Software Quality Assurance

What is software quality assurance?

Software quality assurance is a practice where software is tested to ensure the functionality, security, and performance are working as expected (as it is documented).

This is important because abnormal behavior of software can cost organizations much of it's capital. Let alone it costs more to fix issues in production.

Like any practice, there is a process to testing and resources needed to keep track of the testing efforts.

The Notions of Testing

Black Box testing: The user is testing the functionality of the software without knowledge of the internal code or structure. Example: Testing what functions appear in the user interface like dropdown menus or login page.

White box testing: The user is testing the functionality of the application and is fully aware of the code and internal structure. Doing unit testing or integration testing.

Functional testing

Smoke testing: Testing the major functionality of the application. Particularly features that are frequently used and depend heavily on the application. For example login verification or checking if payment has been confirmed for online shopping.

Functionality testing: Testing all functionalities of the application.

System testing: testing the system as a whole functionally and nonfunctional. As in features and performance.

Regression testing: This form of testing is retesting the application after any code changes occurred to ensure there are no abnormalities.

End-to-end testing: Checks an entire software application from beginning to end, mimicking real user interactions and data.

User Acceptance Testing: Is when stakeholders, clients, or UAT team tests the functionality of the application before signing it off and deploying it to production.

Non-functional Testing

Performance testing: This is to see how the application performs such as checking the response time. Is it fast or is it slow?

Load testing: Tests apply the expected load to see how the system performs during normal or above-average usage.

Stress testing: Tests apply extreme load to determine the point at which your system breaks due to high traffic and heavy usage.

Security testing: Identifying and addressing security vulnerabilities in a software application.

There are other forms of testing however these are basic concepts to know when doing tests. Many of these concepts, especially end to end testing, are applicable to Salesforce when testing out processes.

If you are a Salesforce professional configuring and building out solutions on the instance. Chances are you may have done both black box and white box testing.