

## QA Process Assignment Questions

- 1) The major difference between QA and QC lies in their objective. Quality Assurance is proactive processes which focuses on making the processes so that, the errors will be minimum however Quality Control is a reactive process which occurs after the production to identify and control any bugs or defects in the code.
- 2) The role of a QA engineer is to make sure that the processes involve in creation of the product and the product itself is devoid of any defects and meets the quality standards. They participate in all phases like Requirements, Design, Development , Testing and Deployment Phase.
- 3) Testing is of two types, functional testing and non functional testing. Functional testing focuses on what the system does, verifying that the software behaves as expected i.e. it does what it was supposed to do. Non functional testing focuses on how system does it. It focuses on the speed, efficiency, latency and user experience etc.
- 4) A Test Plan is a document that outlines the strategy, objectives, scope, resources, schedule and deliverables of the testing process. It acts as a blueprint for testing, ensuring that everyone involved understands what will be tested and by whom.

### Test Plan Outline

Test Plan ID : TP-LOGIN-001

Objectives : Ensure login functions are expected, only valid users can access the system, identify and report security issues, validate UI/UX elements.

Scope : Input field validation, successful and unsuccessful login elements, UI/UX of login form, error messages, remember me functionality, password masking, login button behaviour, basic security checks.

Test Strategy : Functional Testing, UI Testing, Boundary Testing, Negative Testing, Cross Browser Testing and Security Testing (basic)

Test Environment : Web application deployed on staging server

Resources : QA Engineers (1-2 testers), Chrome, TestRail, Selenium, Developer Support, QA Lead, Project Manager.

- 5) Test coverage is a metric that measures how much of your code, features and requirements are tested by your test cases. It ensures that critical parts are not being left. Test coverage can be increased by, Starting early, writing comprehensive test cases, using code coverage tools, automating regression and critical tests and including unit, integration and system tests.

**6) Test Strategy** is a high level, comprehensive document that outlines the overall approach to testing for an organization or a specific project. It defines what and how to test, what tools and tech to be used, how will bugs be reported and resolved and what will be the exit criteria for testing.

The major difference between the both is that Test Plan is a project specific document which is based on test strategy but dives into execution level details.

Following things can be added to a test strategy

1. Test Objectives
2. Testing Scope
3. Test Types
4. Test Design Techniques
5. Test Tools
6. Test Environment
7. Test Data Strategy
8. Entry and Exit Criteria
9. Defect Management
10. Risk and Mitigation

**7) A Test case** is a detailed set of steps, inputs, expected results, and conditions used to verify that a software feature works as intended. It answers why are you testing, how will you test it and what should happen.

#### Positive Test Cases

Test Scenario	Input Data	Expected Result
Valid Registration	Username: john123 Email: <a href="mailto:john@mail.com">john@mail.com</a> Password: John@123 Confirm: John@123	Registration successful Message; user created
Password with special characters	Password : John#123!	Accepted; user registered successfully

#### Negative Test Cases

Test Scenario	Input Data	Expected Result
Empty Required fields	All fields blank	Show validation error for all required fields
Invalid email format	Email: johnmail.com	Show “invalid email format” error

**8) Components of test case are:**

1. Test Case ID
2. Test Title
3. Description
4. Preconditions
5. Test Steps
6. Test Data
7. Expected Result
8. Actual Result
9. Status
10. Remarks

Components	Details
Test Case ID	TC_FP_001
Test Title	Verify Forgot Password with valid email
Test Description	Ensure user receives a reset link when entering valid email
Preconditions	User is registered and knows their email
Test Steps	<ol style="list-style-type: none"><li>1. Navigate to login page</li><li>2. Click "Forgot Password" link</li><li>3. Enter registered email</li><li>4. Click "Submit"/"Send Link"</li></ol>
Test Data	Email : user@example.com
Expected Result	A success message appears
Actual Result	[Filled during test execution]
Status	Pass / Fail

**9) Boundary Value Analysis (BVA)** is a black box test design technique used to identify errors at the boundaries of input ranges, where bugs are most likely to occur.

For Age Input

Input Age	Expected Result	Remarks
17	Invalid (below minimum)	Boundary -1

18	Valid	Minimum boundary
19	Valid	Minimum + 1
59	Valid	Maximum - 1
60	Valid	Maximum Boundary
61	Invalid	Boundary + 1

**10)** White box testing is type of testing, which is done within the actual code itself. Eg. Unit Testing. Black box testing is a type of testing which verifies system's working without actual knowledge of the internal codebase. Eg. Boundary Value Analysis

**11)** Regression Testing is the process of retesting an application after changes (like code updates, bug fixes or enhancements) to ensure the existing functionality still works as expected.

**When RT is necessary ?** When updating password validation rules, say your old development team has rule of min 6 characters while the new rule is of min 8 characters, 1 uppercase, 1 symbol. Regression tests will ensure that the user registration still works and existing users can log in.

**12)** UAT (User Acceptance Testing) is the final phase of software testing, where the end users or clients validate whether the system meets their business requirements and is ready for production. Functional testing is the testing of working of the actual code and is done mostly by the developers.

**13)** Exploratory testing is an informal, handson testing approach where testers actively explore the application without predefined test cases to learn how it behaves, find bugs creatively and evaluate it's usability and flow.

For a new feature, say "New Search Feature in an E-commerce App"

1. Understand features briefly
2. Set a time box for exploration
3. Create a Test Charter
4. Explore with Intent (valid input, typos, empty input, special characters, long input string, filtering + sorting, repeated rapid searches, backspacing and re-entering quickly).

**14)** Defect is any flaw, error or unintended behaviour in a software application that causes it to produce incorrect or unexpected results.

#### **Typical Defect States**

- 1) New :- Defect is newly reported by a tester.
- 2) Assigned :- Defect is assigned to a developer
- 3) Open :- Developer has acknowledged and started working on it.

- 4) Fixed :- Developer has fixed the defect
- 5) Closed :- Issue is resolved and no longer active

**15)** Severity means how serious the defect is in terms of system functionality or user experience. Priority is how soon the defect should be fixed.

The major difference is that severity focuses on impact on functionality while priority focuses on urgency of fixing the issue.

**16)** I will perform the following steps, when I find a critical bug

- 1. Document the Bug Propely using JIRA, Bugzilla or Azure DevOps
- 2. Escalate the Issue by informing your lead, notifying developer assigned, updating the bug tracker status.
- 3. Join or Request a Bug Triage Meeting.
- 4. Retest Once Fixed

**17)** Purpose of automated testing tool is execute test cases without manual intervention and improve testing speed, accuracy and coverage. Two of the most popular automated testing tools are Selenium and JUnit.

**18)** Selenium is a popular open-source automated testing tools for web applications which allows testers and developers to automate browser actions like clicking buttons, entering text, navigating pages and verifying page content.

### **Python script to check login functionality**

```
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Keys
import time

driver = webdriver.Chrome()

driver.get("https://example.com/login")

driver.find_element(By.ID, "username").send_keys("testuser")
driver.find_element(By.ID,
"password").send_keys("password123")

driver.find_element(By.ID, "loginBtn").click()

time.sleep(2)
```

```
if "Dashboard" in driver.title:
    print("Login successful!")
else:
    print("Login Failed!")

driver.quit()
```

**19)** Continuous Integration (CI) is a development practice where developers frequently integrate code changes into a shared repository, usually multiple times a day. Each integration is automatically built and tested to catch issues early. Continuous Testing (CT) is the practice of executing automated tests continuously as part of the software delivery pipeline, from development to deployment. Its purpose is to ensure code quality, performance and functionality.

**20)** Performance Testing is a type of non functional testing that evaluates how a software application behaves under various conditions like heavy usage, limited system resources, or extended periods of operation. The goal is to ensure the system is fast, stable, reliable and scalable.

**Load Testing** tests the behaviour of system under expected load.

**Stress Testing** pushes the system beyond limits to see how it breaks or recovers.

**21)** Load Testing helps determine how your web application performs under expected user load such as 100, 500 or 1000 users access it at the same time.

The steps are

1. Define Objectives like homepage performance, login under load and checkout process.
2. Select the Right tool like Apache JMeter, k6, LoadRunner.
3. Prepare the Test Scenarios
4. Set up the test environment
5. Execute the Test
6. Analyze Results
7. Identify Bottlenecks and Optimize
8. Re-Test After Fixes

**22)** Security Testing is a type of software testing that ensures an application is free from vulnerabilities, unauthorized access and data breaches. Its goal is to verify that the system protects data and maintains functionality as intended, even when under attack.

Eg. SQL Injection, Cross-Site Scripting, Authentication Bypass

**23)** In manual testing, tester executes test cases by hand, step by step where as in automated testing, tests are executed by scripts or tools automatically.

Manual testing is used over automated testing in tasks like testing look and feel, exploratory or Ad-hoc testing, dynamic user interactions etc

**24)** A test report is generated in most cases. It's a formal document that summarizes the outcomes of testing activities.

It's components are :-

- 1) Test Summary
- 2) Objectives & Scope
- 3) Test Environment
- 4) Test Case Execution
- 5) Defects / Bugs Found

**25)** A TSR, is a high level document that summarizes the testing activities, results and conclusions of a testing phase or full project. It serves as a final quality checkpoint to help stakeholders decide whether the product is ready to release.

**26)** Agile is a software development methodology focusing on delivering working software incrementally and iteratively, with continuous collaboration between teams and quick responses to change. Agile fundamentally changes how Quality Assurance is integrated into the project as the QA happens continuously after every short sprint.

**27)** Test Driven Development (TDD) is a software development practice where tests are written before writing the actual code. It works on following cycle

1. Red -> Write a failing test case for a new feature or functionality.
2. Green -> Write just enough code to make the test pass.
3. Refactor -> Clean up the code while ensuring the test still passes.

With TDD, QA is involved early in the development process, the code quality improves, collaboration improves, and focus is shifted to higher-level testing.

**28)** In Agile, testing occurs simulatenously with the development, it's not a separate phase. QA engineers are part of the cross functional team and testing activities begin on Day 1 of the sprint. The role of QA in sprint planning will be to review user stories, define acceptance criteria, estimate testing efforts and plan test coverage.

**29)** Some commonly used QA metrics that help assess the effectiveness and health of testing process are

1. Defect Density :- Number of defects per size of the code, helps identify which modules are more error-prone and need more testing or refactoring.
2. Test Coverage :- Percentage of code, features or requirements covered by tests

3. Test Execution Rate :- Measures how many test cases were executed in a given time.
4. Defect Leakage :- Number of defects found in production after release vs those found during testing
5. Defect Removal Efficiency (DRE) :- Measures how effectively QA catches bugs before release.

**30)** Root Cause Analysis (RCA) is a systematic process in QA to identify the underlying reason why a defect or problem occurred, not just its symptoms. It's purpose is to fix the actual cause of defects so they don't happen again, improving long-term product quality and process efficiency.

**31)** To ensure testing process is efficient and valuable, QA teams uses key performance indicators (KPIs), measurable metrics that track how well testing is identifying issues, reducing risk and contributing to high quality releases. Some KPI are Defect Detetion Rate, Defect Leakage Rate, Test Case execution Rate, Test Pass Rate, Automation Coverage, Average Time to detect and fix defects, Requirement coverage, customer reported defects, escaped defects trend etc.

**32)** Risk Based Testing is a testing approach that prioritizes test cases based on level of risk associated with features or components of the software. In simple terms, you test the most critical, risky or likely to fail parts first and less risky ones later or less frequently.

**33)**

Risk	Impact	Probablity	Risk Level	Risk Mitigation
Inoccrect order processing	High	Medium	High	Implement end-t0-end automated test cases for order flows.
Payment failures or double charges	High	Low	High	Integrate robust payment gateway error handling.
Unauthorized purchase	High	Medium	High	Perform thorough security testing (OWASP), token validation.

**34)** Cross Browser testing is the process of verifying that a web application works correctly and consistently across different browsers, browsers versions and devices.



Ensures that users have a consistent experience regardless of what browser or device they use.

Tools for Cross Browser testing are, BrowserStack, SauceLabs, Selenium Grid etc.

**35)** Mobile Testing is the process of testing mobile applications, both native apps, hybrid apps and mobile web apps to ensure they work correctly across different devices, OS versions, screen sizes and network conditions. It includes functional, performance, UI/UX security, and compatibility testing specific to mobile environments.