

BeatBlip



APPROACHING SOFTWARE TEST AUTOMATION HOW TO SELECT TEST CASES?

Software test automation saves a lot of time and costs when done right. Users are cognizant of the importance of automation, but their challenge is to identify the ideal processes to automate to maximize the ROI.

WHAT ARE TEST CASES?

A test case is a detailed document shared by the business users regarding the Application Under Test (AUT). It contains a list of specifications, testing conditions, inputs, and expected outcomes.

WHICH TEST CASES SHOULD BE AUTOMATED & WHICH SHOULDN'T?

There are many underlying reasons businesses fail to maximize their ROI with software test automation, but the most common cause is the incorrect selection of test cases.

TEST CASES TO AUTOMATE



Regression Tests

This test is repetitive. Automating regression tests can help the testers to identify any errors in functions easily after a code change.

TEST CASES FOR MANUAL TESTING



Exploratory & Ad-hoc Tests

Involves learning and exploration, which should be done by manual testers before initiating testing.



Performance Tests

Automating performance test cases can speed up the testing process, thereby accelerating the product's time-to-market.



User Experience Tests

Measures a product's suitability and accessibility for various users. As each human interacts with an app differently, automation doesn't fit the bill here.



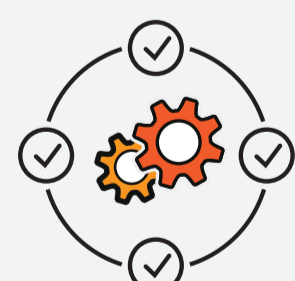
Data-driven Tests

Data-critical processes are prone to human errors. Test automation can eliminate such errors significantly.



Anti-automation Tests

Testing security features like CAPTCHA recognition need human intervention, and can be termed anti-automation test cases.



Other Tests

Integration tests, API tests, Unit tests, Cross-browser tests, can benefit significantly from automation.



Intermittent Tests & Redundant Tests

When automated these test cases can produce unreliable results.

MAXIMIZE THE IMPACT OF TEST AUTOMATION WITH BeatBlip

Leverage HuLoop's award-winning AI-enabled software test automation solution BeatBlip to keep test maintenance efforts and costs to a minimum. BeatBlip's auto-healing and auto-change detection capabilities automatically adapt test scripts to any changes in the Application Under Test (AUT).