

UDC 004.05

## HOW TO SELECT CORRECT TEST CASES FOR AUTOMATED TESTING

I. MISEVICH, Y. PASTUKHOV  
Polotsk State University, Belarus

*I am planning to cite down a few important points based on my experience how to select the correct test cases for automation and determine various other factors that will produce better test results and benefits.*

Automated testing is a Software testing technique to test and compare the actual outcome with the expected outcome. It can be achieved by writing test scripts or using automation testing tools. Automated testing is used to automate repetitive tasks and other testing tasks which is difficult to perform manually.

Automated testing can be defined as a way to run a set of tests over and over again without having to execute them manually. Introducing automation tests in your test strategy is a way to save money and time.

One of the most commonly automated test suites is the regression test suite. Regression, as you may already know, is the test that is done at the end of testing a new module to ensure that none of the existing modules have been affected by it.

It is repeated after each new iteration of testing and the main test cases stay fixed with usually a few new additions after a new iteration. As it is frequently run almost all the test teams try to automate this pack.

The smoke testing is a famous test performed in the test life cycle. These are post-build tests, they are executed immediately after some structure is given out of the application to ensure that the application is still functioning after the structure is done.

This is a small test suite and is something that will be executed many times and thereby it makes sense to automate it. These tests will usually be of a functional nature and depending on the type of application a tool picked for them.

Most of the testing projects are trying to translate their manual test cases to automated ones to improve productivity and coverage.

One of the key steps to commence Automation Testing is – selecting the appropriate test cases.

Automation does not overpower or replaces Manual Testing but it compliments it. Like Manual, Automation needs a strategy with proper planning, monitoring & control. Automation, when implemented correctly, can become an asset to the team, project and ultimately to the organization.

There are many advantages of Automation. Here are a few important to mention:

1. Useful to execute routine tasks like Smoke tests and Regression tests.
2. Useful in preparing the Test data.
3. Helps to execute the Test cases which involve complex business logic.
4. Good to execute the cross-platform test cases (like different OS, browsers, etc.)
5. Great to execute the test cases which are a bit difficult to execute manually.
6. When the number of iterations of the test case executions is not known.

Stakeholders often feel that automated tests act as a support tool for Manual Testing, so it's vital to understand that automation is the best way to increase the effectiveness, efficiency, and coverage of testing. It not only saves time but also improves accuracy as repetitive tasks through the manual approach pronged to human errors and which can be time-consuming.

One of the most basic mistakes which testers make is NOT Selecting the correct test cases for automation.

There is no standard procedure for determining the correct test cases for automation. It all depends on the application you are testing.

**Step 1:** Identify the parameters on which you will base your test case as a candidate for automation.

As of now, I am identifying the following parameters, you can have your own parameters depending on your application.

1. Test case executed with different sets of data.
2. Test case executed with different browsers.
3. Test case executed with different environments.
4. Test case executed with complex business logic
5. Test case executed with a different set of users

---

ICT, Electronics, Programming, Geodesy

6. Test case involves a large amount of data
7. Test case has any dependency
8. Test case requires Special data

**Step 2:** Break each application into modules. For each module, analyze and try to identify the test cases which should be automated based on the parameters.

**Step 3:** Consolidate and group the number of test cases for each module.

**Step 4:** We should also take into account the following attributes:

1. Purchasing and licensing cost of the tool
2. Time to develop the scripts
3. Time to maintain the scripts.
4. Time to analyze the results manually and automatically
5. Time and cost to train the resources.
6. Management overheads

Automating a 100% application is a big task. Not that it is impossible, but it requires proper planning and monitoring and of course; some time. There are lots of permutations and combinations of data, n number of environments with n number of authentication and authorization attributes that need to be validated and hence require a strategy to automate.

In most cases, we prefer to Automate the Regression suite (here are some challenges in automating regression suite in an agile environment) as it contains a larger number of test cases. In that case, we can break down the regression suits into smaller suits and decide to run the appropriate suite as per the release requirement.

Let's suppose that a regression suite contains 1500 test cases, you can break it to 3 suits of 500 test cases per suit and automate it.

Instead of automating the entire suite, you can opt for phase-wise automation. In other words, you can follow the prototype model for developing the automation suite. Create a structure or framework with the implementation of fewer numbers of test cases and start using that and gradually enhance it by adding more test cases to it.

## REFERENCES

1. Википедия [Электронный ресурс] / Автоматизированное тестирование — Режим доступа: [https://ru.wikipedia.org/wiki/%D0%90%D0%B2%D1%82%D0%BE%D0%BC%D0%B0%D1%82%D0%B8%D0%B7%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%BE%D0%B5\\_%D1%82%D0%B5%D1%81%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5](https://ru.wikipedia.org/wiki/%D0%90%D0%B2%D1%82%D0%BE%D0%BC%D0%B0%D1%82%D0%B8%D0%B7%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%BE%D0%B5_%D1%82%D0%B5%D1%81%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5). — Дата доступа: 26.09.2019.
2. habr [Электронный ресурс] / Автоматизация тестирования программных систем — Режим доступа: <https://habr.com/ru/post/160257>. — Дата доступа: 26.09.2019.
3. QA Academy [Электронный ресурс] / Автоматизация тестирования — Режим доступа: <https://qa-academy.by/qaacademy/news/avtomatizaciya-testirovaniya/>. — Дата доступа: 26.09.2019.