



The Difference Between Manual and Automated Testing

2019 | Ramin Mammadov

Introduction

Depending on the product or application that is being tested, an organization can use an automated or manual testing method. Which method to choose comes down to the associated costs and benefits on a particular project. Both manual and automated testing have advantages and disadvantages. The following need to be taken into consideration when deciding on testing methods: time, resources, budget, the size of the project, the quality of the tools, and knowledge of the testing team. Combining both manual and automation is also an option that can help to develop the best software possible.

Manual Testing

Manual testing is the process through which QA testers run their tests manually, comparing expected and actual results to find software defects. The tester would approach manual testing as an end-user and then determine whether or not the application acts appropriately. The tester will check all the essential features of the given application or software. In this process, the software testers execute the test cases and generate test reports without the help of any automation testing tools. Manual testing is usually a good fit for small size projects.

Advantages of Manual Testing

- » The short-term cost is lower. Automation tools can be expensive, depending on the project size and budget. Manual testing does not require up-front costs into the software.
- » Real user issues are more likely to be found. Automated tests don't act as a real user would, and any bugs that may pop up in the production are more likely to be caught with manual testing.
- » Flexibility. When something needs to be changed in the course of the project, it can be done immediately during manual testing. This is complicated with automated testing. It will require to set up test cases, program it into the automated tool, and only then run the test. With manual testing, it can be quickly tested to see the results.

Disadvantages of Manual Testing

- » Certain testing is difficult to do manually. For example, low-level interface regression testing can be difficult to perform manually, and as a result, would be prone to mistakes when done by hand. This type of testing is better done by setting up automation.
- » Not stimulating. Manual testing can be repetitive and therefore, boring. As a result, many testers may have a hard time staying engaged in this process, and errors are more likely to occur.
- » Manual tests cannot be reused. If there is any change to the software, the entire test will have to be run manually again, which is very time-consuming. With automated tests, if there are any changes in the software, required tests can run instantly because the test set is already set up.

Automated Testing

Automated testing is the process through which automated tools run tests that repeat predefined actions, comparing expected and actual outcomes. Testers use appropriate automation tools to develop test scripts and validate the software. The goal is to complete test execution in a shorter amount of time. Automated testing allows the execution of repetitive tasks and regression testing without the intervention of a manual tester. This method of testing is good to use when the project has a large scope of testing.

Advantages of Automated Testing

- » Runs tests quickly and effectively. Initial set up might take a while, but once the test cases are automated, the process will run faster. Testing can be reused for large regression suites during constant code change which will reduce the manual efforts.
- » It can be cost-effective. Automation tools can be expensive in the short-term. However, these tools save money in the long-term. They can perform faster and can do more than a human tester in the given amount of time, and these tools find defects quicker.
- » The whole team can see the test results. The automated testing framework allows people to sign into the testing system and see the results, creating an opportunity for greater team collaboration and the creation of a better final product.

Disadvantages of Automated Testing

- » Tools can be expensive. Depending on the budget allocated for a particular project, an automation tool can be a costly purchase.
- » Tools take time to set up. While the automation process cuts down on the testing time, it is still a time-intensive process. A considerable amount of time goes into developing the framework and writing automated tests.
- » Tools have limitations. While automated tests will detect most bugs in the system, there are still limitations. For example, automated tools cannot test image color or font size—testing like this can only be done manually.

Deciding Between In-house Testing or Outsourcing

One benefit of outsourcing is that it does not require the staff to be on hand within the company. It provides the financial benefit of outsourcing to an area where taxation and employee pay is low. Outsourcing can enhance the testing processes because the company can employ testers who are experts in a wide variety of software testing methodologies and techniques.

There are also disadvantages to outsourcing. It brings the concern of language barriers and time zones, which can affect the productivity of in-house developers. Outsourcing also brings a lack of managerial control as the project might suffer without direct oversight. Employing in-house testers improves communication between developers, managers, and QA professionals.

There is also a component of pride to an in-house testing team. Since their company's name is attached to the product they're testing; they might be more thorough in their testing than an outsourced contractor. In the end, the decision to outsource versus employ an in-house testing team is one that depends significantly on a company's specific situation. It is also possible to use either or both depending on the particular project.

About Level

Level helps clients transform their business with strategic consulting and technical execution services. We work with your IT organization, product groups, and innovation teams to design and deliver on your technical priorities.

Our Engineering team is made up of technology agnostic enthusiasts with a wide array of knowledge in popular modern and legacy languages such as Java, .NET, Ruby on Rails, Node.js & JavaScript, Python, and PHP.

We firmly believe that mentoring can be integrated with delivery. Our main focus is on saving our partners as much as possible on the lifetime-total-cost of ownership and maintainability of their systems. For more information, contact us at hello@level.io.