1. When do you start automating the project?

During regression testing we can do automation and it’s good to automate only the features which are stable. For the features which keep on changing or get updated for every release, it is good to test manually.

1. How to deal with defects that are NOT reproducible?

* Verify the steps used to produce the error
* Verify the system/environment used to produce the error
* Gather screenshots and logs
* Gather step-by-step description from the user
* Try alternate approaches to produce the error

1. What is difference between use case and test case?

* Use case is an overview of the software functions to reach the end goal from business perspective
* Test case is a document written based on functional document which contains the test data, preconditions, expected results and post conditions developed from different test scenarios.

1. Common challenges faced in software testing?

* Testing the complete application – There are millions of test combinations. It’s not possible to test each and every combination both in manual as well as in automation testing. If you try all these combinations you will never ship the product.
* Misunderstanding of company processes – Sometimes we don’t pay attention to company defined processes and these are for what purposes. Sometimes testers have to go with company processes even these processes are not applicable for their current testing scenario. This results in incomplete and inappropriate application testing.
* Regression testing – When project goes on expanding the regression testing work simply becomes uncontrolled. Pressure to handle the current functionality changes, previous working functionality checks and bug tracking.
* Lack of skilled testers – Sometimes due to wrong management decision, while selecting and training testers for their project task results in unskilled testers, because of whom testing work becomes more chaos than simplifying. As a result, we have to face incomplete, insufficient and ad-hoc testing throughout the testing life cycle.
* Which tests to execute first – If all of sudden, test manager came and ask if we are ready for completion by this week? This order from boss, makes testers just to focus on task completion, rather than full test coverage or quality of work and testers don’t understand or decide which test cases to execute first or automate first and this effects on the test coverage and quality of work.
* Decision to stop the testing: It is very difficult to take the decision on when to stop the testing. Requires core judgement of testing processes and importance of each process. Also requires ‘on the fly’ decision ability.
* Reuse of test scripts: Application development methods are changing rapidly, making it difficult to manage the test tools and test scripts. Test scripts migration or reuse is very essential but difficult task.
* Understanding the requirements: Sometimes testers are responsible for communicating with customers for understanding the requirements. If tester fails to understand the requirements, will he be able to test the application properly? Definitely no! Testers require good listening and understanding capabilities.
* Relationship with developers: To maintain a good relation and handle that is a very big challenge. There are hundreds of simple excuses developers or testers can make when they are not agree with some points. For this testers also require good communication, troubleshooting and analysing skills.

These are some top testing challenges we face daily. Project success or failure depends largely on how you address these basic issues.

1. How to resolve inconsistent defect?

Inconsistent bugs are those which are not stable, that means bug appears sometimes not frequently. It occurs once in one of the scenario, if we try to reproduce that in the same way it is unable to reproduce. So for this type of bugs, if that bug effects the functionality of the product, definitely we need to focus on it, attach the log while raising that inconsistent defect along with the screenshots. But if the defect doesn’t affect anywhere, then can just leave it.

1. What is RTM? What does RTM contain?

* Req. tracing Matrix is a process of documenting the links between req. & the work products developed to implement and verify those requirements.
* RTM captures all the req. and their traceability in a single document delivered at the end of the life cycle.
* Matric created at the beginning of the project, as it forms the basis of the project’s scope and the deliverables.
* Matrix is bidirectional
  + Req - > o/p of the deliverables (Forward)
  + Particular feature -> business requirement (Backward)

RTM Parameters

1. Req ID
2. Risks
3. Req. Type
4. Req. Description
5. Trace to Design Specification
6. Unit Test Cases
7. Integration Test Cases
8. System Test Cases
9. User Acceptance Test Cases
10. Trace to Test script.
11. Differences between Alpha testing and beta testing?

According to ISTQB material both the alpha and beta testing is to get feedback from the existing or potential customers before the commercial deploying or before the sale of software.

* Alpha testing is **performed** at the developing organizations site but **not by the developing team**.
* Beta or field testing is performed by customers / potential customers at their own location.

1. Explain test strategy?

The test strategy is a living document that is created in the Project’s Requirements Definition phase, after the requirements have been specified. The test strategy document describes the scope, approach, resources and schedule for the testing activities of the project. This includes defining what will be tested, who will perform testing, how testing will be managed, and the associated risks and contingencies. The test strategy document is maintained throughout the life of a project.

1. Why do you need test plan and what does test plan contain?

Test plan is the project plan for the testing work to be done. Like the famous saying “if you fail to plan then you plan to fail”. **It is not** a test design specification, a collection of test cases or a set of test procedure. **IEEE 829 STANDARD TEST PLAN TEMPLATE** contains:

1. Test plan identifier
2. Test deliverables
3. Introduction
4. Test tasks
5. Test items
6. Environmental needs
7. Features to be tested
8. Responsibilities
9. Features not to be tested
10. Staffing and training needs
11. Approach Schedule
12. Item Pass/Fail criteria
13. Risks and contingencies
14. Suspension and resumption criteria Approvals
15. What is the process for creating a manual test script

As mentioned earlier test case and test scripts are interchangeably used and is same.

So you have any idea of the process of creating a manual test script?

It’s pretty easy – first thing is to start writing test condition, get it approved from Test Lead and start writing your test case. Test case is a document written based on functional document.

1. What are the fields in a bug report?

Defect ID

Description

Version

Feature

Test Case title

Steps to reproduce

Severity

Priority

Status

Reviewer Name

Data Detected

Developer Assigned

Suggested Fix

1. How to overcome the challenge of not having input documentation for testing?

There will be situations where you don’t have BRD or SRS to get enough information about the requirement/module. So as a tester we should have discussions with the BA to get an understanding about the feature or the requirement.

I will explain this with the recent project that I am working on, in which there is no documentation available. So the first thing that we did is explore the application as much as possible and cleared the process/ function with the BA. So depending on our understanding of the module, we started creating test conditions and test scripts.

1. Difference between QA and QC?

QA

* Process oriented
* Focus on defect prevention

QC

* Product Oriented
* Focus on defect identification.

1. What is Gray box testing?

Gray box testing is a technique to test the software product or application with partial knowledge of the internal workings of an application.

It is a combination of both White Box testing and Black Box testing method.

Gray box testing gives the ability to test both sides of an application, presentation layer as well as the code part. It is primarily useful in Integration Testing and Penetration Testing.

**Example of Gray box testing:**  While testing websites feature like links or orphan links, if tester encounters any problem with these links, then he can make the changes straightaway in HTML code and can check in real time.

**Why Gray Box testing:**

1. It provides combined benefits of black box testing and white box testing both
2. It combines the input of developers as well as testers and improves overall product quality
3. It reduces the overhead of long process of testing functional and non-functional types
4. It gives enough free time for developer to fix defects
5. Testing is done from the user point of view rather than designer point of view
6. Difference between System Testing and System Integration Testing?

System Testing is concerned with the behaviour of a whole system/product. The mechanism of testing a completely integrated system to verify/ensure that it meets the defined means specified requirement. The main focus of system testing is verification against the specified requirement. It includes both the functional and non-functional testing.

System Integration testing tests the interactions between different systems and may be done after system testing. I had worked an application (AA) in which the Road Side Assistance system is integrated with the CARDS system which is already running in the organization. Our team had done integration testing in that project.

1. What are test documents involved in testing?