International Journal of Electrical and Computer Engineering (IJECE)
Vol. 9, No. 5, October 2019, pp. 4473~4478
ISSN: 2088-8708, DOI: 10.11591/ijece.v9i5.pp4473-4478

Comparative review of the features of automated software testing tools

Heidilyn V. Gamido, Marlon V. Gamido
Tarlac State University, College of Computer Studies, Tarlac City, Tarlac, Philippines

Article Info

Article history:

Received Jan 4, 2019
Revised Apr 11, 2019
Accepted May 8, 2019

Keywords:

Automated tools
Software development
Software engineering
Software testing
Testing tools

ABSTRACT

Software testing is considered to be one of the most important processes in software development for it verifies if the system meets the user requirements and specification. Manual testing and automated testing are two ways of conducting software testing. Automated testing gives software testers the ease to automate the process of software testing thus considered more effective when time, cost and usability are concerned. There are a wide variety of automated testing tools available, either open source or commercial. This paper provides a comparative review of features of open source and commercial testing tools that may help users to select the appropriate software testing tool based on their requirements.

Corresponding Author:

Heidilyn V. Gamido,
Tarlac State University, College of Computer Studies,
Tarlac City, Tarlac, Philippines.
Email: htvgamido@tsu.edu.ph

1. INTRODUCTION

Software development involves processes such as software programming, documenting, testing to develop a functional application finally. To check whether the software meets the user requirements and to deliver a functional application, software testing becomes a critical process [1-3]. It plays a significant role in the successful implementation of any system application.

Testing an application can either be done manually or be automated using software testing tools. A tester acts as an end-user and tests the correct behavior of most of all features of the application for manual testing. Because of this, manual testing is time-consuming and demanding, and it does not always get rid of all bugs effectively. It is an excellent choice for smaller companies that do not have sufficient financial resources for automated systems.

Automated testing addresses the challenges presented by manual testing. Automated testing allows tester the ability to create repeatable and reusable test scenarios. These test scenarios can then be executed as often as needed. Also with the increasing complexity of software development, it demands that software team use automated software testing tools to test the quality and functionality of the application. An automated test is more effective when time, cost, and usability are concerned.

There are a wide variety of automated testing tools available in the market, either open source or commercial. There are software tools that only perform a specific kind of testing and limited to specific type of language. While those software testing tools that support a wide range of applications, with better features and functionality may require additional costs. To know the differences of one from the other would allow the user to determine the right testing tool for their environment. This study aims to make a comparative analysis of available software automated testing tools by comparing the features of each tool in terms of type of testing, software support, licensing and cost and others.

2. REVIEW OF RELATED LITERATURE

Systematic literature review and practitioner survey discusses that the benefits of test software include reusability, repeatability, and effort saved in test executions [4]. The paper also supports the superiority of test automation when several regressions testing rounds are needed. Among the identified limitation of automated testing tools mentioned is the high initial cost in designing test cases, the initial investment in the purchase of the test automation tool and may require additional investment in the need to train staff. Test automation is more reliable, programmable, reusable, comprehensive and maintainable, saves money and time in the long run, has greater test coverage and is faster than human interactions compared to manual testing [5].

Efficiency and accuracy are the two main benefits of automated testing over manual testing [6]. Using automated test scripts yields a better return on equity over manual testing. Software testing is one of the most and time- consuming processes in software development [7]. Because of this, software developers have become increasingly interested in attempting to optimize testing to reduce development costs. A discussion of test metrics which serves an an important indicator of the effectiveness of software testing process was presented in [8]. These include organization metrics, project metrics, process metrics, product metrics and static and dynamic metrics. Organization metric refers to usefulness in overall project planning and management. Project metrics are useful in monitoring and controlling specific project. Process metrics use some test process while product metrics relates to a specific product like a compiler for a programming language. Static metrics are those computed without having to execute the product while dynamic metrics require code execution. Testing metrics are used to improve software productivity and quality. A methodological framework in [9] was used to evaluate testing techniques or tools that can be applied in various case studies. The implementation of the case studies is measured from their effectiveness, efficiency, and user satisfaction. Software test tools help software developer to examine software bugs, verify functionality and ensure the reliability of the software developed [10]. Various testing tools were grouped based on their types of applications and were analyzed by their cost and features. Software testing is still one of the most widely used approaches for checking and improving the quality of a software application [11]. One of the contributions in testing research is automated test input generation. Aside from this, there are new frameworks for test execution which promotes shorter cycles in the testing process. In [12], the paper experimented automation testing using three different software. Although automation have an initial high implementation and maintenance cost, test automation can give remarkable remarks in the long run when it is rerun multiple times. Because of repeatability and reusability of test scenarios, test automation increases the overall effectiveness of the testing process. For mobile testing, identify the mobile platform support, the lead time for the new OS, test coverage, text support (languages), test workflow, scripting capabilities, price and service support are needed when considering the tool to guide software developers and researchers [13].

3. RESULTS AND ANALYSIS

For this paper, the automated software testing tools chosen are the following:

a. Selenium IDE. Selenium is an open source and portable testing tool to test web application that supports a different browser, platforms, and operating system [14].
b. QTP/UFT (Quick Test Professional/Unified Functional Testing). UFT(formerly QTP) is a graphical interface record-playback automation tool [15].
c. TestComplete. TestComplete is an application that helps automate software quality tests for websites, web applications, and Windows desktop applications [16].
d. Ranorex. Ranorex is a graphic user interface automation framework used for testing desktop, web-based, and mobile applications.
e. Load Runner. Load Runner is a software testing tool developed by Hewlett Packard and is used to test applications, check system behavior and performance under load.
   Watir. Watir is a simple, flexible and open source tool used for automation testing.
   SahiPro. Sahi Pro is an open source cross platform testing tool used for web application.
   SoapUI. SoapUI is an open-source testing tool used for web service testing.

3.1. Evaluation Parameter

In order to make sense and study the different features of automated testing tools, we need to identify the features to be used for the analysis of distinguishing similarities and differences of each tool. According to [5], that in selecting the best tool among automated testing tools, we can consider these key points: Support to platforms and technology, flexibility for testers of all skill levels, feature-rich but easy to

create automated test and automated tests that are reusable, maintainable and resistant to changes in the applications user interface. Table 1 shows the tool features for comparing software testing tools to address the research purpose of the paper.

Table 1. Tool features of software testing tools

Criteria used	Definition	References
Cross Platform	Operating system supported	[17-18]
Cross Browser	Browser tools supported	[17]
Record Playback	Ability of tool to record scripts	[17, 19, 20]
Script Language	Programming languages used to edit test scripts or for the creation of testing scripts	[17, 18, 20]
Ease of learning	How easy the tool is used	[17, 19, 21]
Data driven	The ability of tool to reduce efforts like making it possible to make the scripts access the different sets of input data from external source like data tables, excel sheets	[17, 20]
Programming skills	Programming skills needed	[17, 19]
Report generation	How result is represented	[17, 19-21]
Cost	Whether free or licensed	[17, 19-20, 22]
Function	Type of testing supported	
Others	Advantage/ Disadvantage/ Comment	

3.2. Comparative Review

Table 2 shows the comparative review of the selected automated software testing tools based from the evaluation parameter used.

Table 2. Comparative review of automated software testing tools

	Selenium IDE [15, 18-19, 21, 23-27, 28]	QTP/UFT [15, 17-19, 22, 26, 28-31]	TestComplete [17-20, 22, 31-32]	Ranorex [17, 19, 29, 31-32]	Watir [17, 19, 24-25]	Load Runner [23, 27, 28, 30]	Sahi – Pro [17, 19, 27, 31]	SoapUI [19, 20, 33]
Developer	Jason Huggins	HP	Smartbear	Ranorex GmbH	Bret Pettichord and Paul Rogers	HP	Tyto	Smartbear
Cross platforms	Windows Linux, Unix, Mac	Windows	Windows	Windows	Windows, Mac, Linux	Windows, linux mac	Windows, linux, mac	Windows
Cross-Browsers	Chrome, Firefox, Opera, IE	Chrome, Firefox, IE	Chrome, Firefox, Opera, IE	Chrome, Firefox, Opera, IE, Netscape, Safari	Chrome, Firefox, Opera, IE, Safari	Any browser	Any browser	Chrome, Firefox, IE
Record-Playback	Support	Support	Support	Support	Support	Support	Support	Support
Script-language	Java Ruby, python, php C#, .net	Vb script (supports java,.net, Delphi)	Vbscript, C#, jscript C++,delphi	Vb script but supports .net C++,C#, python	Ruby but supports Java, C#,.net	C, Vb, Vbscript, C#, Javascript	Javascript but supports Ruby	Java
Ease of learning	Experience needed	Easy to learn	Experience needed	Easy to learn	Easy to learn	Experience needed	Easy to learn	Easy to learn
Data-driven framework	Excel, Csv Xml	Excel, Text file Db files, Xml	Csv, Excel, Sql	Csv, Excel, Sql	Xml, Excel	Csv	Csv	Excel, Xml
Programming skills	Needs to have programming skills	Partial Quite easy to edit, navigate, parametize	Needs to have programming skills	Partial	Partial	Partial (script can be complex and difficult to understand	Partial	Partial

Table 2. Comparative review of automated software testing tools (c*ontinue*)

	Selenium IDE [15, 18-19, 21, 23-27, 28]	QTP/UFT [15, 17-19, 22, 26, 28-31]	TestComplete [17-20, 22, 31-32]	Ranorex [17, 19, 29, 31-32]	Watir [17, 19, 24-25]	Load Runner [23, 27, 28, 30]	Sahi – Pro [17, 19, 27, 31]	SoapUI [19, 20, 33]
Report generation	Html	Html Xml -gives executive summary of test, gives statistics in the form of pie charts	Html, xml	Html -with executive summar, with graphs for faster and better comparison of defects in every run	Html, xml	Does not provide graphical representation of results	Html	Html Easy to use reporting functionality
Cost	Open source	Licensed	Licensed	Licensed	Open source	Licensed	Freeware/ Licensed	Freeware/ Licensed
Function	For web application -not for performance testing	Web testing -regression, unit, distributed, manual	Functional,13 -regression, unit, distribute, load, web, manual	GUI test for web and mobile based application	Web testing	Load testing and performance testing	Web test automation	Functional testing tool, regression and load testing
Others	-No error handling and no database testing -no facility to feature network effect and no resource monitoring capability -needs to exert hard to do the testing with respect to database applications -slower execution speed	-utilizes more of CPU and ram -works well with database application -expensive, huge investment and training cost is separate	Supports also mobile (ios) applications -very fast and uses less cpu and ram than UFT -no graphical representation of result -higher computing resources than selenium (whole pc monitoring)	Low consumer base Unavailability of trained resources	Doesn't seem to be widely used	Integrated with SHUNRA Virtual Enterprise Suit to get network effects - works well with database applications	-good reporting -proxy-related issue during installation -control browsers	High cost and maintenance

4. CONCLUSION

The research is about the analyzation of the different features of automated testing tools such as Selenium, QTP/UFT, TestComplete, Ranorex, Watir, Sahi, and SoapUI. Since automated software testing has become a necessity for companies, based on the discussion, we can choose which among the automated testing tool is to be used for a certain type of testing purposes. In selecting tools, if the project cost is to be given higher consideration, open source tools such as Selenium is the better option. If the availability of support, ease of learning, report generation are to be considered, licensed tools such as QTP/UFT is a good option. For future works, other automated tools can also be included in the study, including their response time.

REFERENCES

[1] Jupudy, N. Saraf and R. Manjula, "Comparative Analysis of Model Based and Formal Based Software Testing Methods," *International Journal of Advanced Research in Computer Science and Software Engineering,* vol. 6, no. 3, pp. 66-70, March 2016.

[2] Fachrurrazi, "The End User Requirement for Project Management Software Accurcay, "*International Journal of Electrical and Computer Engineering (IJECE),"* vol. 8, no. 2, pp. 1112-1121, April 2018.

[3] S. K. Pandey and K. Mustafa, "A Comparative Study of Risk Assessment Methodologies for Information Systems," *Bulltin of Electrical Engineering and Informatics,* vol. 1, no. 2, pp. 111-122, June 2012.

[4] D. M. Rafi, K. R. K. Moses and K. Petersen, "Benefits and Limitations of Automated Software Testing: Systematic Literature Review and Practitioner Survey," *2012 7th International Workshop on Automation of Software Test,* 2012.

[5] Lavkesh and H. Mittal, "Comparative Analysis of Automated Functional Testing Tools," *Journal of Network Communications and Emerging Technologies (JNCET)*, vol. 6, no. 6, pp. 50-53, June 2016.
[6] D. Asfaw, "Benefits of Automated Testing Over Manual Testing," *International Journal of Innovative Research in Information Security (IJIRIS)*, vol. 2, no. 1, pp. 5-13, January 2015.
[7] C. M. Thomas, "An Overview of the Current State of the Test-First vs. Tst-Last Debate," *Scholarly Horizons: University of Minnesota, Morris Undergraduate Journal*, vol. 1, no. 2, pp. 1-6, 2014.
[8] V. Verma and S. Malhotra, "Applications of Software Testing Metrics in Constructing Models of the Software Development Process," *Journal of Global Research in Computer Science*, vol. 2, pp. 96-98, May 2011.
[9] T. E. Vos, B. Marin, M. J. Escalona and A. Marchetto, "A Methodological Framework for Evaluating Software Testing Techniques and Tools," *12th International Conference on Quality Software*, pp. 230-239, 2012.
[10] J. B. Prajapati, "Significance of Testing Through Testing Tools: Cost Analysis," *e-Journal of Science & Technology (e-JST)*, vol. 3, no. 8, pp. 106-114, 2013.
[11] A. Orso and G. Rothermel, "Software Testing: A Research Travelogue (2000–2014)," *FOSE'14,* 2014.
[12] D. Kumar and K. K. Mishra, "The Impacts of Test Automation on Software's Cost, Quality and Time to Market," *7th International Conference on Communication, Computing and Virtualization 2016*, pp. 8-15, 2016.
[13] N. H. Saad and N. S. A. Abu Bakar, "Automated Testing Tools for Mobile Applications," *The 5th International Conference on Information and Communication Technology for The Muslim World (ICT4M)*, 2014.
[14] Vishawjyoti and S. Sharma, "Study and Analysis of Automation Testing Techniques," *Journal of Global Research in Computer Science*, vol. 3, pp. 37-43, 2012.
[15] H. Kaur and G. Gupta, "Comparative Study of Automated Testing Tools: Selenium, Quick Test Professional and Testcomplete," *International Journal of Engineering Research and Applications*, vol. 3, no. 5, pp. 1739-1743, 2013.
[16] T. Bharti and E. Vidhu dutt, "Functionality Appraisal of Automated Testing Tools, "*International Journal of Computer Science Trends and Technology (IJCST)*, vol. 3, no. 1, pp. 129-134, 2015.
[17] M. Monier and M. M. El-mahdy, "Evaluation of automated web testing tools," *International Journal of Computer Applications Technology and Research*, vol. 4, no. 5, pp. 405-408, 2015.
[18] N. Tiku, A. Parkar, I. Rampurawala and A. Menon, "Performance Analysis of Automation Testing Tools for WPF Application," *International Journal of Research (IJR)*, vol. 2, no. 6, pp. 540-544, 2015.
[19] V. Malik and M. Gahlan, "Comparative Study of Automated Web Testing Tools," *International Journal of Latest Trends in Engineering and Technology (IJLTET)*, vol. 6, no. 3, pp. 367-374, 2016.
[20] Meenu and Y. Kumar, "Comparative Study of Automated Testing Tools: Selenium, SoapUI, HP Unified Functional Testing and Test Complete," *Journal of Emerging Technologies and Innovative Research (JETIR)*, vol. 2, no. 9, pp. 42-48, 2015.
[21] T. Sheth and S. K. Singh, "Software Test Automation - Approach on evaluating test automation tools," *International Journal of Scientific and Research Publications*, vol. 5, no. 8, pp. 1-4, 2015.
[22] M. Kaur and R. Kumari, "Comparative Study of Automated Testing Tools: TestComplete and QuickTest Pro," *International Journal of Computer Applications*, vol. 24, pp. 1-7, 2011.
[23] K. Kamble and J. Kharade, "Quantitative Analysis of Manual and Automation Testing and Comparative Study of Selenium and Load Runner Automated Tseting Tools," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 4, no. 5, pp. 9577-9583, 2016.
[24] R. Angmo and M. Sharma, "Performance Evaluation of Web Based Automation Testing Tools," *5th International Conference on the Next Generation Information Technology Summit*, pp. 731-735, 2014.
[25] I. Singh and B. Tarika, "Comparative Analysis of Open Source Automated Software Tools: Selenium, Sikuli and Watir," *International Journal of Information and Computation Technology*, vol. 4, pp. 1507-1518, 2015.
[26] R. Bhavithra, G. Lavanya and D. Thiruveni, "Comparative Study of Automated Testing Tools: Selenium and Quick Test Professional," *International Journal of Computer Systems*, vol. 03, no. 02, pp. 117-120, 2016.
[27] J. Singh and M. Sharma, "A Comprehensive Review of Web-based Automation Testing TOols," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 3, no. 10, pp. 9255-9259, 2015.
[28] Shaveta, S. Kumar, Nitika and Snehlata, "Comparative Study of Automated Testing Tools: Quick Test Pro and Load Runner," *International Journal of Computer Science and Information Technologies*, vol. 3, no. 4, pp. 4562-4567, 2012.
[29] A. Jain, M. Jain and S. Dhankar, "A Comparison of RANOREX and QTP Automated Testing Tools and their impact on Software Testing," *International Journal of Engineering, Management and Sciences (IJEMS)*, vol. 1, no. 1, pp. 8-12, 2014.
[30] R. N. Khan and S. Gupta, "Comparative Study of Automated Testing Tools: Rational Functional Tester, Quick Test Professional, Silk Test and Loadrunner," *International Journal of Advanced Technology in Engineering and Science*, vol. 03, no. 01, pp. 167-172, 2015.
[31] J. Singh and M. Sharma, "Performance Evaluation and Comparison of Sahi Pro and Selenium Webdriver," *International Journal of Computer Applications*, vol. 129, pp. 23-26, 2015.
[32] N. Dubey and S. Shiwani, "Studying and Comparing Automated Testing Tools: Ranorex and TestComplete," *International Journal of Engineering and Computer Science*, vol. 3, no. 5, pp. 5916-5923, 2014.
[33] S. Hussain, Z. Wang, I. K. Toure and A. Diop, "Web Service Testing Tools: A Comparative Study," *International Journal of Computer Science*, 2013.

BIOGRAPHIES OF AUTHORS

Heidilyn V. Gamido is a graduate of Doctor in Information Technology at Technological Institute of the Philippines, Quezon City under the CHED K-12 Transition Program Scholarship. She obtained her Masters of Engineering major in Information and Communications in 2006 at Pai Chai University, Daejeon South Korea on a scholarship. She finished her BS Information Technology at Saint Louis University, Baguio City Philippines in 2002. She is an Associate Professor of Tarlac State University - College of Computer Studies and designated as the Director of the Management of Information Systems Office. Her research interests include data security, image processing, and information system.

Marlon V. Gamido is the Dean of College of Computer Studies in Tarlac State University. He holds a degree in MS in Information Technology from Hannam University, Daejeon Korea on CHED Scholarship. He is currently pursuing his PhD in Educational Management at TSU. He is also a registered Electrical Engineer. His research interests include security and project management.