International Journal of Innovative Research in Information Security (IJIRIS)
Issue 1, Volume 2 (January 2015)
ISSN: 2349-7017(O)
ISSN: 2349-7009(P)
www.ijiris.com

Benefits of Automated Testing Over Manual Testing

Daniel L Asfaw
PhD Student in Computer Science and Software Testing
Inter-University Program
Universidad Azteca and Universidad Central de Nicaragua (UCN)
Mexico City – Managua Nicaragua

Abstract -- There is no enough sound and solid scientific researches expounding the benefits of using automated scripts over manual testing (Samuel R. , 2014). The ones available out there are virtuously promotional trailers made for marketing drive (Udin, 2014). This dissertation is made to fill up this gap. To this end, a comparative analysis of the test results achieved from both automated and manual testing have been piloted. Complementary research inputs such as data collected thru questionnaire, interview and group discussion have also been analyzed and synthesized to back up the outcome. Unified Functional Tester (UFT) is used to build test artifacts and execute automated scripts. The conclusion exhibits that using computerized scripts might offer considerable returns in terms of acquiring enhanced efficiency and enriched accuracy over manually testing, provided that the test is labor intensive, time taking and reoccurring.

Key words -- Automated testing, manual testing, accuracy, and efficiency.

I. INTRODUCTION

Things change promptly in the IT industry. New technologies and versions pop up almost on everyday bases (Samuel R. , 2014). As software testing is part of this change, the need to coup up with the change, in fact, without compromising quality, is one of the need of the day (Samuel S. , Computer Scienties, 2014).

Of course, the quality assurance process has to shadow that same speed to address the pressing demands of the needs of existing customers (UK Essays, 2014). Consequently, in attempt to pace with contemporary needs in the industry, quite a lot of investments on automated testing has evolved (Rudd, Team Lead, Quality Assurance at WESTA, 2014).

This paper is about the benefits of such testing over the manual. It has five brief sections. The first one, which is this, give a brief introduction about the topic: research purpose, hypothesis, statement of the problem, limitation of the study. The second part reviews relevant literatures and current industry practices. Then, the third part details with the methodology used in the study, while the forth one discusses the findings of the research. Finally, it wraps up with a brief conclusion and recommendation.

STATEMENT OF THE PROBLEM

Ensuring a quality user experience across the wide variety of handsets, networks and platforms would be key to increased and continued growth (Galde, 2014). Following same, a lot of demand has been observed for comprehensive service portfolio, which can address generic as well as unique requirements with rapid assessments (Roper, 1989). In this regard, a huge human labor has already been involved in the testing industry. And yet the demand does not seem to be fully satisfied (Lin, 2010). As the testing industry become more cumbersome, tedious and arduous, quality would be compromised in its own course of process (Varma, 2014).

This in turn might challenge efficiency and accuracy of the testers involved in the industry (Mall, 2010). That might be one of the causes to initiate the automation industry. Today, we have more test automation tools than ever (Reiner Musier, 2013). There is also a growing demand for quality, in fact, through using automated scripts (Kinfe, Automation Engineer, 2014). Nevertheless, adequate and scientific researches that pinpoints the benefits of automated testing over manual seems to be lacking in the field.

RESEARCH PURPOSE

The main intention of this paper is to find if using automated test script have any advantage over manual testing.

HYPOTHESIS OF THE RESEARCH

This research assumes that automated testing has considerable advantages over manual testing, especially with test cases that are labor intensive and reoccurring by their very nature.

LIMITATIONS OF THE RESEARCH

The term automation invokes different meanings even for members of the software development and testing community (What is automated software testing, 2014). Some think, it is a unit testing. Others see it as a custom developed test script (Varma, 2014). Still others confuse it with performance, stress, and security testing (Rudd, Manual testing, 2014).

This shows that the concept of automation is wide ranging and all inclusive. However, it is not the intention of this study to give coverage to all of them.

© 2015, IJIRIS- All Rights Reserved

Hence, this study is limited to a kind of custom made scripts on the top of QTP v11.5 using VB scripting as a scripting language. In fact, the scripts are arranged in the form of a keyword framework.
There are a lot of diverse automated software testing tools currently on the market (Software, 2014). They are quite numerous and comparable to select from. This study used QTP v11.5 for a couple of reasons. First of all, QTP has over 50% of the market coverage in the industry (Gurru99, 2014). This gives higher usability to the outcome of the research. Of all the researcher has better understanding with HP tools compared to any other in the market.

This might designate one of the limitation of the study in terms of types of test tools. It doesn't cover all, even never attempted to do so. This is so because, it is never manageable to cover all and even if it were the research will lose focus. In fact, to get that level of coverage of the test tools, quite a lot of research have to be involved.

Therefore, the finding might not apply for all types of testing tools. It is limited to the type of test tools that applies to this study. However, research might invoke other researchers in the area.

II. Review of Related Literature and Best Industry Practices

It is difficult, to unceasingly sustain and improve the quality of software systems development by using only manual testers (Why Automation, 2014). This might gradually lead to deficiency of product quality, followed by customer disappointment and amplified inclusive quality costs (Bennet, Manual testers team lead, 2014).

As a solution to this the testing trend is inclining towards automation (Hanip, Complexity of software testing, 2014). It appears that companies have started to believe in the power of using automated scripts over manual testing. (SoftSmith, 2012).
According to a market search report conducted on 699 respondents from 504 companies, the need automation seems to intensely expand over the last few years (Reiner Musier, 2013). The majority of these companies are primarily located in North America and Europe. In the said research most of them are reported to have an annual revenues greater than $500 million USD,

This research reports that of the 504 responses related to investment, 204 (34.3%) have planned additional investment in automated testing. They do this by minimizing their manual involvement in regression testing, in particular regression for risk. They do this in an attempt to enhance the software quality assurance process of their respective companies in the next 12 months. Only 24.6% said, they have not planned investment (Reiner Musier, 2013).

The current rapid adoption rate of test automation is likely to continue through 2013 (Reiner Musier, 2013). However, the report doesn't say if these companies have a well-defined research carried out to lead them to this end. Nevertheless, (Kolawa & Huizinga, 2007) claims that automated software testing provides a means to reduce errors, cut maintenance and overall software costs.
Automation is not applicable for all sorts of testing. First of all, all testing types are not repetitive, time consuming and labor intensive. Besides, automation may not work for all test scenarios. Even if it does, it may not be a good candidate (Bennet, Manual testers team lead, 2014). Hence, it is important to determine which test type will produce the most benefit from automation (Samuel R. , 2014). Good candidates of automated testing might need to be very repetitive, reoccurring, time consuming, and labor intensive, and error prone if tried by manual means (Samuel S. , Computer Scienties, 2014).

They might also need to be very large in terms of length, needs more accuracy since it is mission critical, and need stronger security (Hanip, Complexity of software testing, 2014). Unless otherwise, a test case meets these standards automation might not be helpful. This is so because LOE would overwhelm ROI (Kinfe, Automation Engineer, 2014). Additional criteria might include the following: it evaluates high risk conditions, the test is costly to do it manually, the test is impossible to do manually, and the test is data intensive (Rudd, Manual testing, 2014).
The objective of automation is not to eliminate testers. Instead it should aim to help them to make better use of their time (Samuel R. , 2014). If automation is not well supported by the manual testers themselves, it might cause a kind of deprecation in the quality of the end product. This deprecation might not been foreseeable, but will come to attention during its course of time. It should not be overseen that there are circumstance were we have some tests that require human intervention. Nevertheless, if the automation leverage time and money while augmenting quality and accuracy, it would be injudicious to stick to manual testing (Udin, Software testing , 2014).

© 2015, IJIRIS- All Rights Reserved

The objective of automated testing is to simplify as much of the testing effort with a minimum set of scripts (Kolawa & Huizinga, 2007). If regression testing consumes a large percentage of a quality assurance (QA) team's resources, for example, then this process might be a good candidate for automation (Samuel R. , 2014).
Automated testing should target on rendering better accuracy, better efficiency and better quality assurance at the end (Bennet, Manual testers team lead, 2014), This in return has to render better achievement, high profitability, and better satisfaction by all stakeholders, even by the manual testers themselves (Samuel S. , Computer Scienties, 2014).
Automation takes the use of a third party software to generate the required script to regulate the execution of an automated tests (Harrold, 1966). He further indicated that the script could be produced in various programming languages depending the tools used to automate it. IBM rationales usually use Java, while HP products depend on VB script. Open source tools such as Selenium could take Java, Rube and other common languages in the market (Rahman, 2014).

Most of these tools have common or identical features, such as record and paly back, expert view where the programming test place, object repository where the objects reside, object inspector which inspects or spy the property of the object, object inserter which inserts objects in to the script, object identification (Hanip, Complexity of software testing, 2014).
Verification points ensure if there is no regression from one build to the next (Samuel R. , 2014). It also help check if the system is still healthy after having served for a considerable span of time (Rudd, Manual testing, 2014). Hence, the more the diversity of verification point exist in the automation tool, the quality would become (Hanip, Complexity of software testing, 2014).
As to the automation tool, picking up one over the other is not any easy task. Nevertheless, for the purpose of this study, HP products (QTP v11.5) has been used for a couple of reasons. First and for most, it is widely used and more popular (Guru 99, 2014). Secondly, the researcher has easier access and familiarity to these tool.

III. METHODOLOGY

This article, as a part of an ongoing PhD dissertation in computer science and software testing in an Inter-University Program at Universidad Azteca and Universidad Central de Nicaragua (UCN) in Mexico City – Managua Nicaragua, profoundly depends on the data collected form test results executed both manually and using computerized scripts. To this end various test scenarios have been implemented.

These scenario helps to better understand how the application would perform in the hands of an end user (Varma, 2014). This gives the tester the advantages to undertake a role that signifies the client. By so doing the tester could verify and validates the various functionalities that might exist in the application under test. To the purpose of this study, four recurrent scenarios have been selected: count maximum number of links and validate if the expected number matches with the actual, click on a link and validate if the new window opens up and verify if the expected text exists, click on a link and validate if the new window opens up and verify expected snapshot matches with actual, and insert data into textboxes and submit same by clicking a submit button and verify if the expected output matches with the actual.

The above mentioned scenarios and their attached operations are not believed to be exhaustive and conclusive. Nevertheless, they are the most common and recurrent operations that testers face in their daily life of testing (Bennet, Manual testers team lead, 2014). What's more, they are not that complex, and all the testers involved in the study are quite accustomed with such scenarios. Of all, they are good enough to meet the objective of the study.

Two web based applications are selected for the purpose of this study. These applications are used only to execute the three test cases created on the basis of the two requirements and the aforementioned scenarios. The first requirement is traced to the first two test cases and has one to many relationship with them. The second requirement, on the other hand, is traced only to the third test case and exhibits merely a one to one relationship. For the first two scenarios an Indian newsletter entitled The Indian Express is used. It is located at http://indianexpress.com/. Below is the snapshot of the header part of the newsletter. The other web based application used to execute test cases fulfilling the last three scenarios is Mortgage Calculator located at http://www.mortgagecalculator.org/.
Based on the above mentioned scenarios test cases have been built using most common type of test case template. This test case template contains basic ingredients of test case such as: test case ID, requirement ID, number of iteration, number of build, type of test, test preconditions, test steps, verifications points, expected results, and the likes. These are more or less what a test case has to embrace (Kinfe, Automation Engineer, 2014). Both manual testes and automation engineers have been given these test cases. Both manual testers and automation engineers were required to execute the steps in the respective test cases by hands and using automated scripts, respectably. They had to answer the questions in the control checklist questionnaire proximately before and after conducting each test case.

© 2015, IJIRIS- All Rights Reserved

ISSN: 2349-7017(O)
ISSN: 2349-7009(P)
www.ijiris.com

The control checklist questionnaires help manual testers to log such information as test start time, test end time, expected outputs, and other related questions presented in the form of multiple choice and fill in the blank space. The answers are used as inputs to the research and are analyzed using SPCC to objectively address the research questions. The operations that the testers are supposed to execute against the applications are clearly stated in the test steps.
Requirements that will encompass the aforesaid functionalities have been designed. Proper review processes were conducted on these requirements. These reviews led to the creation of test plan, and test scenarios. All the reviews were conducted by both the manual and automation testers who participated in the study. Following the reviews the researcher built the test cases that were meant to address the reviewed test scenarios and conditions. These test cases have also gone through review peer review, walk through, and inspection process. Side by side, control questionnaire was designed to collect the results from both manual and automated testing. This control questionnaire inquires the tester to register the start time, end time and the finding in the process of the test execution. For instance, if the test case is about counting the maximum number of links on a certain web page, the tester has to register his/her time of start and of fish, and also answer the multiple choice question that gives him the options to select the maximum number of the links s/he discovered.

Only two expertise were involved to take care of the automation side of this research. These automation engineers had developed computerized scripts on the top of QTP v11.5. These scripts have been reviewed a couple of times and based on the feedbacks from the review, they have been fine tuned to best serve the intention of the study. Apart from the test plan, the requirements, and other test artifacts which were prepared for the manual testing were used in the automation side as well. It had its own type of automation test plan peculiarly prepared to it based on the other artifacts used for the other type of testing. The other responsibility of these engineers were to execute the scripts based on the test cases that were made available to them. Since the script is made to give detailed reports as to how long the testing time took to run the test, the engineers were not required to log start and end time of the testing. However, they had to log the total test time along with other information required in the control questionnaire. As was in the manual testers, the data collected from the control questionnaire were analyzed and systemized using SPSS and other statistical tools.

IV. Discussion and finding

This part comparatively analyzes and discusses the benefits of using computerized scripts over manual testing. To this end three test cases have been implemented. The first two help contrasting efficiency by analyzing the best test time achieved by both manual and automated meanness. The last test case is meant to analyses accuracy by the way of measuring the number of times testers were able to correctly count the links existing in the homepage of the Indian Express out of 100 attempts.

4.1. Efficiency: Automated testing verses manual testing

Table 1

Test case	Manual best test time	Automation test time
Test case 1	300 seconds	20 seconds
Test case 2	459 seconds	30 seconds

As seen in Table 1, the best time the manual testers could register is 300 seconds to run test case one, while the automated script was able to perform same in 20 seconds. The variation between them is 280 seconds. In a similar token, the best time registered for test case two was 459 seconds when run by the manual testers, and 30 seconds when by automation. Again the difference here is 429 seconds. We observe pairs of relationship for the time spend to execute these test cases (manual verses automation). Based on these order pairs, we can craft a liner equation with the following slope:

$$m = \frac{(y1 - y2)}{(x1 - x2)}$$

Where: m = slope,
yi = time spent for manual testing for test case i
xi = time spent for automated testing for test case

© 2015, IJIRIS- All Rights Reserved

ISSN: 2349-7017(O)
ISSN: 2349-7009(P)
www.ijiris.com

$$m = \frac{(459 - 300)}{(30 - 20)}$$

$$m = 15.9$$

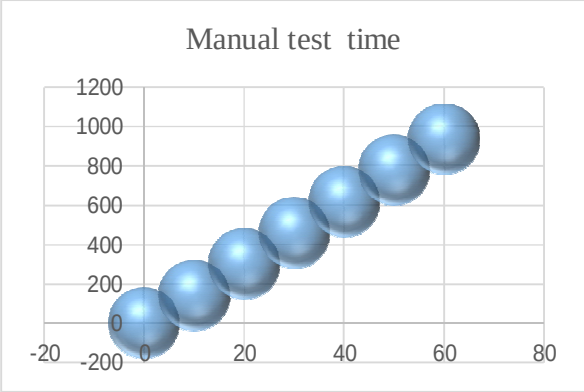
$$y - y1 = m\,(x - x1)$$

$$y - 300 = 15.9\,(x - 20)$$

$$y = (15.9\,x - 318) + 300$$

$$y = 15.9\,x - 18$$

Figure 1



In Figure 1 above the "X" axis represents automation test time in seconds while "Y" axis does manual test time. According to this graph, the test case that takes 1000 seconds in manual testing, it is took only 60 seconds when automated test case is used. Likewise, the test case that took 500 seconds in manual testing, it took 40 seconds in automation. On the other hand as we approach 0 point in the graph, the advantage of automated testing gets slow down. This designates that fact that, as the manual test case execution time takes longer, automated scripts turns out to be more profitable and advantages in that it gives better timing and efficiency.

For example based on the above scenario, if it takes 57222 seconds (158.95 hrs.) to manually execute a given test case, the corresponding time in automation would be 3600 seconds (1hr). This is just a onetime scenario. Suppose this test case runs every week for one full year, it will take 8265.4 hrs. (2975544 second) to execute it manually, and 52 hours in automation. The leveraged time due to accelerated efficiency produced by implementing automated test scripts would, therefore, be 8213.4 hrs. If this is projected for a period of 10 years, the outcome would arithmetically progress to a substantial number of hours.

However, this doesn't necessarily signify that the return of investment would be equivalent to the number hours leverage by accelerating the efficiency. This is so because manual and automated testers do not make equal in the job market. The market tends to pay automation tester a little higher than the manual. Supposing the manual tester get "X" amount of money per hour and the automation tester makes double of that, in the above scenario the manual tester would get 8265.4X pay per year while the automated tester might make 104X pay. The amount of money saved from using automated test case, therefore, would be 8101X.

From this premises, it is possible to that using automated scripts would help accelerate efficiency and achieve better return of investment. On the contrary, if the test case is not repeating, not time consuming and labor intensive, manual testing might be a better option over automated scripts.

© 2015, IJIRIS- All Rights Reserved

Hence, in regression for risk, automated scripts is exceedingly recommended over manual testing for several reasons. First of all, regression testing is repetitive. It reoccurs periodically. Secondly, it involves huge and multiple test cases. Third, it is labor intensive and time taking. Last but not least, it is costly.

Therefore, using automated scripts would make it more efficient, less labor intensive, and less costly. Putting all together, it would render good return of investment.

4.2. ACCURACY: AUTOMATED TESTING VERSES MANUAL TESTING

Computing accuracy tells how close a set of measurements are to an accepted reference value (Samuel S. , Computer Scienties, 2014). Nonetheless, calculating accuracy is a pain in the neck for most of the manual testers, whom I spoke with about the matter. As an automation engineer, I, myself, have quite a lot experiences where I was able to observe test cases which were passed by manual testers getting a fail result when executed thru automated scripts.
Several reasons could be assumed for this. First of all, the requirements are either untestable or challenging to handle them by manual means. On the top of that, manual testers are error prune when it comes to challenging and complicated test cases such as tallying the number of links in a web page.

For the purpose of this particular research a single requirement is designated to measure accuracy. This requirement requires the overall count the links on the home page of the Indian Express Newsletter.
A test case has been designed to handle the test steps and verification points consistent to this requirement. 50 manual testers are involved in executing this test case. Each one of them have to do it 100 times to measure the percentage of the corrected attempts. These manual testers are sorted out into five groups according to the level of their skill sets. The same test case has been run for same number of times using automated script as well.
Table 2

A = Group average for correct answer out of 100 attempts, B = Group average years of experience, C = Mean of 'A' - 'A', D = Mean of 'B' - 'B'

Where the second row from the last represents automation output as one group

A	B	C	D
2.9166	1309.583	24.2349	1439.806
4	1581.25	23.1515	1168.139
5.5714	1809.25	21.5801	940.1388
20.421	2853.75	-20.421	-2853.75
30	7847.5	-30	-7847.5
100	1095	-100	-1095
27.1515	2749.388833		-7153.17

As seen in table 2 above, testers with no and/or indirect relevance of academic skill sets along with 3-5 years of experience were able to get the correct count with an average score of 2.9 and 4 times, respectively, out of 100 attempts. Testers in the last three groups with direct relevance of academic skill sets but different years of experience got the expected answer with average scores of 5.57, 20.4, and 30 times, again out of 100 attempts. This indicates the fact that when academic relevance increases accuracy get better. However, academic relevance by itself does not necessarily guarantee accuracy at all.

© 2015, IJIRIS- All Rights Reserved

ISSN: 2349-7017(O)
ISSN: 2349-7009(P)
www.ijiris.com

The automated test execution result, on the other hand, designates that the correct and expected answer has been achieved 100% in all attempts. This might lead to a conclusion that using automated test scripts could give better accuracy over manual testing.

V: CONCLUSION AND RECOMMENDATION

The finding of this research demonstrates that automation testing has at least two main benefits over manual testing: efficiency and accuracy. As seen in the research, when efficiency increases, cost gets down, and better return of investment is inevitable.

On the contrary, the research demonstrates that automated testing is more efficient, accurate and cost effective over the manual testing. The variation seen in the level of accuracy between the best maul and automation test time is 70%. This shows how manual testing is error prone compared to automated testing. In fact, mistakes committed during manual testing process are minimized as academic skill sets augments, and yet it never reached precision.

Hence, based on these findings, the research would like to suggest the following recommendation for all concerned stake holders:
Regression testing, in particular, regression for risk, is a kind of reoccurring testing type. Hence, the researcher recommends concerned stakeholder to consider regression for risk for automated scripts for better efficiency and accuracy.
Using automated test scripts yields better ROE over manual testing. Hence, the researcher advices the stakeholders to mind their way in the world of software testing, if they have, in any way, anything to do with regression testing, in particular, regression for risk.

References

Bibliography

Beizer, B. (1990). *Software Testing Techniques.* Boston: International Thompson Computer.

Bennet, L. (2014, May 27). Manual testers team lead. (D. Asfaw, Interviewer)

Buzzle. (2014, May 27). Retrieved from Different Types of Application Software : http://www.buzzle.com/articles/different-types-of-application-software.html

DeMillo, R. A. (1991). Software engineering. *Proceedings of the 13th international conference on Software engineering*. 13th international conference on Software engineering.

Guru 99. (2014, May 07). Retrieved from Introduction to QTP: http://www.guru99.com/quick-test-professional-qtp-tutorial-1.html

Hanip, A. (2014, 05). Professor at PNT Institute of Technology . (D. Asfaw, Interviewer)

Harrold, G. R. (1966). *Analyzing Regression Test Selection Techniques.* New York.

Hetzel, W. C. (1988). *The Complete Guide to Software Testing, 2nd ed*. New york: Wellesley.

Howden, W. E. (1987). *Functional program Testing and Analysis.* McGraw-Hill.

Huizinga, D. (2007). *Automated Defect Prevention: Best Practices in Software Management.* Wiley-IEEE Computer Society Press.

Katrick. (2012, May). QTP online. *Vedio* . QTPILearn.

Kinfe, S. (2014, May 7). Automation Engineer. (D. Asfaw, Interviewer)

Kolawa, A., & Huizinga, D. (2007). Automated Defect Prevention: Best Practices in Software Management. *Wiley-IEEE Computer Society Press*.

L.White, K. A. (1997). A firewall approach for the regression testing of object-oriented software. *In Proceedings of 10th Annual Software Quality Week,* (p. 27).

© 2015, IJIRIS- All Rights Reserved

Lin, X. (2010). *Regression Testing in Research and Practice.* Lincoln: Computer Science and Engineering Department University of Nebraska.

Mall, S. B. (2010). *Regression Test Selection Techniques: A Survey.* Kharagpur, India: Dept. of Computer Science and Engineering IIT.

Maneging complexity . (2014, May). Retrieved from Maneging complexity : http://www.economist.com/node/3423238

Mohammed, N. (2014, May). Professor at PNT Institute of Technology . (D. Asfaw, Interviewer)

Musa., J. D. (1997). Understanding software testing. *Proceedings of the 1997 international conference on Software engineering.*

Myers, Glenford J. (1979). The art of software testing. In G. J. Myers, *The art of software testing.* New York: Wiley. Retrieved from 1979. , Publication info: New York : Wiley, c1979. ISBN: 0471043281.

Nair, R. P. (2008 , June). *Types of testing*. Retrieved from http://rajeevprabhakaran.wordpress.com

PC Magazine Encyclopedia. (2014). Retrieved from Definition of:test scenario: http://www.pcmag.com/encyclopedia/term/52773/test-scenario

Peleska, J. (2008). *A Formal Introduction to Model-Based Testing Part I: Exhaustive Testing Methods.* Verified Systems International GmbH and University of Bremen.

Perlis, A. J. (2009, 10 15). *The Origin of Software Bugs*. Retrieved from The Origin of Software Bugs: http://www.informit.com/articles/article.aspx?p=1398775

Quality Digest. (2014, May). Retrieved from Defintion of quality: http://www.qualitydigest.com/magazine/2001/nov/article/definition-quality.html

R. Gorthi, A. P. (2008). Specification-based approach to select regression test suite to validate changed software. *In Proceed- ings of the 2008 15th Asia-Pacific Software Engi- neering Conference*, (pp. 153–160).

Rahman, M. (2014 , June 2). PhD Questionnier . (D. Asfaw, Interviewer)

Reiner Musier, P. (2013). *Trends in Automated Testing For Enterprise Systems* . WorkSoft.

Roper, N. P. (1989). *Understanding Software Testing*. New York: John Willey & Sons.

Rothermel, G. a. (1996). *Analyzing regression test selection techniques.*

Rudd, B. (2014, May). Team Lead, Quality Assurance at WESTA. (D. ASfaw, Interviewer)

Samuel, R. (2014, July 20). Automation Engineer. (D. Asfaw, Interviewer)

Samuel, S. (2014, May 03). Computer Scienties. (D. Asfaw, Interviewer)

SoftSmith. (2012, May). *QTP.* Retrieved from Open Mentor: http://learn.openmentor.net:8080/openmentor/view/user/recweb/testAutomationQTP.jsp

T. Graves, M. H. (2001). *A. Porter, and G. Rothermel. An empirical study of regression test selection techniques. ACM Transactions on Software Engineering and Methodology.*

Test Condtion. (2014, May 3). Retrieved from http://www.allinterview.com/showanswers/67697.html

Testing Articles. (2014, May). Retrieved from What is Automated Software Testing: http://www.testingperformance.org/definitions/what-is-automated-software-testing

Total number of websites. (2014, June 1). Retrieved from http://www.internetlivestats.com/total-number-of-websites/

© 2015, IJIRIS- All Rights Reserved

UCS Libraries . (2014, May 17). Retrieved from Organizing Your Social Sciences Research Paper : http://libguides.usc.edu/content.php?pid=83009&sid=616083

Udin, M. (2014, May). Manual Tester at Colabera. (D. Asfaw, Interviewer)

UK Essays. (2014, May 9). Retrieved from Customer satisfaction and quality assurance: http://www.ukessays.com/essays/business/customer-satisfaction-and-quality-assurance.php

V. N. Mauraya, a. R. (2009). *Analytical Study on Manual vs. Automated Testing Using with Simplistic Cost Model* . India : Aligarh UP.

V.Kharytonov. (2012, December 15). *Software Measurement: Its Estimation and Metrics Used* . Retrieved from http://it-cisq.org/software-meausrement-estimation-metrics/

Verma, R. (2014, April 27). Scientist/ Manger, Quality Assurance at Global ISI. (D. Asfaw, Interviewer)

W.E. Wong, J. H. (1997). A Study of Effective Regression Testing in Practice. *Proc. Eighth Int'l Symposium Software Reliability*.

Web Definition. (2014, May). Retrieved from http://www.google.com/search?client=safari&rls=en&q=test+scenario+definition&ie=UTF-8&oe=UTF-8#q=manual+testing+definition&rls=en

What is a Test Case. (2006, February 27). Retrieved from What is a Test Case?: http://testingsoftware.blogspot.com/2006/02/test-case.html

What is automated software testing. (2014). Retrieved from http://idtus.com/what-is-automated-software-testing/

What is scope and limitation of the study. (2013, May 12). Retrieved from http://www.reference.com/motif/Education/what-is-scope-and-limitations-of-research

White, H. L. (1989). Insights into regression testing. *In proceedings of the conference of software maintenance*, (pp. 60-69).

Why outomation. (2014). Retrieved from http://www.ranorex.com/why-test-automation.html

Wikipedia. (2014, May 07). Retrieved from Internet-Speed Development: http://en.wikipedia.org/wiki/Internet-Speed_Development

Wikipedia. (2014, May). Retrieved from Software Testing : http://en.wikipedia.org/wiki/Software_testing#Testing_methods

Xest, M. N. (2010). An automated framework for regression testing of embedded soft- ware. *In Proceedings of the 2010 Workshop on Em- bedded Systems Education, WESE* , (p. 8).

Zafar, R. (2014, May 6). *What is software testing? What are the different types of testing?* Retrieved from What is software testing? What are the different types of testing?: http://www.codeproject.com/Tips

Ziegler, J. G. (1989, May). Ziegler, J., Grasso, J. M., and Burgermeister, L. G. 1989. An Ada based real-time closed-loop integration and regression test tool. Proceedings of the Conference on Software Maintenance. *An Ada based real-time closed-loop integration and regression test tool*.

© 2015, IJIRIS- All Rights Reserved