TM Research & Development

Understanding the Role of Web Testing in Product Development

White paper

Executive Summary

Web testing has become an important aspect of product development to ensure it meets the requirements while functioning at an optimal level. There are two methods for web testing and they are manual and automation. Manual web testing differs from automation testing as it uses different testing tools to run its test scripts before verifying the website. Once the test suite is automated, it does not require much human intervention.

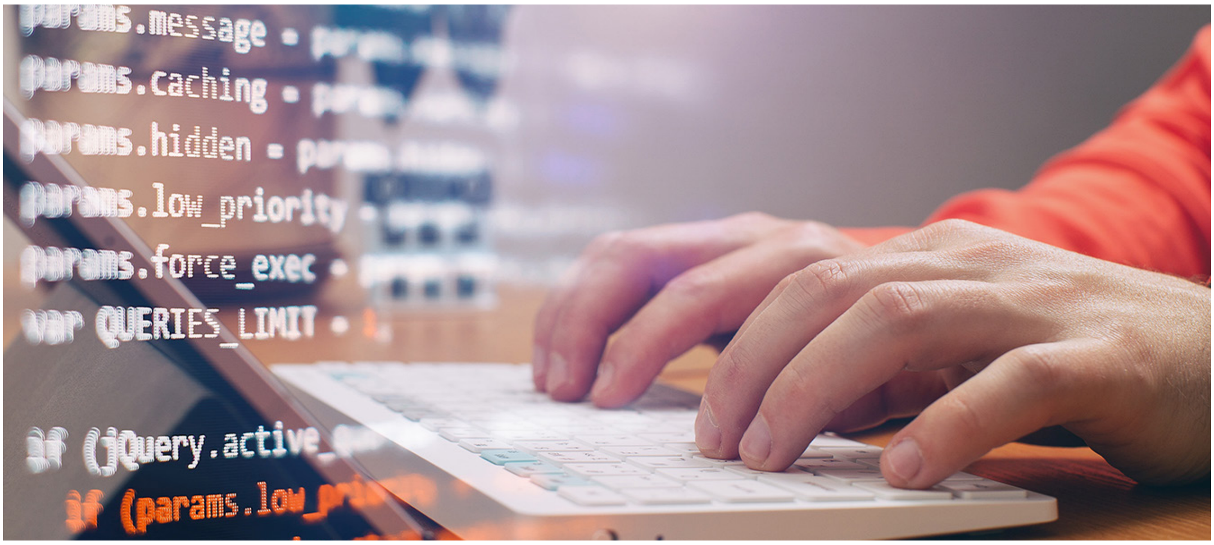
In the 1990s, only a few browsers had been created including Netscape (the present-day Firefox) and the current Internet Explorer. Given the infancy of the internet and the scarcity of available browsers at that time, manual testing appeared to be the chosen method used by developers to test their websites.

With the constant development and release of new operating systems and browsers, the Internet underwent an evolution. Because of this, the Quality & Assurance (Q&A) team faced a challenge in testing websites performance across various devices that uses different browsers and operating systems. Different browsers displayed websites differently, so it became imperative for developers to create pages that were compatible with various browsers. The test had to be both functional and show visual competence.

Here is where automation testing comes into the picture. It helps to save time for cross-browser testing as the web infrastructure became more diversified globally.

Manual testing soon proved to be insufficient and appeared to have difficulty in fully covering the entire range of test configurations contained in the current web infrastructure. An innovative solution to this issue is automation testing. Web automation testing turned out to be a very workable technique for obtaining extensive test coverage.

This white paper will discuss the ins and outs of Manual and Web automation testing, and why it is important for the success of any web product.



Introduction

Web testing is a type of software testing that focuses on web applications, analyses and monitors performance, functionality, compatibility, usability and security. It is conducted during the development process as well as before the product is delivered to the market.

Testing life cycle

A testing life cycle can be divided into requirement analysis, test planning, test case development, environment setup, test execution and test cycle closure. During the requirement analysis phase, the QA test team is able to review the Functional Requirement Document (FRD) to analyse the needs and features warranted for the web application. This information then helps them determine what and how the test will be carried out.

After reviewing the FRD, the test team needs to define the test strategy and test plan to determine exactly what to test and how to carry it out. During this phase, the test plan and test effort estimation document will be completed.

In the test strategy phase, the test team will need to define the scope, testing approach, test environment, test scheduling and testing tool to be used.

Next comes the test case development phase, where the test cases are developed to trial out the web application. Test data needed can be prepared in this phase to avoid time wastage when running the test.

The Requirement Traceability Matrix (RTM) is prepared during this phase. RTM is a format for tracking requirements to ensure each test case is mapped with the requirements.

For the environment setup phase, the QA test team does not necessarily play a role. The developer may create a test environment depending on the company or situation. However, the test team will still need to check through the setup to ensure the test data needed is prepared and ready to be used.

After the test environment is set up, the test team can start running the tests. The testers will execute each test and compare the test results with the expected results. They will then mark each test case with a pass or fail. If the test is deemed as failed, testers need to investigate what happened during the test.

If there are bugs or any defects found, the test report is then returned to the development team. After the defects are fixed, the testers then rerun the test case to make sure the bugs are solved.

After all tests are concluded, the QA test lead makes sure all test cases have been run and completed. After that, a final test report is filed which will include the analysis of defects found and other metrics. Evaluation of test coverage, cost and time spent can be reported with the aim of improving any future test projects.

Types of web testing

There are a few types of web testing. Testers can choose which is the most suitable one depending on the testing requirements.

1 Functional Testing

Functional testing is carried out using a variety of testing criteria including interface, APIs, database, security and basic website functionalities. The aim is to test how well each website feature and function works. Functional testing activities include testing the links, forms, cookies, HTML and CSS. For instance, testing the form if the mandatory fields are not filled, should not allow the form to be submitted. If the form has been successfully submitted, the test case is deemed as failed and will be considered a defect.

2 Usability Testing

Usability testing has become increasingly important in web-based testing projects. This testing can be divided into two parts, testing the site navigation and the content. Site navigation consists of the menu bar, buttons and links while content testing looks to make sure it is legible and has no spelling errors. If there are images on the page, the images should be clear.

3 Interface testing

Functional testing is carried out using a variety of testing criteria including interface, APIs, database, security and basic website functionalities. The aim is to test how well each website feature and function works. Functional testing activities include testing the links, forms, cookies, HTML and CSS. For instance, testing the form if the mandatory fields are not filled, should not allow the form to be submitted. If the form has been successfully submitted, the test case is deemed as failed and will be considered a defect.

4 Compatibility testing

Compatibility testing is done to make sure that the web application works properly on different devices using different browsers and operating systems. Some of the websites may be displayed differently on different browsers. With a change in the operating system, the rendering of web elements like buttons could change.

5 Performance testing

Performance testing is conducted to determine if the performance of the web application is successful under various loads. The activities in this testing include testing the website response time at different internet speeds, testing the behavior of the web applications under normal and huge loads, and testing how the site recovers from a crash due to peak load.

6 Security testing

Security testing is important especially for e-commerce websites as it stores vital and private customer data. Therefore, testing needs to be done to make sure that the website can secure the users' personal information. During this phase, tests are conducted to determine whether a web page can block unauthorised access while restricting files that are not allowed to be downloaded. A test is also done to check if sessions are automatically turned off after user inactivity

Web automation testing tools

There has been an avalanche of choices for web testing tools but for this paper, three chosen ones will be discussed, Selenium, Playwright and Tosca.

1. Selenium

Selenium is a popular web automation testing tool where over a decade ago, this open-source testing tool became the number one choice for testers. It allows testers to use the Selenium WebDriver library along with a testing framework to automate test cases for selected browsers. Selenium supports different programming languages, browsers and operating systems. It also allows parallel testing across different browsers. One of its strengths is that it is able to integrate with many tools to support additional features.

2. Playwright

Playwright is an open-source and cross-browser automation testing tool. It started off as a JavaScript-based library but grew to support Python, Java and Net. Playwright supports automated testing on any browser, it can also support cross-platform testing. Testers can use it to test on Windows, Linux and MacOS, headless or headed.

3. Tosca

Tricentis Tosca is an automation tool mainly used for functional and regression testing. Unlike Selenium and Playwright, Tosca is a licensed automation tool, which means there is a fee that is charged if it is to be used. It has a special feature that does not require any type of coding, meaning it is easy to learn as it does not require programming skills from the testers. Because it is still new to the testing industry, not many have begun using it yet, causing it to have little support from within the industry.

Now let's do a simple comparison of these three automation testing tools.

	Selenium	Playwright	Tosca
Availability	Open source	Open source	Licensed
Languages support	Java, Python, Ruby, C#, Perl	JavaScript, Java, Python, C#	Visual basic, Java, C#
Supported Browsers	Google Chrome, Internet Explorer, Safari, Microsoft Edge, Firefox, Opera	Google Chrome, Opera, Firefox, Safari	Google Chrome, Microsoft Edge, Firefox, Internet Explorer
Coding Skills	Require coding skill	Require coding skill	No coding is required
Reporting features	Lack of built-in reporting feature, needs to integrate with other reporting tools like TestNG	Comes with a few built-in reporters	Has built-in reporting and also a control panel for viewing detailed information like timing graphs

Automation vs Manual

The most basic difference between manual and automation is, one requires human intervention while the other is fully automated.

For manual testing, a tester performs the test step by step, often without a fixed scenario. Meanwhile, in automation testing, the tool fully operates the job, thanks to the automation framework, which dots the i's and predictably crosses the t's.

What's important to know is that there's no right or wrong way of testing software. Both manual and automation testing have their rightful place in the quality assurance process. Depending on the project, one can be used more often than the other, but it's worth noting that neither is inherently better than the other. Both are simply used in different scenarios and to varying extents.

Here are brief explanations for both formats of testing, alongside the pros and cons. We will also look at how testing can benefit businesses in the long run.

Advantages of manual testing

The most positive aspect of manual testing is human intervention and attention during the entire process.

Manual testers can handle complex scenarios and execute tests even when there's a nuance that hasn't been taken into consideration beforehand. For now, it is said that there's no better software testing tool than the human brain.

Other advantages of manual testing include:

- quick feedback based on human intuition and critical thinking
- performing ad-hoc testing almost instantaneously – which is perfect for testing small changes in codes.
- experimenting with newly designed test cases
- testing different kinds of software
- a flexible approach that is compliant with agile development
- smaller costs since none of the automation tools and processes are involved

Generally speaking, all software has to be tested manually before it is automated because humans are the end-users; if something occurs to the tester, there is a high chance that it will be noticeable to the user.

Disadvantages of manual testing

Despite the many advantages, at the end of the day testers are all people, and with that comes the risk of human errors.

Some disadvantages of manual testing include:

- prone to errors
- can be more time consuming
- some tasks may be difficult to perform manually
- can become expensive in the long run

In short, the biggest advantage and disadvantage of manual testing is the human aspect. Testers aren't immune to conditions like fatigue or boredom, which can negatively affect the test results.

Advantages of automation testing

Automation testing is favorable because of these factors:

- a higher detectability of bugs
- faster speed
- reusability of the test scripts as opposed to manual testing – automation testing can be recorded
- increased test coverage
- lesser cost in the long run

In general, automation testing has all the advantages machines have over humans.

Disadvantages of automation testing

Even though automation testing often seems like the perfect option, it does come with some drawbacks, which include:

- the cost of automation tools can be very costly
- automation testing tools are not fool proof and require maintenance by testers
- it doesn't work for every digital product because it's not as flexible as manual testing
- automation testing does not take into account the visual aspects of digital products
- automation testing is the most suitable for stable products but it's not as effective in projects where the requirements are prone to changes.

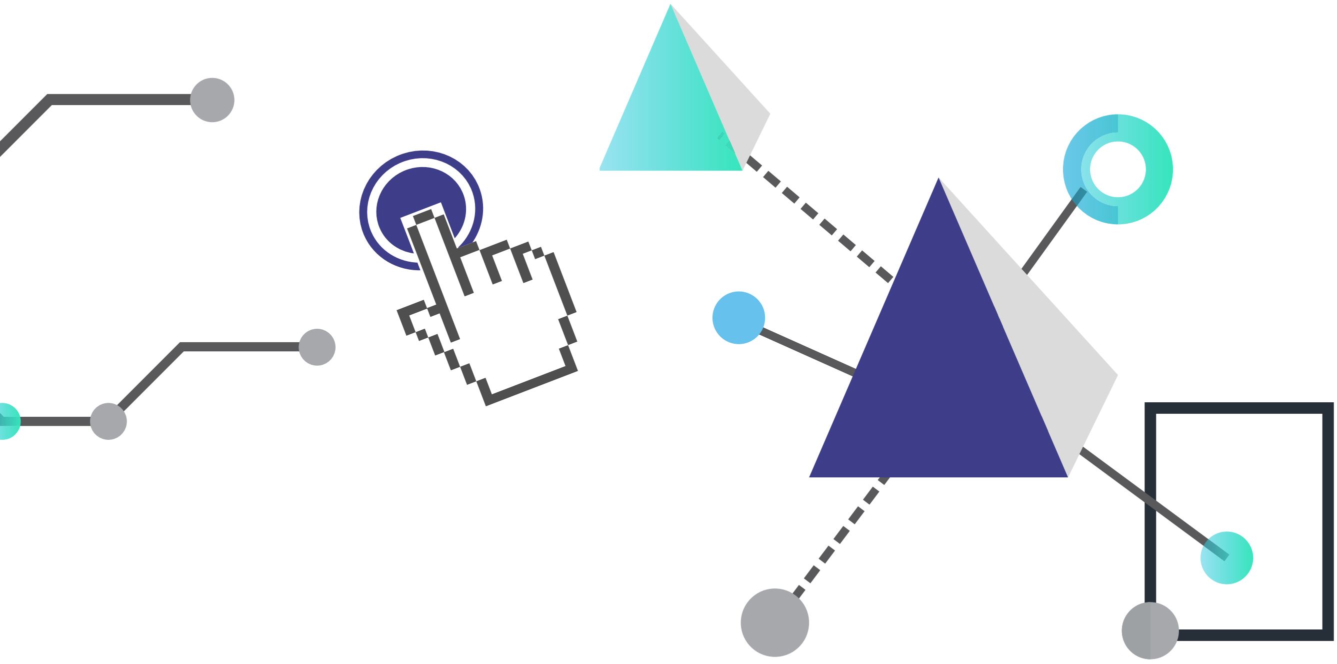
Summary

The Quality Assurance phase is a complex process in software development where testing is just one part of it. Despite the wide variety of testing types, it is not advisable to rely on only one component. Both automated and manual testing have their strengths and weaknesses. While automation testing is fast and efficient, not everything can be automated. The digital product benefits the most when automation and manual testing are performed side by side in the relevant stages of development. Therefore, combining both can bring out the most optimal results.

TM R&D chooses Selenium as the automation tool for web testing. More importantly, it is an open-source tool. Selenium is more flexible as it supports more programming languages and browsers. Its strong community support and feature-rich test framework are also significant strengths of Selenium. Selenium can be integrated with different frameworks and tools to eliminate or minimise its weaknesses.

At TM R&D automation testing is done during the web development process so that any defects can be discovered immediately while developing the web. Defects or bugs discovered will be reported and returned to the development team to debug the errors. After debugging, testers will do the test again to make sure the defects are fixed and no other bugs are caused by the changes. It saves time and increases efficiency.

Testing both manually and automation are not only important for the success of the product but also to gain customers' trust so that they keep using it.



Creator of Smarter Ecosystems for a better Malaysia

Established in 2000, TM R&D is the innovation arm for TM Group focusing on creating smarter ecosystems to make business and life easier for a better Malaysia. TM R&D's solutions are clustered around four (4) pillars namely Intelligent Platforms, Data Brokerage, Connectivity/Tools and IR4.0/Digital Solutions.

Growing from strength to strength since 2016, TM R&D has won multiple global awards and generated more than 2,800 Intellectual Property Rights (IPRs) and 1,400 digital assets to-date.

TM R&D's innovations are all developed in-house and cut across multiple verticals such as Utilities, Retail, Agriculture, Healthcare and Education with safety and productivity as the top priority.

As TM R&D continues to expand beyond connectivity and into smarter digital ecosystems, its role in TM has become more prominent and exciting.

We are looking for remarkable people to join us. People who are courageous enough to push boundaries, curious enough to experiment with new technologies, and who have the determination to drive new ideas forward. A new opportunity awaits you here in TM R&D.

Be a part of our family at https://www.tmrnd.com.my/jobs/ or email recruit@tmrnd.com.my

For more information about TM R&D and its products and services, visit **www.tmrnd.com.my**