TESTING THE DOCUMENTATION

Documentation testing is part of the non-functional testing of a product. It may be a type of black-box testing that ensures that documentation about how to use the system matches with what the system does, providing proof that system changes and improvements have been documented.

Test documentation is a document that contains the Specification of a software project. It includes a Test plan, Test Case, Test Scenario, and Traceability Matrix.

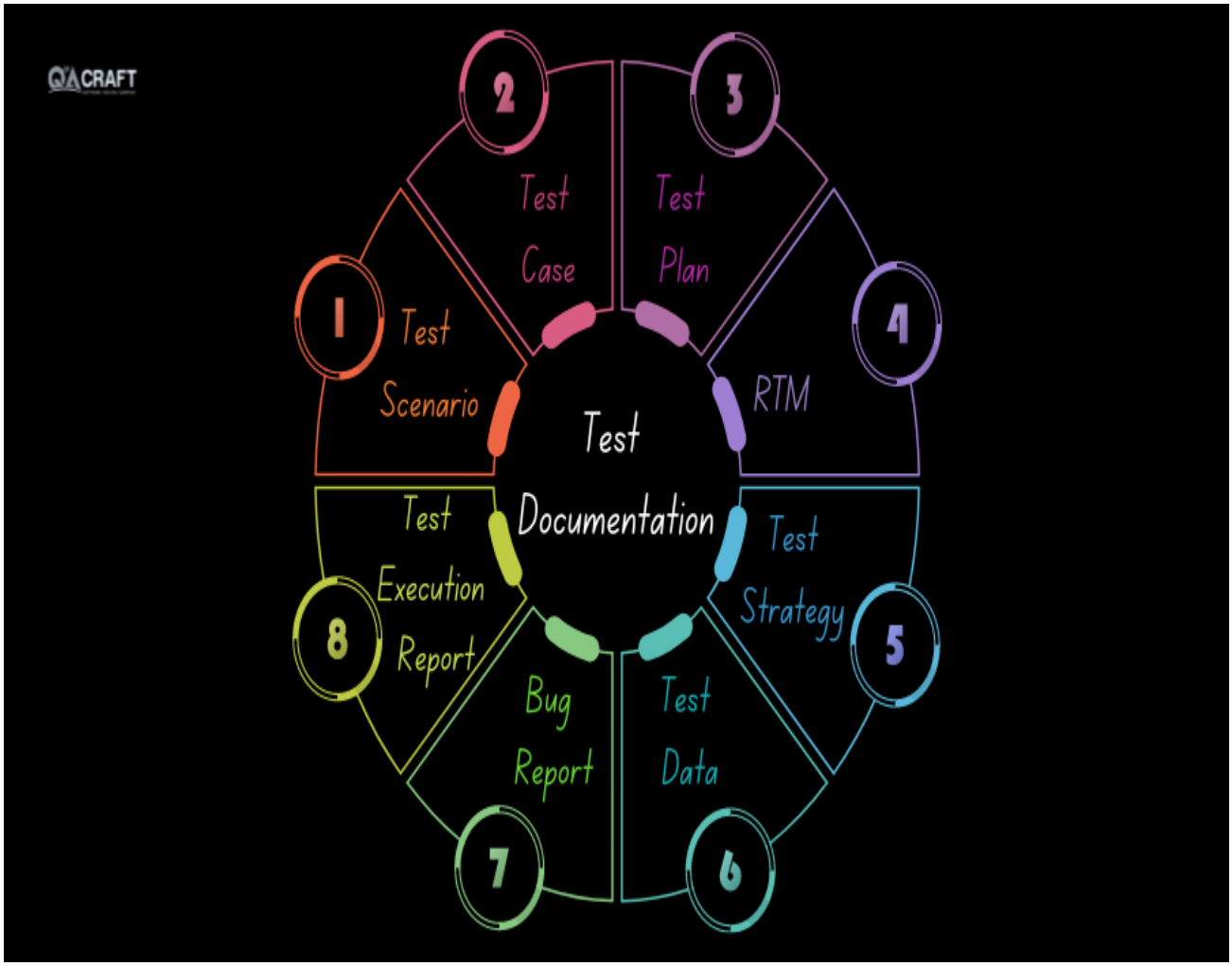
Any project needs to have good test documentation. Good quality documentation reduces the cost, time, and effort of an organization.

Why Test Formality?

We can say that we are living in a digital era. We use mobile applications for primary requirements like Electricity bill payment, G-Pay, Paytm, and online shopping. so this thing needs to be tested properly then only it can be useable.

So after the development of the software, the next phase is to test formality to make sure that the particular project is working fine as per specification.

EXAMPLES OF TEST DOCUMENTATION



Test Scenario: This document describes the various scenarios of the application to test. It helps to understand the flow of the application. It does not consist of any input data and navigation steps.

Test Case: A test case is a very descriptive test document. It consists of input data and navigation steps to test data. It also contains expected results and actual results for a clear understanding of the application's working status.

Test Plan: A test plan is a test document prepared by the test lead or manager. The test plan consists of all activities for testing an application. There are multiple components of a Test plan (Test Environments, Test methodology, Template, Role and responsibility, Effort estimation, Objectives, Scope, Approach, Entry

and Exit criteria, Schedule, Tools, Defect tracking, Test Deliverable, Assumption, Risk, and Mitigation Plan or Contingency Plan.)

Requirement Traceability Matrix(RTM): RTM stands for Requirement traceability matrix. RTM makes sure that all the test cases have been covered. RTM is used to ensure that we have not missed any particular requirement or test cases to execute in an application.

Test strategy: Test strategy Is a very important document for the testing process. It describes what kind of technique has been used for testing particular applications and which module going to be tested. It consists of multiple components test processes, scope, customer communication strategy, documentation formats, and objectives.

Test Data: Test data is a document that is created before test execution starts. Test data is used while implementing test cases. Test data is an Excel document prepared manually.

Bug Report: The "Bug Report" is a document that consists of details of all the bugs that occurred during the testing procedure. It is a very important document for developers and the QA team because by referencing this document team can track the process of a project. Using this document team came to know the number of open issues, solved issues, worked under process issues, Rejected issues, and solved issues.

Test Execution Report: The test lead prepares the "Test Execution Report" after the test execution process is completed. It defines the constancy of the project. It has information like the module, the number of written test cases, the number of passed test cases, number of failed test cases.

Best Practice to Achieve Test Documentation

- Involve the QA team from the beginning of the project so that testing documents can be prepared in parallel.
- Regularly update the test document as the testing procedure is proceeding.
- Verify the test document after the testing process is completed.

ADVANTAGES OF TEST DOCUMENTATION

- Test document is important to reduce or remove uncertainties about the testing procedure.
- Test document is helpful to follow one structure for testing an application and it is helpful to fresher for training purposes.
- A mature test document is important to prepare a bug-free product within the specified time limit.
- Using the test document QA team becomes very clear about the required configuration of the system for testing any particular application.
- It helps to establish a transparent relationship between the client and the team.

DISADVANTAGES OF TEST DOCUMENTATION

- Sometimes it is a time-consuming task for the QA team to prepare test documents.
- Sometimes it is written by people who are not good writers so it becomes difficult to understand the content by other team members.
- Sometimes it is difficult to make up-to-date test documents as per the client's requirements.
- If the quality of the document is not good then it creates a misunderstanding between the client and the team.

SECURITY TESTING

Security testing is an important aspect of software testing focused on identifying and addressing security vulnerabilities in a software application. It aims to ensure that the software is secure from malicious attacks, unauthorized access, and data breaches.

Security testing helps minimize data corruption risks and ensures cybersecurity in the workplace. Many business aspects can benefit from security testing, including departmental databases and website codes. Learning more about security testing can help you address vulnerabilities in a company's technology systems.

Security testing is a process that checks whether company software is vulnerable to cyberattacks. Software for security testing evaluates what effects malicious software has on databases and websites. Security tests help ensure that only allowed inputs enter a system, keeping it safe from cyberattacks.

All security testing programs are functional testers, meaning it verifies whether other software systems work according to their specifications and only reacts when it senses an anomaly. It can also identify potential security risks, allowing businesses to prevent them effectively.

TYPES OF SECURITY TESTING

There are many types of security testing, which include:

PENETRATION TESTING

Penetration testing or ethical hacking is the process of manually testing a software's security protocols. Software developers and programmers use penetration testing to make sure that all security firewalls in place work according to pre-determined standards. They may also use penetration testing to test a website's usability. For example, developers can log into the maximum amount of user accounts available on a website to see whether its servers can handle them.