

Comparative Research Analysis of Automated Testing Tools Selenium & QTP for Banking Website

Ghanesh Jain

M.Tech (Scholar) Computer Science,
MITRC, Alwar, Rajasthan, India

Arun Singh Chouhan

Assistant Professor, Computer Science,
MITRC, Alwar, Rajasthan, India

Abstract—

Software testing is an activity which is aimed for evaluating quality of a program and also for improving it, by identifying defects and problem. Testing can be described as a process used for revealing defects in software and for establishing that the software has attained a specified degree of quality with respect to selected attributes. The paper would provide distinction between automation and manual testing of an internet banking website, specifically highlighting the contrast in cost, time and ROI. It would also bring up a comparison between some of the test automation tools available in the market and their effect on the project if used.

A business critical application like an internet banking website needs to be thoroughly tested for any residual bugs remaining in the code as any failure could lead to huge business loss. Using functional testing, some of the possible defects that may lead to failure could be removed. Automated testing included the development of scripts that not only saves time and resources when applications are updated, but also speeds up the process of testing. Test automation is extensively used for regression testing, which seeks out new bugs in a program and separates them.

This paper highlights the comparison between QTP and Selenium automation tools available in market and their use in the software project scenario.

Index Terms— Software testing, manual testing, automated testing

I. INTRODUCTION

Software testing is a crucial part of software development in delivering a quality software product that is free from bugs and defects and the process of automating software testing is vital to its success. Testing is important because software reliability is defined using testing and approximately fifty per cent of the software development budget for software projects is spent on testing [7].

Software testing is laboured intensive and expensive; therefore, there is a need to reduce human testing [3]. Software testing is necessary because errors are often introduced into software inadvertently as it is designed and constructed [7]. Software has become even more complex today, which means there are more lines of code, and more thorough testing that needs to be done.

William Howden, a professor from University of California at San Diego, wrote that “testing is the unavoidable part of any responsible effort to develop a software system”[4]. Basically testing is done by manually and automatically[1]. Manual software testing is done manually that is it requires human input, analysis and evaluation. Software test automation is the process of automating, the steps of manual test cases using an automated tool or utility to shorten the testing life cycle with respect to time[1].

The purpose of testing can be quality assurance, verification, and validation or reliability estimation. It is a trade-off between budget, time and quality. Software Quality is the central concern of software engineering. Testing is the single most widely used approach to ensuring software quality. [2]

Testing could be further classified into Functional and Performance testing. Performance testing analyses if the performance parameters like turnaround time, reliability, load capability etc. are up to the client expectations. Functional testing aims to check, if the functionalities provided are correct as per business requirements and are working as intended.

It needs to be critically analysed if the investment on testing could provide a suitable return. Though it is always recommended to go for testing, timelines and budget usually makes the project team skip the same.

II. OBJECTIVE

The main objective of this research paper is to

- (i) To identify the advantages of automated testing tools and their impact on software testing
- (ii) Analysis of automation tools available in the market so as to infer, that which tool will create test ROI for the customer and

III. BASIC STRATEGIES OF SOFTWARE TESTING

Software testing is the execution of code using combinations of input and state selected to reveal bugs.

In this section we introduce to manual testing and automated testing. In order to reduce the cost of manual software testing researchers are working towards increasing the automation of software testing [5].

3.1 Manual Testing

A process in which all the phases of Software Testing Life Cycle (i.e. Test Planning, Test Development, Test Execution, Result Analysis, Bug Tracking and Reporting) are accomplished by manual means i.e. with human efforts.

Drawbacks of Manual Testing:

1. More people are required.
2. It is more time consuming.
3. It renders less accuracy.
4. Simultaneous actions are difficult.
5. One cannot reuse a Manual Test.
6. Manual Tests provide limited Visibility [8]
7. Scripting facilities are not in manual testing [8]
8. Performance testing is not possible with the help of manual testing.
9. Comparing large amount of data in case of manual testing is difficult.
10. Automated testing generates logs and repositories automatically but in manual testing we need to create the test case manually.

3.2 Automated Testing

The real use and purpose of automated test tools is to automate regression testing. To perform it there should be a database of detailed test cases that are repeatable, and this suite of tests must be run every time there is a change to the application to ensure that the change does not produce unintended consequences.

Automation testing is not a replacement for manual testing. It is in fact a continuation of manual testing aimed to provide speed and accuracy to the testing effort. Automated testing primarily uses an automation tool, which is assistance to the test engineers.

Automated tools are broadly of three Types like:

- 1) Functional Tools (like HP UFT, IBM RFT, Selenium)
- 2) Management Tools (Like HP ALM, JIRA)
- 3) Performance Tools (Like HP Load Runner, JMeter)

An automated test script is a program. Automated script development, to be effective, must be subject to the same rules and standards that are applied to software development.

The needed automated testing tool should provide ability to perform the following types of testing:

- Regression Testing (for Web interfaces)
- Unit/Integration Testing

Drawbacks of Automation Testing:

- (i) More expensive.
- (ii) We cannot automate all the areas.
- (iii) Manual testing cannot be fully eliminated
- (iv) Requires a higher skill level of team members.
- (v) Proficiency is required to write the automation test scripts.
- (vi) Automation testing cannot be stopped in between in case any analysis required.

3.3 Automation versus Manual Testing

Though, manual testing can never be ruled out, automating the test cases is a good idea for regression suits as it saves a lot of time, man force and thus, investment. The main differences of the two are as given below:

S. No	Automation Testing	Manual Testing
1	Fast: Automated test cases could be executed a lot faster than manual counterparts.	Time consuming and tedious: As the test cases are executed manually, it is very slow and tedious.
2	Less investment in human resources: Test cases are executed by using automation tool so fewer testers are required in automation testing.	Huge investment in human resources: As test cases need to be executed manually so more testers are required in manual testing.
3	More reliable: Automation tests perform precisely same operation each time they are run.	Less reliable: Manual testing is less reliable as tests may not be performed with precision each time because of human errors.
4	To run the test multiple number of times.	For executing test cases for small number of times.
5	Helps performing "compatibility testing" - testing the software on different configurations.	It allows the tester to do more ad-hoc (random testing).
6	Long Term Cost reduction.	Short term costs are reduced.
7	Capable of finding only expected defects.	The more time tester spends testing a module the grater the odds to find real user bugs.

IV. AUTOMATED TESTING TOOLS

4.1 QTP (Quick Test Professional)

Vendor-HP Software Division

It is known to be the industries best solution for functional testing and regression test automation. It uses keyword driven approach providing the test automation engineer full access to underlying test and object properties via an integrated scripting and debugging interface.

Object Repository: It has a well-defined object repository which stores the information of all objects being used in the script.

Robust Function Libraries: Enables sharing of function libraries within tester workgroups.

Intelligent Pre-Run Validation: Runs a pre-execution resource check automatically, notifying users of missing files or resources.

Enhanced Keyword View: Drag-and-drop test steps within the Keyword View's natural language environment.

Open XML Report Format for Test Results: Stores test results in an open XML format, enabling you to easily customize the reports according to your own requirements, and to integrate the test result information with other applications. Test results can now be exported to HTML.

New IDE Environment: Offers a highly customizable test development environment.

New Debugger: Enables testers to pin-point test errors when building and maintaining test cases.

Keyword Management: Manage keywords, including turning on/off specific methods from the Keyword View.

Multiple Document Interface for Function

Libraries, Object Repositories: Allows users to concurrently open and edit multiple function libraries and Object Repositories within the QuickTest Professional interface.

Environment Support: QuickTest Professional supports functional testing of all popular environments, including Windows, Web, .Net, Visual Basic, ActiveX, Java, SAP, Siebel, Oracle, PeopleSoft, and terminal emulators. New Environment Support: Supports Web services, .NET 2.0, Firefox 1.5, Netscape 8, Macromedia Flex 2, Win XP 64 bit, Internet Explorer 7, and the latest ERP/CRM applications. .

Scripting Language: The choice of VBScript is not very good, because VBScript has some significant limitations: like, it does not support modules/libraries and cannot be

4.2 Selenium

Open source Tool

Selenium is the fastest growing automation tool which has gulped the features of the best automation tools available in market and providing an edge over them.

Object Repository: It has a highly defined object repository which properly stores and displays the objects in the hierarchy of their presence on the screen.

User Codes: It enables the user to create any kind of scenario using coding which adds to any feature already present in the tool.

Code Compiler: Brings out any code related issue in the test script while compiling the code for execution.

Enhanced Keyword View: Elaborate keyword view with all the functionalities and add-ons

Graphical Report: The test report generated is complimented with graphs for faster and better comparison of defects in every run using TestNG framework.

Step View Scripting: The interface provided is a non-code based interface which makes it easy to track missing steps.

Image Based Testing: It has a very developed image based object recognition system which could be useful to perform action on the objects which are not otherwise visible uniquely to the tool.

Environment Support: Having a highly developed image based processing; it can support any graphic based environment.

Scripting Language: It supports highly evolved languages like C#, python, java, ruby, perl which provides it an edge over other testing tools. Also, it provides the tester ability to code in his comfort language.

QTP AND Selenium CONTRAST

After considering the benefits of both the tools, we can compare them on the following criteria's:

Table 1: Result of Time and Cost Metric

S. No.	Features	QTP	Selenium
1	Software cost	Licensed and very Expensive, i.e. 8000 USD	Selenium is an open source tool.
2	Script creation time	More time	Lower time due to ease of scripting, better interface and more options to ease the work.
3	Execution time & Running Cost	Execution time and cost is more than selenium	Lower time and minimum cost.
4	Training time & cost	Lower Training time and trainer is to be paid separately as the vendor	Higher training time required.

		does not provide free training.	
5	Experience of coding engineering	Not Much required	Much required
6	Setup cost	QTP is a lot higher	No cost

The following table shows a contrast between QTP and Selenium with points on the scale of 1-10 with 10 being the best based on my experience with the tools:-

Table 2-Comparison of automated testing tool QTP and Selenium

S. No.	Criteria	QTP	Selenium
1	Execution Speed	5.6	7
2	Environments	3	6
3	Browser Supported	3	5
4	Online Support	7	4
5	Coding Languages Supported	4	7
6	Ease Of Learning	8	6
7	Strongly Typed	3	9
8	Training cost	6.6	5
9	Evolution With Age	6	7
10	Integration With QA Tools	5	7

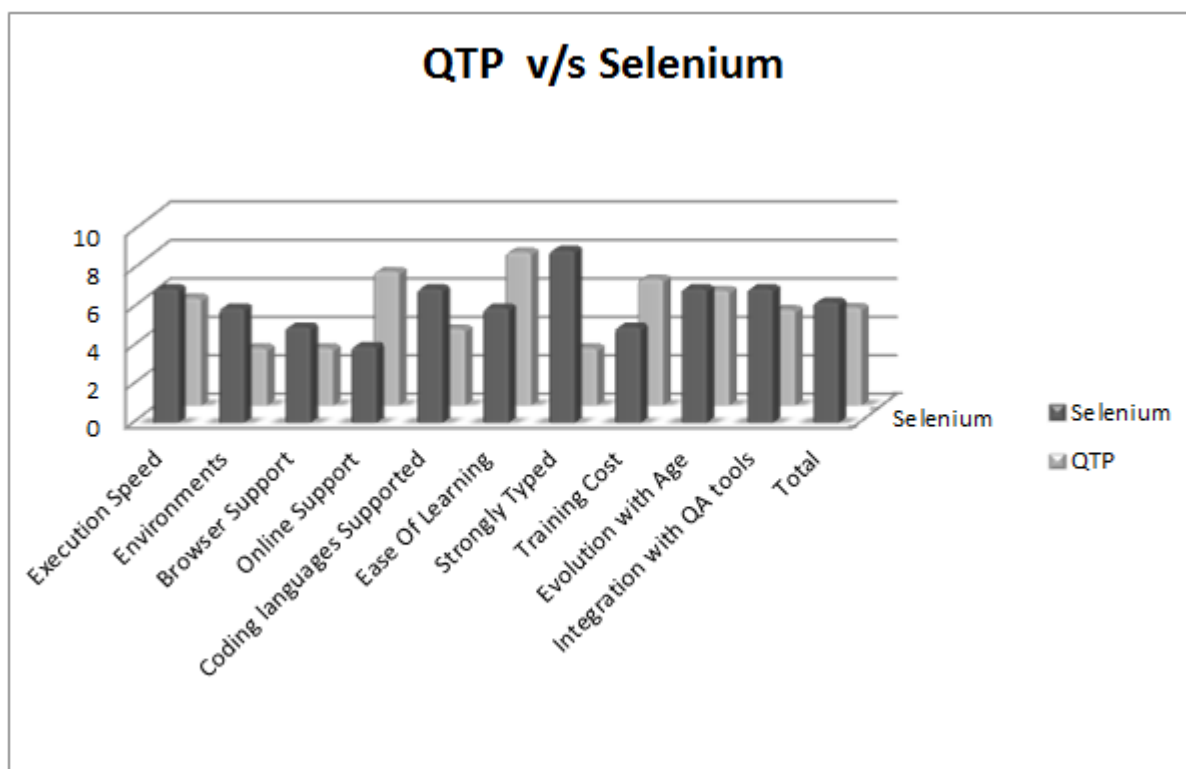


Figure 1: QTP v/s Selenium

4.3 SELECTION OF TOOL BETWEEN QTP AND Selenium

Software testing plays a important role in different phases of SDLC. We expect that the client is more interested in speedy and reliable testing.

If the resources are to be trained for both QTP and Selenium, the time and cost would depend on the following factors:-

Project Timeline

QTP's Edge- The availability of already trained resources on QTP will make it easier and faster for the new resources to learn.

QTP's Weakness- Due to the high license cost, it is a huge investment to train the resources when there is no surety of returns. Also, the evaluation version is available only for 30 days which is less to train resources.

Selenium is the open source testing tool which enables the resources good hands on during training. Also, the selenium community available for the training and solutions for this tool. Weakness- The only weakness I find in the product is low consumer base. This leads to unavailability of trained resources which makes the learning for new resources difficult. Final Outcome:

The time required for training resources on QTP is much lower than training them in Selenium.

Project Cost

The initial license cost for QTP is a lot higher than Selenium which makes the project initiate at a higher cost as compared to Selenium. For QTP, the trainer is to be paid separately as the vendor does not provide free training. If already trained resources are used for QTP, the running cost becomes high for QTP than for Selenium.

Final Outcome:

Project cost using Selenium is drastically lower than by using QTP. research, we were comparing two testing tool QTP and Selenium. Coming to the tool selection, Selenium could provide serious cost benefit. Though it loses its edge due to a very strong market hold of QTP due to which, clients do not prefer to experiment on other technologies. Though, initially switching is both costly and time consuming, the switch to Selenium has a lot of long term benefits.

V. CONCLUSION

In this research paper one point is indubitable that automation testing is much more suitable than manual testing. In this research, we were comparing two testing tool QTP and Selenium. Coming to the tool selection, Selenium could provide serious cost benefit. Though it loses its edge due to a very strong market hold of QTP due to which, clients do not prefer to experiment on other technologies. Though, initially switching is both costly and time consuming, the switch to Selenium has a lot of long term benefits.

If a project is already implementing QTP and has no plan to add new modules, which is a least possible scenario, then the switch to Selenium would be unfruitful. But if the project is to be expanded and new licenses for tools are to be procured for the Project, switching to Selenium is highly advisable.

REFERENCES

- [1] Andreas Leitner, Ilinca Ciupa, Bertrand Meyer and Mark Haoward," Reconciling Manual and Automated Testing: the AutoTest Experience", Proc-40th Hawaii international conference on system science pp 261-271, January 2007.
- [2] Sneha Khorria and Pragati Upadhyay , " Performance Evaluation and Comparison of software testing tools", VSRD International Journal of computer science & information Technology Vol. 2 No. 10 October 2012/ 801ISSN No. 2231-2471 (Online)
- [3] Pak-Lok Poon, Sau-Fun Tang, GH Tse, TY Chen, Choc"late A Framework for Specification-based Testing, ACM, Vol 53, No. 4, April 2010.
- [4] Roger Pressman, Software Engineering, A Practitioner's Approach, Sixth Edition, 2005.
- [5] Deepti Gaur, Dr. Rajender Singh Chhillar" How Automated Testing Tools Are Showing Its Impact In The Field Of Software Testing", IJCSMS International Journal of Computer Science & Management Studies, Vol. 12,
- [6] Rakesh Agarwal and Nonika Bajaj "Automation testing- Applying Intelligence", Proceedings of the 2nd National Conference Indiacom-2008 Computing for National Development, February08-08, 2008
- [7] David Crowther, Peter Clarke, Examining Software Testing Tools, Dr. Dobb's Journal: Software Tools for the Professional Programmer, ISSN# 1044789X, Academic Search Premier, June 2005, Vol. 30, Issue 6
- [8] Prof. (Dr.) V. N. Maurya, Er. Rajender Kumar "Analytical Study on Manual vs. Automated Testing Using with Simplistic Cost Model", International Journal of Electronics and Electrical Engineering ISSN: 2277-7040 Volume 2 Issue 1 (January 2012)