes in the system
2. A good understanding of the application under test and how the users would be using the application
3. A good domain knowledge
4. Good communication skills are important for a tester
5. Good contacts within and outside the team help the tester reach out to people and get issues resolved easily speeding up the process.

Q 38: What is cross-browser testing?

Ans: Cross-browser testing refers to testing a web application on different browsers like Chrome, Firefox, Edge, Safari, etc. to ensure it performs equally well in each.

Q 39: What is responsive testing?

Ans: Responsive testing involves testing a website or mobile app on devices of different screen sizes to ensure it renders properly in each device. This testing also helps to eliminate any alignment issues during the rendering.

Q 40: What is a software bug?

Ans: Any deviation from the expected functionality of the application as documented in the specifications document is called a software bug.

Q 41: What is black box testing?

Ans: A type of testing where the tester understands and has access only to the input and output of a system through a graphics interface is called black-box testing. Here the tester would not know the code or how it is written.

Q 42: What is white box testing:

Ans: In white-box testing, the tester has access to the code implementation. The test cases are designed to cover the different paths and loops in the implementation. For white-box testing, some basic coding knowledge and background are an additional advantage.

Q 43: What are the advantages of manual testing?

Ans: The advantages of manual testing are:

1. It is more suitable for small projects
2. A tester can find more bugs when performing test cases unlike automation
3. The test cases can be easily tuned to adjust to the changes in the code and UI
4. Difficult scenarios and switching between systems and modules can be easily done
5. Very useful in ad-hoc, exploratory, and usability testing
6. Needs fewer skills and can be learned easily

Q 44: What are the disadvantages of manual testing?

Ans: Some disadvantages of manual testing are:

1. It is time-consuming and not suitable for projects with strict timelines
2. It can become redundant for functionalities that are stable with no code change
3. Performance tests can not be performed manually
4. It proves more expensive in the long run.
5. With every iteration or release, the timelines keep on increasing.
6. It is prone to human errors

Q 45: When will you choose manual testing over automation?

Ans: Under some situations, manual testing is preferred over automation, these are:

1. Bug re-testing
2. For testing cases, that need to be run only a few times for the project.
3. For Adhoc and exploratory testing.
4. For checking the user-friendliness of the application
5. While handling complex scenarios.
6. For testing cases, where some random decision-making is involved.

Q 46: What is a testbed?

Ans: The environment for testing with the needed parameters is called a testbed. It is a digital platform that can include the OS, hardware, software, database, network, etc.

Q 47: What is a test case?

Ans: A test case is a set of instructions that the tester is expected to perform to test functionality. A good test case can be easily understood and run by any person even if they do not have a testing background.

Q 48: What are the details included in a test case?

Ans: A test case would include:

1. S No.
2. Traceability reference to the requirements document
3. Pre-conditions
4. Steps to reproduce
5. Expected result
6. Actual result
7. Result (Pass/Fail)
8. Bug ID
9. Comments

Q 49: What is user acceptance testing?

Ans: Also known as end-user testing, it is performed by a subset of the actual users to understand if the application is working as expected for the users. In UAT, the users would suggest some improvements in the application that would make it more suitable.

Q 50: What is Soak Testing?

Ans: Soak testing is a kind of performance testing in which the system is kept under continuous load for a long time to find defects and bottlenecks in the system. It is also called as Endurance test. Most commonly memory leaks and system utilization-related issues can be found during these tests.

Q 51: What is localization testing?

Ans: The localization testing is done to check the functionalities that the dependent on the location. These can be related to map, search features, regional languages, etc.

Q 52: What is A/B testing?

Ans: A/B testing is done when there are 2 or more versions of the app. Both these apps are given to the client or each app is given to a set of clients, for testing to decide which one is better and should be deployed to production.

Q 53: What are the different types of integration testing?

Ans: The different types of integration testing are:

1. Big bang testing
2. Bottom-up testing
3. Top-down testing

Q 54: What is called the pesticides paradox?

Ans: When the same tests are conducted on the same system, again and again, no new defects can be found and the system becomes immune to these tests. This situation is called the pesticides paradox. To avoid such a scenario, the tests should be modified with new strategies and approaches.

Q 55: Name some test deliverables?

Ans: Some test deliverables are:

1. Test Plan
2. Effort estimation report
3. Test cases
4. Traceability matrix
5. Defect report
6. Test report
7. Test summary

Q 56: What is big-bang testing?

Ans: Testing all the modules together once the development is complete is called the big-bang testing approach. The defects found in big-bang testing have a very high fixing cost since they are found towards the end of the development cycle.

Q 57: What is top-down testing?

Ans: In the top-down testing approach, the top modules are tested first and then moving on to the smaller modules. Stubs are used as a temporary module to assist in testing the higher modules.

Q 58: What is bottom-up testing?

Ans: Bottom-up approach is the opposite of the top-down approach, here the lower modules are tested first as and when they are ready. Moving up to bigger integrated modules. Drivers are used to doing integration testing till the module development is complete.

Q 59: What is end-to-end testing?

Ans: End-to-end testing means testing the entire flow of the application from the start to the end. One of the biggest challenges in end-to-end testing is that it spans multiple systems, so the verification can become complex.

Q 60: What is GUI testing?

Ans: Testing an application using only the graphical user interface is called GUI testing. In this, apart from the functionality, a tester would also check the alignment, color combinations, readability, ease of navigation, etc.

Q 61: What is recovery testing?

Ans: This testing is done to check how soon the system can recover after a system crash. It is a type of non-functional testing.

Q 62: What is risk-based testing?

Ans: In risk-based testing, the critical modules and those which are likely to have more failures are tested first. This type of test prioritization based on risk is the basis of risk-based testing.

Q 63: What is security testing?

Ans: The testing that is done to find the vulnerabilities in the system, protect the data, and ensure the functionalities are working as expected is called security testing. It also involves checking the access to systems. This means people with different access levels should be able to access only parts of the system they are allowed to as per the business.

Q 64: What is load testing?

Ans: Load testing is done to see the performance of the application under load. A load is simulated on the system and then functional testing is performed to find the bottlenecks in the system.

Q 65: Name some best practices in software testing.

Ans: Some best practices are:

1. Start early
2. Update your test cases, and test approach frequently to identify more defects
3. Always think like the client
4. Think out of the box
5. Automate repeated tests and focus more testing on new modules

Q 66: What is exhaustive testing?

Ans: Exhaustive testing is testing the system with all the valid and invalid inputs. In most cases, exhaustive testing is not possible and a tester would go with BVA or equivalence partition techniques to test the application.

Q 67: What is the 80:20 rule?

Ans: The 80:20 rule states that 80% of the defects are found in 20% of the code. This is also called defect clustering.

Q 68: What is defect cascading?

Ans: The phenomenon where one defect if left undetected, triggers multiple defects in other modules and later stages are called defect cascading. The cost of fixing this defect cascading can be high, due to the greater number of defects in the system.

Q 69: What is outsourced testing?

Ans: Assigning the testing task to a third party or separate team within or outside the company is called outsourced testing. This is done to ensure free and unbiased testing. Sometimes it is also helpful in giving a fresh perspective to the testing activity.

Q 70: What is bug leakage?

Ans: A bug that is missed by the testing team during the testing phase and escapes undetected to production, where it is discovered by the actual user is called leaked bug and the phenomenon is called bug leakage.

Q 71: What is a show stopper defect?

Ans: A defect that does not allow the testing activities to proceed is called a show stopper defect. App crash on launch and login failures are examples of show stopper defects.

Q 72: What is a hotfix?

Ans: Sometimes a build that is deployed to production would have some critical error and it would be rolled back. Now an emergency fix for the issue is released as a new patch and this build is called a hotfix.

Q 73: What are the different models of SDLC?

Ans: The main models are:

1. Waterfall model
2. V model
3. Spiral model
4. Agile

Q 74: Name some challenges in software testing?

Ans: Some challenges in software testing

1. Insufficient documentation or unfrozen requirements.
2. Unavailability of skilled testing resources
3. Stringent timelines
4. Unstable systems and incomplete modules are given for testing
5.

Q 75: What is the age of a defect?

Ans: The age of the defect is defined as the number of days between the day the defect is accepted or assigned by the development team and the day it is verified and closed by the testing team. The age of the defect can be defined in hours and days.