zappletech

100% focused on
Test Automation

AUTOMATED VS. MANUAL TESTING: GETTING THE THE RIGHT MIX

Automated vs. Manual Testing: Getting the Right Mix

The QA industry is actively developing. This is evidenced by the research of the analytical department of Markets and Marketers. Experts predict that the field of automated testing will reach a value of almost $50 billion by 2026. This is 2.41 times more than the same figure for 2021. The CAGR level will remain at 19.2% throughout the forecast period.

This rapid growth is due to a number of factors:

- Demand for testing services.
- Need to improve processes.
- Market trends and the introduction of new technologies.

At the same time, the need for classic manual testing has not disappeared, although it has moved to the back burner compared to automation.

Modern testing methods and algorithms involve almost complete automation of processes. Such vectors of industry development contribute to this: RPA, codeless, and the globalization of automation. But even they cannot wholly eliminate manual tests from the practice of QA experts. That is why top experts prefer to mix automated and manual types of testing, achieving 100% coverage of test tasks and higher test completion rates.

Testing Industry Trends

The modern testing market is a testing ground for studying innovative practices and techniques. That is why trends and tests are often the first to appear here. In this way, the industry gets maximum profits much earlier than the technology reaches its target market, such as development or user services.

In order to fully utilize the potential of new technologies, it is necessary to introduce and test them as soon as they appear. This applies to all trending technologies, techniques, or algorithms available in markets. ZappleTech experts keep their finger on the pulse of innovations and are among the first to implement solutions based on them.

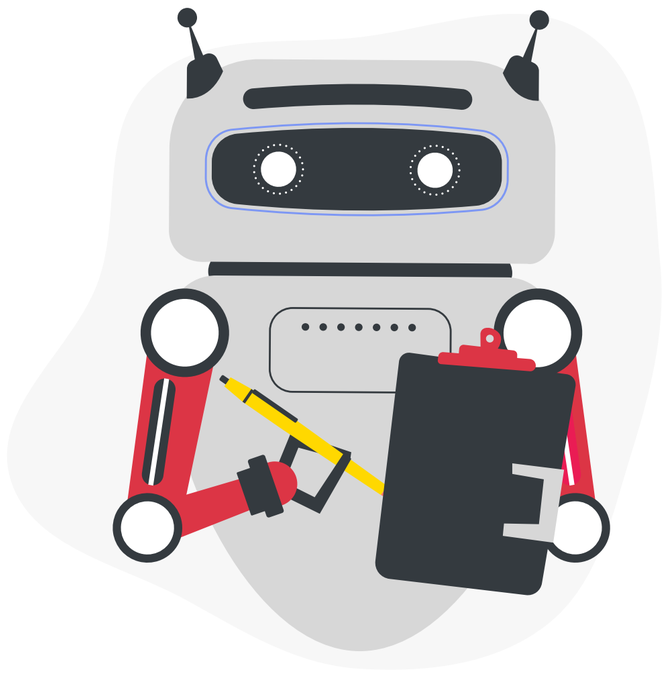
Robotic Process Automation

The point of NoOps (an acronym for "No" and "OPerationS") is to create an automated programming environment with no manual processes. Unlike DevOps, which implies close cooperation between developers and testers, NoOps excludes human presence in operational processes at all.

RPA capabilities for test automation:

- Unquestioning adherence to the set algorithm.
- Cyclic repetition of the test N-number of times.
- Flexible customization of the test script.
- Low-code approach to scripting.
- 100% testing support.
- AI/ML component handling.

RPA simplifies and speeds up many QA processes. That is why trend development for the testing industry is significant.



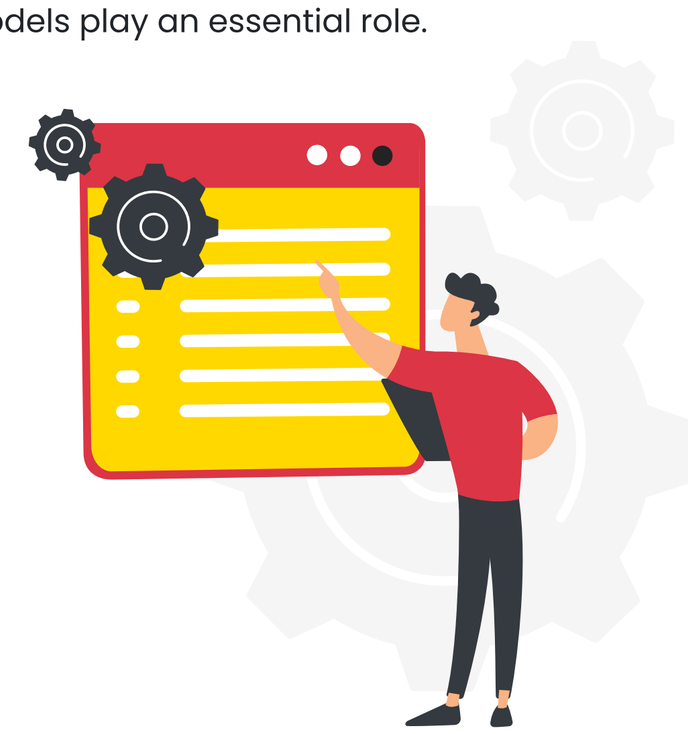
Codeless and low-code

Codeless and low-code programming are called the #1 trend in the development industry. And for a good reason. These models of creating apps or other IT solutions are innovative. They allow you to implement any digital product intuitively without knowledge of frameworks or development languages. In QA, these models play an essential role.

Codeless and Low-code capabilities in testing:

- Intuitive script development.
- Full automation of processes.
- Coverage of 100% of test cases.
- Comprehensible interpretation of reports.
- Flexible customization of cases.
- Low threshold for learning.

All this makes low-code, codeless must-have practices, and tools for teams specializing in testing all kinds of IT solutions.



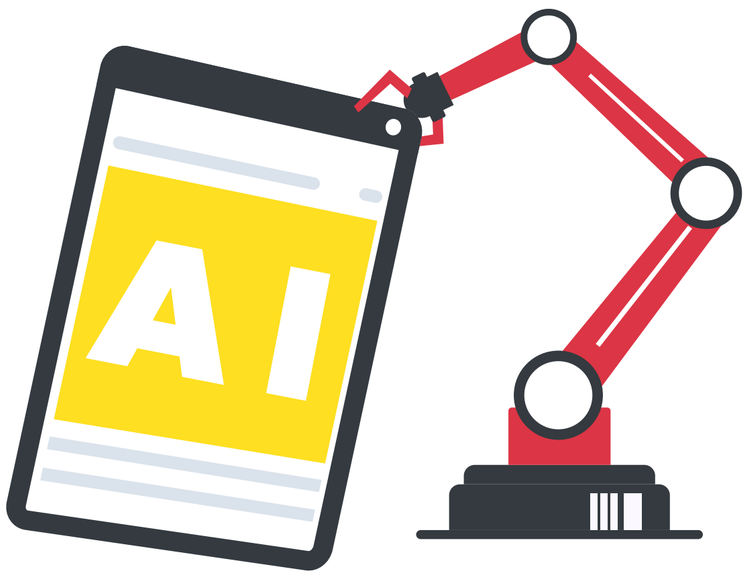
AI

Artificial Intelligence plays a key role in the development and testing industry. Its potential is still not 100% unlocked, although the depth of its integration into QA is already admirable. For example, this technology can be used to autonomously develop and edit test scripts, analyze reports and compile cases.

AI capabilities for testing:

- Automation of script creation.
- Automation of report analysis.
- Automation of test launching.
- Automation of algorithm development.
- Automation of bad cases fixing.
- Automation of code research.

With the development of AI, it will be possible to minimize the use of manual work in software testing. A single engineer will be enough to develop the algorithm, train the AI and get it up and running. For analytics, though, you will still need a staff of experts capable of evaluating testing progress.



Global automation of testing processes

Global automation is a trend throughout the IT industry.

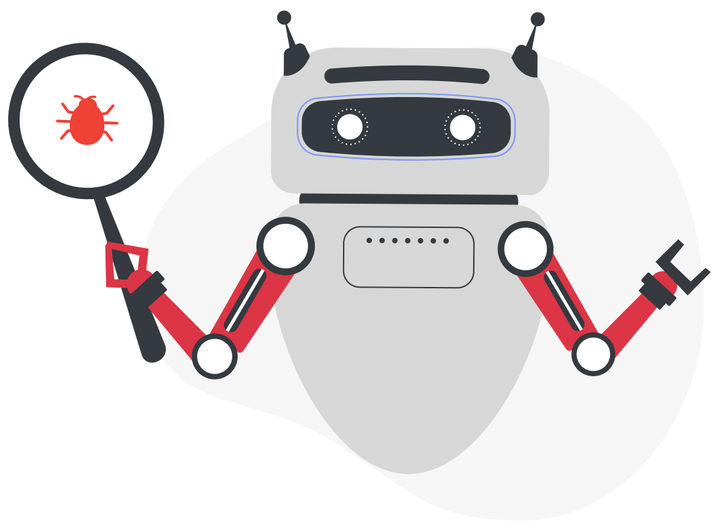
The trend is aimed at reducing the cost of resources in routine processes. For example, when developing a digital product, part of the operations can be delegated to AI, and the other part can be automated with RPA. Thus, experts will only need to monitor the work of the algorithms, correcting them periodically.

> According to:
>
> **Fabrizio Biscotti**
> vice president of research at Gartner
>
> “Hyper-automation has gone from being an opportunity to a condition of survival.”

Global automation in testing:

- Performs most of the routine tasks.
- Reacts to changes of TOR and code.
- Self-improving.
- Develops cases and scenarios.
- Controls processes.
- Analyzes report.

This trend is aimed at optimizing work in all areas. By parallelizing processes and correcting them, global automation reduces the human impact on testing and minimizes the risk of missing critical errors.



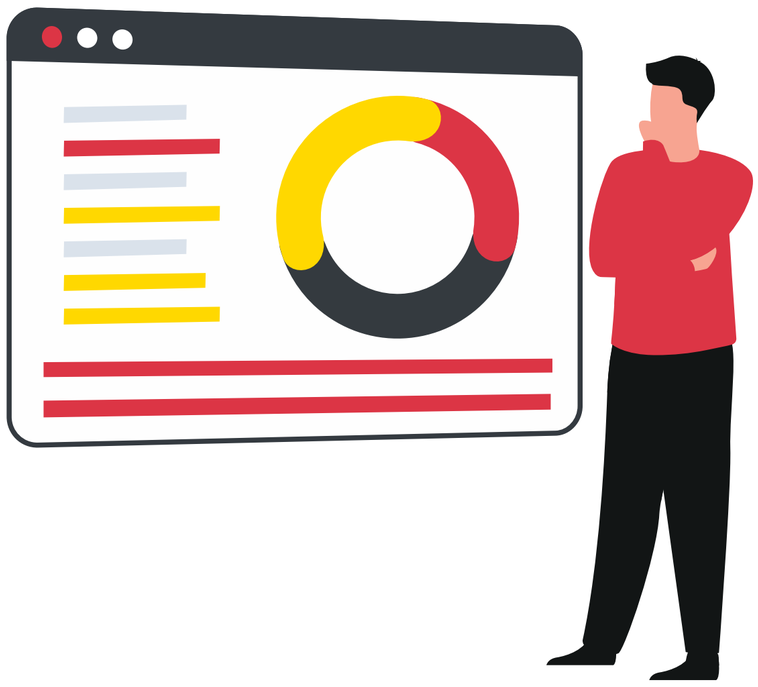
UX as a key factor in QA effectiveness

The trend to improve user UX is designed to solve a key problem in the development and testing industries. Namely, eliminate critical errors when using software, a website, or any other IT product. This will unlock the software's full potential and ensure maximum performance, usability, and quality.

The importance of UX for testing:

- Helps find key interface problems.
- Eliminates functional bugs.
- Optimizes architecture and control logic.
- Ensures design adaptability.
- Unleashes the potential of AI in the software.
- Collects reports from real users.

That's why it's essential to use UX and customer feedback during development and testing. Focusing on real-world scenarios and cases allows you to leverage the most engaged audience and optimize the product for user needs.



General Types of Testing

In the context of automation, you need to consider its feasibility for different types of testing. Some of them can be fully delegated to scripts and code, and some are easier and faster to test manually. The cost-effectiveness of automation depends on how often a test repeats, how many pass cycles are needed, or how many tests need to be run.

Unit testing

Unit tests require careful analysis because they test each component in the software. Using automation for their testing is more reasonable because you need several full cycles of running tests with different relevant values. This routine is best delegated to a program, except when you need one particular test for quality control.

Component testing

There are two variants of component testing: CTIS and CTIL, on a small and large scale, respectively. Unlike unit testing, it is performed by QA experts and can be automated by engineers. Is it reasonable? Yes, if you need to check a large algorithm. But it is better to control the process manually to observe the results personally.

Smoke testing

Smoke tests should be automated. This will speed up the process and reduce the amount of routine for the QA specialist. There is often a need to run more than 500 tests in a single sprint. Automation will do away with the manual processing of each test, although it will require additional time to write complex scripts.

Sanity testing

It is not reasonable to automate sanity tests. The small number of repetitions, the comprehensiveness of tests, and their peculiarities allow you to check functions and components rather quickly in manual mode. It is performed after small changes in the code, requiring minimal time and effort from the tester.

Regression testing

Regression testing is always automated. First of all, because of the large number of different tests, and second, because of the possibility to reuse scenarios at later stages. This minimizes resource costs and fully recovers the initial investment in developing an automated test cycle.

Integration testing

Integration testing can and should be fully automated. Instead of manually running dozens of test cases, it makes more sense to use cloud computing and process parallelization. This saves time and resources for QA experts.

API testing

API testing is performed both manually and automatically. In any case, before creating complex repeatable scenarios, QA specialists will have to develop a testing algorithm on their own. This is necessary both for understanding the essence of testing and for a more correct analysis of reports.

UI testing

The interface is tested automatically, except when you need to test a certain element after changes. Since the algorithm of actions usually does not change, it makes sense to loop the test script and deploy it in a test environment. This way, you can speed up interface testing on different devices by paralleling tests in virtual environments or cloud emulators.

Manual or Automated: key benefits of both methods

There is always a favorite in arguments and debates about the best testing method. Some people prefer slow but reliable manual testing, while others try to automate the process as much as possible. And they are all right in their own way.

Let's look at the key advantages of both methods.

Strengths of manual testing:

- **Control of the process.** The tester can independently determine the course of action and analyze test results on the fly.
- **Validation of results.** During the process, you can observe how the software or its separate fragment works, allowing you to isolate problem areas faster.
- **Quick launch of single tests.** Scenarios that do not require multiple runs are easier to run manually and report back afterward.
- **Case detailing.** By running a test on their own, the QA expert observes the process and has a better understanding of what exactly the bug is about.
- **Random tests.** In the development process, you can randomly run tests to check the functionality of a function or code.
- **Creativity.** Using non-standard tests, you can completely rule out the possibility of an error in the end-users.

Automation tricks:

- **Quickly run repeatable tests.** You can loop most of the scripts and see the results of multiple runs. The chance of error detection is increased.
- **Code reuse.** Once a script has been developed, a QA expert can run it repeatedly at different stages, making minimal adjustments.
- **Accelerated routine.** When running 100+ single-type tests, it's easier and more suitable to use automation, reducing the amount of manual work.
- **Eliminating errors.** Repeated tests should be used for better error indication. This eliminates the risk of an undetected bug appearing after changing the code.
- **Load balancing.** By using cloud computing power, it is possible to run more tests simultaneously than on the production machine.
- **Test parallelization.** External emulators enable cross-platform testing without physical devices.

As you can see, both methods are necessary to provide the highest possible level of QA services. In this case, the testers' main task is to learn how to properly combine automation and manual tests to achieve the best results.

Combination is a recipe for success

Combining manual and automated testing methods is necessary in today's QA industry. To achieve maximum results, testers must correctly estimate the value of each test and distribute the load. The unspoken rule is: automate everything we can, and what we can't, test manually.

The initial cost of automation more than pays off during testing, when the team can do the work instead of creating scenarios from scratch. That's the value of reusable code. It helps reduce the load on QA experts and allows them to focus on current problems.

Manual tests are performed for verification of results, random checks, or in cases where it takes longer to write tests than to perform a cycle of actions. This is called resource optimization. While the engineer creates the test, the QA specialist checks each script manually, adjusting efforts and incidentally identifying problems.

Naturally, 100% of test cases are difficult to cover with automation. There is always a risk of missing an obvious bug when re-running the same type of script. This is why you should combine automatic and manual testing. You will achieve the most efficient results while saving your team's resources.

Proof

According to Venturebeat report, about 97% of QA experts have high hopes for automation, although they continue to use manual techniques. It is connected first of all with a trend of hyper-automation of processes, which market has already reached the value of $0.5 trillion.

The same report states that 76% of respondents have automated only 50% of their test tasks. This indicates that there are a number of test cases that are easier and cheaper to perform manually. And although this takes longer, the results are more reliable than running a systematic cycle of scenarios.

Let's Summarize

Each tester decides which expertise they want to increase: automatic or manual. They are both in demand in the complex QA process and compensate for each other's disadvantages. That's why the eternal argument about the better approach can be considered closed. Only by combining manual and automatic testing will you get the maximum QA process efficiency.

Thanks for reading!

Want to get high quality testing services?

Click the button below for more information about our automated testing services

Contact us

Zappletech - First-class Automation Testing Company

100% focused on Test Automation Outsourcing and Services to deliver automated test solutions that reduce costs & increase efficiency to get better software delivery.

Zappletech implements the most effective automation solutions for business benefits for over ten years, ready to provide our customers with the best products on the market.

Working with us, you will have quick engagement, proficiency in automated tests, and an organized engineering team with a product development mindset that will develop necessary test automation and infrastructure quickly in a cost-effective manner.

Web Test Automation

Mobile Test Automation

DevOps Test Automation