Automated Software Testing: An Insight into Local Industry

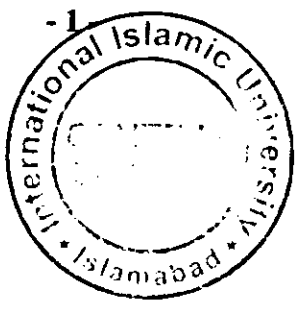
Submitted by:

Zara Shaheen
281-FBAS/MSSE/F09

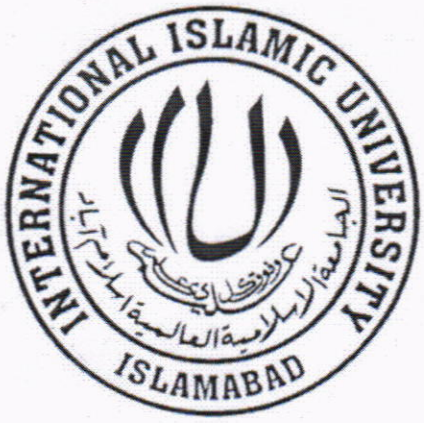
Supervised by:

Dr. Abdul Rauf

Department of Computer Science and Software Engineering
Faculty of Basic and Applied Sciences
International Islamic University Islamabad
(June 2012)



Automated Software Testing: An Insight into Local Industry



Submitted by:

Zara Shaheen
281-FBAS/MSSE/F09

Supervised by:

Dr. Abdul Rauf

Department of Computer Science and Software Engineering
Faculty of Basic and Applied Sciences
International Islamic University Islamabad
(June 2012)

MS
005.14
ZAA

-1 Computer Software

APPROVAL

SUBJECT: External APPROVAL OF THE RESEARCH THESIS "Automated Software Testing: An Insight into Local Industry"

It is certified that we have read this research thesis report and have fully evaluated the research undertaken by **Zara Shaheen** Registration No. **281-FBAS/MSSE/F09**. This research thesis fully meets the requirements of Department of Computer Science & Software Engineering and hence, the International Islamic University, Islamabad.

Supervisor:

Dr.Abdul Rauf
Assistant Professor,
Department of Computer Science & Software Engineering,
Faculty of Basic and Applied Sciences,
International Islamic University, Islamabad

Internal Examiners:

Dr. Zunera Jalil
Head of the Department,
Department of Computer Science & Software Engineering,
Faculty of Basic and Applied Sciences,
International Islamic University, Islamabad

External Examiners:

Dr. Abdul Basit
Assistant Professor,
Department of Engineering and Information Technology,
Foundation University Institute of Engineering and Management Sciences, Rawalpindi

Dedication...

To my family especially my mom

A dissertation Submitted to

Department of Computer Science and Software Engineering,

Faculty of Basic and Applied Sciences,

International Islamic University, Islamabad

As a Partial Fulfillment of the Requirement for the Award of the

Degree of *Masters of Science in Software Engineering (MSSE).*

Declaration

I hereby declare that this thesis ***"Automated Software Testing: An Insight into Local Industry"*** neither as a whole nor as a part has been copied out from any source. It is further declared that I have done this research with the accompanied report entirely on the basis of my personal efforts, under the proficient guidance of my teachers especially my supervisor *Dr. Abdul Rauf.* If any part of the system is proved to be copied out from any source or found to be reproduction of any project from any of the training institute or educational institutions, I shall stand by the consequences.

Zara Shaheen

Acknowledgement

In the name of Allah, the most passionate and the most merciful whose blessings made it possible for me to complete this research work. It is a great pleasure for me to complete it successfully. It is all because of **Almighty Allah's** guidance that made me so able.

I owe my deepest gratitude to my thesis supervisor Dr. Abdul Rauf whose brain was behind the theoretical raw idea of this research work. Without his guidance and supports it was not possible to complete this research work. He has made available his support in a number of ways. His commendable advices, sincere supervision and gracious attitude are worth mentioning and for which I am extremely grateful.

I would also like to show my gratitude to my parents for their continuous support for the completion of my work. Especially my mother, whom is responsible for my intellectual curiosity and is always there for me in my hard time, this is the reason I am on this stage today.

L astly, but by no means least, I would like to acknowledge my friends for their moral support.

F or errors and inadequacies in this research work, I accept the responsibility.

Zara Shaheen

Project In Brief

Project Title:	Automated Software Testing: An Insight into Local Industry
Undertaken By:	Zara Shaheen 281-FBAS/MSSE/F09
Supervised By:	Dr. Abdul Rauf
Start Date:	April 10, 2011
Completion Date:	December 20, 2011

Abstract

Software testing is the most expensive task. It takes more than 50% of software development cost. It is also laborious and time consuming activity [2, 13]. Software testing can be accomplished by two ways i.e. manual and automated. Manual software testing incurs more cost, resources and time. It can affect the quality of the software product. Software quality can be improved through testing. As software quality is very important for success of the system. So automated testing can be used in some part of testing work to improve the effectiveness, quality and to reduce time and testing resources. Thus, it can reduce software testing cost. There are certain benefits and challenges associated with the use of automated software testing.

Awareness of advantages and challenges of automated software testing is necessary before using them in the projects. Therefore, we conducted an industrial survey of the IT market of Pakistan to find out the benefits and challenges in the implementation of automated software testing. Moreover we also found out testing tools available in literature and their implementation in industry. Then, we analyzed the gaps between literature and practices.

Through the results of the study, it is concluded that in local industry of Pakistan "Automated Software Testing (AST) "has been introduced for the last few years. And some gaps have been identified between literature and practices. There is need of further research in automated software testing. The challenges should be solved to make AST process more effective.

Table of Contents

List of Figures

Chapter 1

INTRODUCTION

1.1 Background of the Research

Software systems are becoming increasingly important in all areas of life. The development of a software product quality is not an easy task. One of the key challenges that software industry has been facing for last few decades is to develop a quality software product within time and budget [34]. The quality of software product can be ensured through testing.

Software testing plays an important role to improve the quality of the software product and is one of the most important phases of development life cycle [1]. Today software testing has become an essential part of any organization delivering some products to customers. However, it is very time consuming and costly activity [37]. It has been proved through literature that for software testing at least fifty percent of the software development cost is being paid [2, 4, 35, 36]. Software testing should be cost effective and efficient process of any competitive organization.

There are two ways of performing software testing; manual and automated [7]. Manual testing is very laborious and time consuming activity. It takes too many resources for testing the software. Hence it incurs more cost. Due to time, cost and resource problems extensive software testing is impossible [3, 14]. Therefore it affects the quality of the

software. When quality of software suffers it leads to its failure as well as of Software Company that develop that software.

Efficiency of software has been improved through automating some parts of the testing work, in which software tester can concentrate on significant software features or more complicated cases, putting down recurring tasks to the test automation system. Through automation human resources may be used more efficiently, which accordingly may throw into more comprehensive testing or savings in testing process and on the whole development budget [3].

The two most important goals of software engineering research are; attenuation of cost and betterment in quality of software product [4]. And software testing acquires most of the software development cost and helps in improvement of quality. Thus introduction of automation infrastructure in testing can meet these two objectives [4]. Automated testing decreases time, cost and resources [16]. Hence it improves the quality of software by testing it thoroughly. It reduces cost and improves quality because it does lot of testing in less time [5].

Automated testing is generally appropriate to recurring jobs such as unit testing and regression testing; in which when changes are made tests cases are executed [3]. The predominantly tasks that are automated in testing are test case creation, execution and confirmation of testing outcomes. Automated software testing is not appropriate for non-recurring tasks. For non-recurring tasks manual testing is more appropriate because development of automated software for testing is very extensive task and it is only possible when a task is repeated several times [6]

Hence the most important perceived reason of the adoption of automated testing is quality improvement through better test coverage and thorough testing of software can be done in less time [7, 39]. There are also several challenges related to automated software testing and some researches dream 100% automated testing but Bach analyze that 100% automated testing is not possible because all automation at the end requires human intervention for example to analyze the results, maintenance of automation cases etc [15]. Automated testing is being adopted by Pakistan software industry. Some companies keep continue the application of automated testing and some companies do not find it useful. Hence it is very beneficial to get the experience of software industrialist regarding the pros and cons of automated testing and to share those facts.

The study focuses on the literature survey reporting the issues and benefits of automated testing and the testing tools mentioned in the literature. We have uncovered the benefits and challenges of automated testing through literature and also uncovered the tools related to different phases of testing mention in literature with their benefits and issues. To validate the results with respect to practice, we have conducted a survey regarding the issues and benefits of automated testing and adoption of automated testing tools in software development companies of Pakistan. Hence the similarities and differences of benefits and challenges of automated testing and automated tools in literature and experienced by professionals have been explored.

1.2 Research Statement

This research is intended to deal with Automated Software Testing benefits, challenges

and tools. The main question to address is "what is the current position of Pakistan's IT industry in Automated Software Testing?"

Followings are the list of sub-questions to support main research questions:

i. What are the Benefits of automated software testing with respect to literature and Practices?

ii. What are the Challenges of automated software testing with respect to literature and Practices?

iii. Which software testing tools are generally utilized in Pakistan IT industry?

The challenges and benefits of automated testing were identified by conducting a Literature Review of automated software testing up to 2011.

The practice of automated testing adoption considering challenges and benefits in Pakistan IT market was addressed by conducting a survey in software industry of Pakistan. The professionals were asked to fill the questionnaires related to automated software testing practices and their benefits and challenges. Based on the frequency of reported issues and benefits, a list of issues and benefits has been generated.

1.3 Objectives of the Research

The research study is aimed to achieve following set of objectives to address the issues and problems mentioned before:

i. Review the existing literature on Automated Testing regarding their benefits and challenges plus automated testing tools

ii. Locate the status of practices of Automated Testing in Pakistan IT market.

iii. The literature survey (LR) published up to 2011 was considered to report the benefits and challenges of automated software testing practices plus testing tools.

iv. A survey of Pakistan IT market was conducted to get the apparent benefits and challenges of automated software testing and reported automated testing tools, based on their experience.

v. Based on the findings of LR and Survey, the differentiation in literature and practices is reported and analyzed.

1.4 Motivation

It is mentioned earlier that software testing has a very important role in the SDLC. Software quality can be improved through comprehensive and through testing. But it is not possible with manual software testing, as it incurs huge cost, time and effort. Therefore introduction of automation in software testing can overcome this issue. Software quality may improve through automated testing with less cost, time and effort.

Similar studies have been conducted on software testing and automation testing i.e. in Australia, Alberta, and Finland [51, 5]. They conducted industrial survey and ascertained information about software testing methodologies, testing tool, metrics and standards. Their studies concluded that organization don't use new testing methodologies because they have lack of expertise in new testing methodologies and high cost of the testing tool [51].

As there are many studies related to the software testing but we have found that there no such study which investigate and compare the software testing automation practice in Pakistan.

The main objective of this study was to explore the status of testing automation in the software industry. The second objective was to find out the gap between literature and industry practices. This research study provides the information to new researchers who want to work in this area. They can locate the most obstructing issue in adoption of test automation and help to solve these problems through further research.

1.5 Significance of the Research

The important thing for accomplished this research is that

I. This study is anticipated to serve as a support or direct Software testing teams aiming to apply the automated software testing.

II. They can choose the automated software testing by evaluating reported issues, characteristics or benefits and recommendations for their use.

III. Hence it helps the software organizations to choose the best and appropriate practice or testing tool with respect to their projects.

1.6 Thesis Outline

Remaining of the thesis is organized as follows:

Chapter 2: The second chapter provides a comprehensive literature review of Automated Software Testing benefits, challenges and tools.

Chapter 3: Third chapter states problem statement and the existing concepts used in this research work.

Chapter 4: Fourth chapter discusses the research methodology. In this chapter; detail about the research methodology has been defined.

Chapter 5: The fifth chapter provides the results and analysis of the research study.

Chapter 6: The conclusion has been provided in sixth chapter of this research thesis.

The contributions of this research work have been discussed in a neutral way. A discussion has been made on how the contributed work answer the research question posed in the first chapter.

Chapter 2

LITERATURE REVIEW

2.1 Introduction

This chapter outlines and describes the Automated Software Testing (AST), the benefits and challenges of AST. It also presents the overview of various software testing tools related to web application testing. Next it describes the features, application and use to software testing tools with regard to testing types and levels.

2.2 Automated Software Testing

Automated Software Testing is to use the software to manage the execution of tests, comparing actual results with estimated results, the establishment of test conditions and test control and test reporting functions [20]. In general, test automation involves automating a manual process already in place that uses a formal testing process. By definition [3,12] Automated Software Testing implies automation of testing activities, including development and execution of test scripts, verification of test requirements, and the make use of automated testing tools. According to them [3,12], causes of using Automated software testing includes, such as manual testing is time consuming, and the test automation improves efficiency particularly in regression testing, where tests are executed repeatedly after formulating modifications to the software. Manual testing is a feasible way out when automation is not cost effective. Software Testing

automation, where appropriate, to reduce the perceived software testing costs, according to [2, 40] it is estimated that it may take more than 50% of the total software development effort.

A case study has been conducted by Karhuet.al [7]; they found different factors that can affect the test automation. These factors are related to the test automation that which type of application can use testing automation, or in the other sense what are the limitations of the test automation. They chose different organization development as a case study; the organizations were from different domains of the software like telecommunication, generic software development etc. The only limitation of this study is number of case organization units. If variety of case organization from other domain was also covered, then it could reveal more details about test automation.

Khaled M et.al categorized and distributed software testing tools with respect to testing methods. They concentrated on these applications i.e. web application, application software and network applications. And they analyzed that which type of software testing have limited and no tools [9].L.S. Chin et.al presented a report related to information assembled about software testing practices for computer and engineering science. They provided brief introduction of software testing and survey of software testing tools for scientific applications written in FORTRAN [8].

2.3 Benefits of Automated Software Testing

Automated Testing is an important field of concern in ongoing testing research. In order to increase the degree of automation, either by developing best practices or the

generation of test inputs, or to find support for the procedures to automate the testing process [14].

The two most important goals of software engineering research are attenuation of cost and betterment in quality of software product [4]. And software testing acquires most of the software development cost and helps in improvement of quality. Thus introduction of automation infrastructure in testing can meet these two objectives [4].Hence the most important perceived reason of the adoption of automated testing is quality improvement through better test coverage and thorough testing of software can be done in less time [7]. Automated testing decreases time, cost and resources [16]. Hence it improves the quality of software by testing it thoroughly. It reduces cost and improves quality because it does lot of testing in less time [5].

Automated testing is generally appropriate to recurring job such as unit testing and regression testing; in which when changes are made tests cases are executed [3]. Automation leads to repetition and reusability of existing test scripts, with each new build. Build is (i) a new version of the product with new functionality in the existing one, (ii) carry out some modifications in existing functionality, or (iii) remove or fix bugs in existing functionality of the product. All these three cases involve regression testing. As regression testing involves verification that adding new features, bug fixes and modifications did effectively fix the basic problem without initiating any new bugs. These all entails recurrence and reuse of existing test scripts with release of new build [11]. In manual testing, such tests are very time taking, expensive, lengthy, and tedious.

This work is not cherished by the software testers. Automated regression testing can be efficient and effective than manual testing [11].

Reusability is the important property of the testing automation, it makes it efficient. The reusability not only reduces effort to create new tests but also the time and cost for its development and maintenance [10, 38].

Automated Software Testing is quick and its execution is done without human involvement. Absolute regression testing, improve test coverage are also the strengths of the automation. Furthermore an automatic classification, exposure of test results, released the tester to compact with core or other important tasks. It has the ability to test features of the application that were complex to test in manual testing [11].

2.4 Challenges Faced by Automated Software Testing

Automated Software Testing has various challenges besides its benefits. These challenges can be associated to the automation or limitations of the tools. Purely automation is not supportive to accomplish improved coverage but in the selection of test cases human participation is required [7].

The major disadvantage of automated testing is associated to cost which includes implementation cost, training cost and maintenance cost. Implementation cost consists of direct investment cost, additional implementation time and human resource cost [7].Implementation of automated testing is expensive and time consuming activity. Automated testing also involves regular maintenance which incurs huge cost. Promoting members of staff to implement new testing practices is also very significant. Automated Testing necessitate testers to acquire training before using the new testing practices. If

maintenance of automated testing is overlooked, then revision of whole automated test suite can be very costly than manual execution of the tests. Implementation cost and maintenance cost are directly connected with each other. If reduction of maintenance costs is considered during design of automation, implementation cost may increase and vice versa [10].

The usage of AST also affects the type of the tested product. Automated testing is most appropriate to generic and autonomous to intermediary software products. Because test scripts and test automation requirements are then simpler to specify. Customized software products have more changeability and interfaces to intermediary systems and consequently, it is more complex to capture test specifications as the system environment and functionalities differ from customer to customer [7]. Various testing activities are hard to implement, particularly if they need wide spread area familiarity. Domain acquaintance is noticed in the testing of customized software products. So little human involvement in automated testing make it possible but elevated human participation is dishonored [7].It has also been observed that the entire automated test suites necessitate human involvement, if only to analyze the results and fix not working tests[15, 50].

Another major challenge being faced by test automation is technological changes; it can be either in development and application infrastructure. If there are constant changes in technology then automation in testing may possibly supe rseded. On the other hand, accommodating technological changes will suffer huge maintenance cost,this is undesirable.

One of the ideas of software testing is 100% automatic testing [14]. But according [7, 15] interpretations, it cannot be accomplished. Automated software testing is not appropriate for non-repeatable testing activities. For non-repetitive activities manual system testing is more appropriate. Implementation of testing automation is very prevalent activity and viable only in repetitive tasks. [7, 14].

It is observed that [11] the implementation of automated testing is a high threat and a high investment project. Automated testing cost estimation models, [21] carry judgment in the trade-off between manual and automated testing.

2.5 Automated Software Testing Tools

There are many Automated Testing tools with respect to web applications. Tools are mostly related to functional, regression, unit, load, stress and performance testing. Each tool has different features and application.

WASATT (Web Application Scenario Automated Test Tool) is commonly used for functional web testing. It has one drawback that its method cannot precisely model some properties of web application such as dynamic semantics of web application [22]

They [12] proposed a list of features that they consider that any automated testing tool should hold. A tool which can assist analysis testers and developers to devise and develop their test cases, execute them with least human interference and to produce reports to establish the outcomes of the test cases. They proposed a tool named AsT, which is a browser-based functional automated testing tool. AsT tool is still way following the existing automated tools in terms of features. However the foremost benefit of this is the

time for creation and maintenance of test cases is relatively smaller compared to the other existing tools. It was primarily designed for HTML and PHP scripts [12].

WALTy (Web Application Load-based Testing Tool), is a set of tools that consent to the performance analysis of web applications as they are superficial by the end users. The proposed approach is based on a workload classification created from information taken out from log files. The workload is created by using of Customer Behavior Model Graphs (CBMG). This graph based formalism consent to illustrate the activities of usual users of the web application under test. One of the most common detractors to this approach is that mock workload produced by web stressing tools is far-off from being realistic. The use of the CBMGs might be useful to overcome this detractor. The key extent of WALTy is to present an intent methodology based on the generation of a representative mark out. It is composed by a set of tools using Customer Behavior Model Graphs for distinguishing web user profiles and for creating virtual users' web sessions [23].

They [24] introduced a testing tool named **WaRR** that trace and repeat connections between user and web application with high trustworthiness. It has two main components i.e. WaRR recorder and WaRR replayer. (i) WaRR Recorder is set in in the web browser that accesses the user interactions. (ii) WaRR replayer uses better developer -specific web browser that enables realistic imitation of the users. WaRR has low runtime overhead during recording, offers high reliability and can locate bugs in real contemporary web applications. They describe two custom scenarios that assist the developers to get better reliability of web applications .i.e. testing web application against realistic human errors. and the other one is the generation of user experience reports. It has also been mentioned

that this tool facilitated in finding out bugs in widely used application such as Google sites. WaRR uses a novel architecture, with the recording feature being a primary part of a web browser [24].

This design architecture has five advantages [24]:

i. WaRR have access to each click and keystroke of users. Therefore it has high reliability recording.

ii. This recorder necessitates no adaptation to web application, it is easy to use.

iii. It has right to use actual HTML code

iv. It can simply be extensive to record different sources of non-determinism.

v. It is platform independent.

There are some limitations of WaRR in addition to its enormous benefits e.g. as it records every keystroke consequently it has way in to sensitive user data like passwords etc. It has no ability to control pop-ups. It can be insufficient for reproducing bugs involving simultaneous users because it offers single user outlook of how a bug was triggered. And it cannot handle the environment it runs in. WaRR have to be complemented by server side aid [24].

There are a few limitations of Selenium according to [24]: WaRR presents superior recording reliability in contrast to existing tools. Existing records and replay tools such as Selenium has squat reliability. Selenium IDE give way partial user interactions. It is not succeeded to generate event handlers related to user actions and it have to be explicitly installed by users. In contrast to this WaRR is a high-reliability tool that records interfaces between user and modern web application [24].

TestInv-P passive testing tool is a web security testing tool and its practice is in particular case of 'e-tourism' web application. This is developed by Montimage. It consents to automated analysis of the captured traces to establish if the given invariants are satisfied or not. It is written in C language and uses the XML2 library to infer the traces and the invariants [28].

TestInv-P follows these steps [28]:

i. First of all, characterize invariants (these are amalgamation of conditions that must be respected by system) and data of significance. This process can be made by the expert of the system under test

ii. The invariants are written in XML format to easily interpret by humans and software.

iii. The defined invariants are then verified with respect to their XML schema.

iv. Finally incarcerate communication traces using Wireshark (this is a network packet sniffing tool) and examine them using TestInv-P tool.

TestInv-P tool is user friendly or learn even by non initiated developers but the main problem is regarding the invariants selection, adaption and usage. These can only be formed by security and formalism expert. But this job can be simpler if we pass on to other security SHIELDS models (like the mitigations of misuse cases). Furthermore the results of the tool appear to be relatively hard to analyze and some well-known susceptibility eradication strategies require to be added to assist the developer task. A user interface for the tool will noticeably aid developers to better employ the tools results (hence the presentation of the results will be better). On the other hand, the tool

formulates users more industrious in finding security defects and avoiding then possible attacks [28].

It is reported [25] that there is no such testing tool which fully tests the web application. So, they proposed a reusable architecture to create web application testing by extending a well estimated architecture and applying some design patterns. The architecture includes six subsystems, and the testing activities (e.g. Test case generation) can be accomplished with the collaboration of these subsystems. To reveal the usability of this architecture, a prototyped web application testing environment is constructed. But there are still some problems revealed like various web applications are associated to databases [25].

There are various Load testing tools that make the load testing easier, faster and accurate .i.e. **Apache JMeter** [26, 29] is an open source Java application developed by Apache Software Foundation. It is designed to load test functional activities and determine performance. JMeter as formerly designed for testing web applications But since then many new features were added therefore it can be used for example to test performance both on static and dynamic resources. The main disadvantage of JMeter is inadequate support for test case recording. It utilizes proxy which can only trace HTTP while HTTPS is carried only in so called, spoofing mode. However, there are other tools which can be used in combination with JMeter and provide HTTPS test case recording [26]

Badboy is [26, 30] a web application functional testing tool. Badboy incorporates with JMeter by permitting exporting test scripts to JMeter file format. Unluckily, there are some limitations of such integration because both JMeter and Badboy features do not

accurately equivalent and in some cases it is not likely to export items directly. In addition, although JMeter embodies some diverse results illustrations, from tables to graphs, results analysis is inferior in contrast to business testing tools. Charts are very confusing and they have no clear terminology. As JMeter is an open source, free and flexible testing tool, therefore it is considered as a great load testing tool [26]

The **Grinder** is also an open source testing tool for load testing and performance measures. [26, 31] its scripts are written in Python programming language. The Grinder is a Java load-testing framework making it easy to manage actions of a test script in many procedures across many machines, via a graphical console application. The most prominent feature of Grinder is that it has generic approach, which performs load testing of any application that has Java API. But the major limitation of this tool is that it does not provide the means for distributing the agents so they must be installed and started manually on all machines [26].

WebLOAD Professional [26, 32] is a RadView's commercial software for performance web applications testing. It supports the generation of infinite load that can be circulated over both Windows and Linux machines. It facilitates complete reporting and analysis starting from runtime to post-test reports [26]. .

NeoLoad [26, 33] is also a commercial load and stress testing tool to measure the performance of the web application. This tool is developed and maintained by a French company Neotys. It provides check of the user response times and infrastructure's statistics. It is compatible with most Web servers and application servers, and it can run

on many operating systems. One of the most important advantages of NeoLoad is easy creation of tests, which can be performed by any non expert [26].

Based on the analysis of load testing tools it was found that commercial tools such as WebLOAD Professional or NeoLoad support all key feature necessary for load testing of web application. Furthermore, open source load testing tools, such as earlier described JMeter and Grinder, are not user friendly. They do not have as good technical support and do not provide as good results analysis as commercial ones do. Yet, these tools are adequate for any Web application load testing although technical expertise is needed to accurately use them [26].

In contrast to JMeter, The Grinder seems to be more appropriate open source tool as it is more flexible and has support for full scenario recording feature. The main advantage of using open source tool is that they are free and their code is available that can be customized according to the requirements [26].

Testing Power Web Stress is a web stress tool for web applications. This provides excessive tests to servers by concerning response time confined during request and response transaction using SDLC. This tool is used to compute the response time taken by a web server to serve client and do comparison of the results with the existing web stress tool, called Pylot [27].

Chapter 3

PROBLEM STATEMENT

As we have mentioned earlier that the aim of this research study is to assist software testers in testing the software projects. They can select the better testing practices with the help of this study.

There are two ways of performing software testing; manual and automated [7]. Manual testing is very laborious and time consuming activity. It takes too many resources for testing the software. Hence it incurs more cost. Due to time, cost and resource problems extensive software testing is impossible [3, 14]. Therefore it affects the quality of the software. When quality of software suffers it leads to its failure as well as of Software Company that develop that software.

As testing automation is an important region of concern in present software testing research. Its obje ctive is to improve the automation, either by developing advanced measures for creating the test inputs, or by finding support procedures to automate the testing method [14]. As software testing is a very important phase of Software Development Life Cycle (SDLC). It incurs most of the software development budget. And quality of the software product is mainly dependent on the testing. One way out for cost reduction and quality improvement is Automated Software Testing.

As there are various research studies and reports on automation of software testing in practice (i.e. [10, 11, 15, 19]. There seems to be short of study that evaluates automation testing practices in local industry of Pakistan. As there are various benefits of Automated

Software testing, which an industry should avail. Besides benefits of AST have certain challenges as mentioned earlier. These should be considered while adopting it in the organization.

The study helps to determine the status of automated software testing in Pakistan IT industry. It also helps the software organizations while adopting AST with pros and cons. For acquiring further empirical awareness about the automated software testing, an industrial survey was conducted. The study was focused on Automated Software Testing benefits, challenges and testing tools.

Similar studies have been conducted on software testing and automation testing i.e. in Australia, Alberta, and Finland [51, 5]. They conducted industrial survey and ascertained information about software testing methodologies, testing tool, metrics and standards. It is concluded from these studies that organizations don't use new testing methodologies because they have lack of expertise in new testing methodologies and high cost of the testing tool [51]. As there are many studies related to the software testing but we have found that there no such study which investigates and compares the software testing automation practices in Pakistan.

The main objective of this study was to explore the status of testing automation in the software industry. The second objective was to find out the gap between literature and industry practices. This study provides the information to new researchers who want to work in this area. They can locate the most obstructing issue in adoption of test automation and help to solve these problems through further research.

For the reliability of the study, a cross check method was employed. Reliability means that measurements are consistent with each other [47]. Literature survey was used to

access the validity of the study. Validity means that we are achieving our objectives [47]. The organizations in the study were selected which are using Automated Software Testing. The data was collected through questionnaire, both online and paper surveys were utilized for acquiring data.

Chapter 4

RESEARCH METHODOLOGY

In this chapter we will describe the research methodology. After that we give detail of the methodology that is made to validate the hypothesis.

4.1 Methodology Overview

Survey was used as a research methodology; it is used when you have to collect the information about people beliefs and feelings about a particular object [52]. Survey is most suitable when you have to assemble information directly from the people, and when you can't study the phenomenon directly. Survey is not just gathering information through the questionnaire etc but it is a complete system for gathering information for description, comparison and explanation of the attitudes, practices and people beliefs [42].

This research was conducted in order to determine status of Automated Software Testing in Pakistan's local industry. The advantages and disadvantages as well as the reliability of this instrument were also part of the objectives. In order to answer these research goals, we chose to obtain the view of software testers in line with this topic. Particularly, more than 100 respondents from 45 companies within Pakistan were randomly selected to make up the sample. Selected participants answered a survey questionnaire structure in Likert scale format [43]. Data gathered from this research instrument were then worked out for analysis. Along with primary data, the study also made use of secondary resources

in the form of published articles and literatures to support the survey results. Figure 4-1 shows the survey process or steps of the survey. It has six steps which have to follow to complete the study.

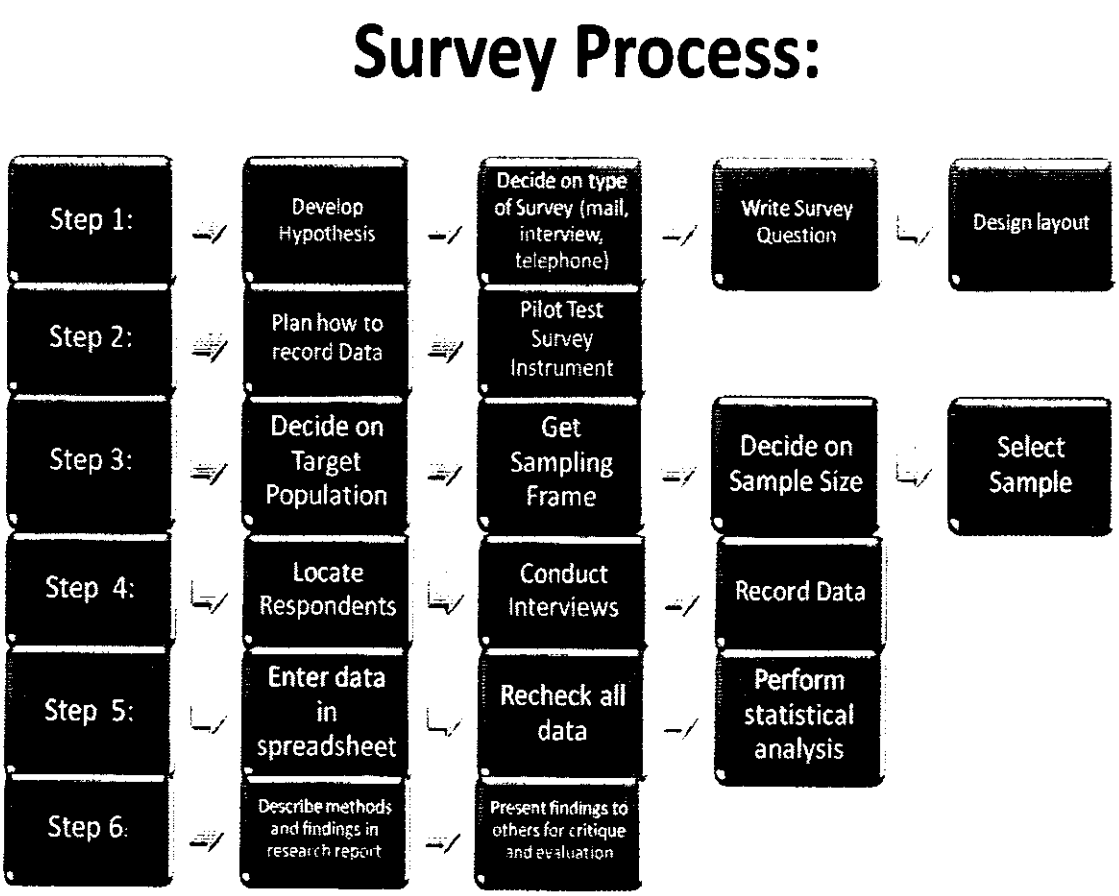


Figure 4-1

4.2 Research Design

As there are two types of survey design [43] i.e. Cross Sectional survey and Longitudinal survey. We chose Cross Sectional Survey that collect data at one given point in time.

This type of research method is use to assemble information about the current existing state [41]. The objective of it is to validate devised hypothesis that refer to the current situation in order to clarify it. The approach is speedy and useful in terms of the economical aspect [42, 48].

In this study, the descriptive research method was used so as to identify the benefits, challenges of Automated Software Testing as well as testing tools during the time of research. We chose to use this research method considering the goals to obtain first hand data from the respondents [43, 49]. The descriptive method is beneficial for the researcher due to its flexibility; this method can use either qualitative or quantitative data or both, giving the researcher greater options in selecting the instrument for data-gathering. The goal of the study was to find out the position of testing automation in the local industry of Pakistan with both the benefits and challenges of automation.

4.3 Instrument

The questionnaire was employed as the core data-gathering instrument for this study. The questionnaire was divided into two major segments: a profile and the survey proper [44]. The profile contains socio-demographic characteristics of the respondents such as role, experience; organization name. The survey proper sections discovered the information regarding automated software testing including its benefits, challenges and testing tools. It comprised of both closed–ended and few open-ended questions. The questions were designed using the Likert-five point Scale [43]. In this survey kind, five options are given for each question. The options correspond to the degree of agreement all respondent has on the specified question.

Scale:

Value	Interpretation
1	Strongly Disagree
2	Disagree
3	Neutral or Do not use
4	Agree
5	Strongly Agree

4.4 Pilot Testing

The validity of the questionnaire was checked through the five respondents. These respondents as well as their responses were not part of the actual study method [45] and were only employed for testing purpose. After the questions have been answered, the respondents were for any propositions or any required improvement to guarantee more perfection and strength of the questionnaire. Then survey questionnaire was revised based on the propositions of the respondents. Irrelevant items were excluded and altered unclear or difficult terminologies into simpler ones in order to ensure comprehension.

4.5 Survey Target Population

The study was using Quality assurance professionals as respondents from 45 organizations in Pakistan in order to gather related data; the descriptive method is then suitable as this can allow the recognition of the similarities and differentiations of the respondents' answers. For this study, two types of data were assembled. These incorporated the primary and secondary data types. The primary data were obtained from

the answers the participants gave during the survey process. The secondary data on the other hand, were obtained from published literatures that were related to 'Automated Software Testing'. With the use of the survey questionnaire and published literatures, this study took on the combined quantitative and qualitative approach of research. By means of using this mutual approach, it was possible to obtain the compensations of both quantitative and qualitative approaches and overcome their limitations.

4.6 Sample Selection

Simple random sampling was made for the sample selection [46]. This sampling method is carrying out where all constituents of a population have an equal chance to become element of the sample. As all constituent of the population have an equal opportunity of becoming a research member, this is said to be the most proficient sampling method. The study covered main cities of Pakistan i.e. Islamabad, Rawalpindi, Lahore and Karachi. In which the organization using automated testing were considered.

4.7 Data Processing and Analysis

After assembling all the accomplished questionnaires from the respondents, the obtained data was stored in the excel spreadsheet. Weighted mean of each item in the questionnaire was calculated to interpret Likert-scale [47]. Different graphs were used to interpret the results i.e. pie chart and columns. From the saved data results were calculated and then on these calculated results, analysis was performed.

Chapter 5

RESULTS AND ANALYSIS

In this chapter we have presented the results of the survey and finally discussed the results.

5.1 Study Respondents

In this research study professionals mostly related to software testing were participated. The roles of participants are i.e. Senior Software Quality Engineers (SSQAE), Testers, Team Lead, Project Mangers, Testing Analysts, Software Test Engineer and Automating Testing Analysts.

Survey Respondents

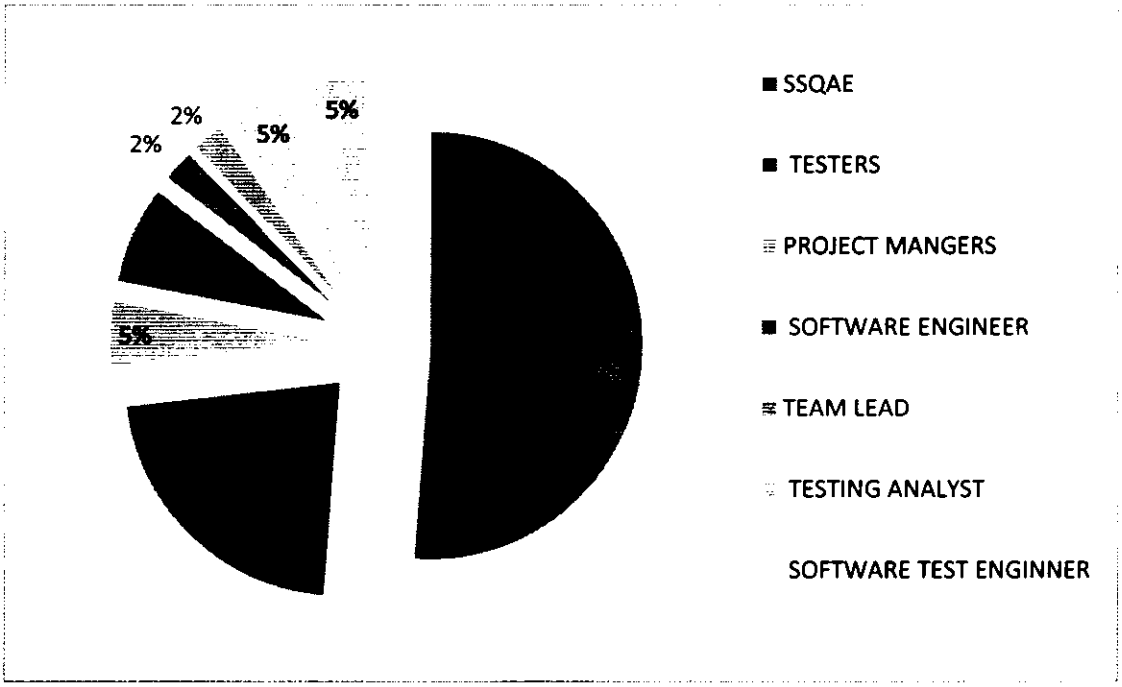


Figure 5-1

The percentage of Senior Software Quality Engineers was more than any other participants, which was 51%. The percentage of Testers, Team Lead, Project Manager, Testing Analyst, Software Test Engineer and Automating Testing Analyst was 22%, 2%, 5%, 5%, 2%, and 2% respectively.

The results are mainly related to three sections of 'Automated Software Testing' i.e. Benefits of AST, Challenges of AST and Software Testing Tools. We will discuss each section one by one.

5.2 Results Related to Benefits of AST

The overall responses of benefits of Automated Software Testing are shown in the figure as below:

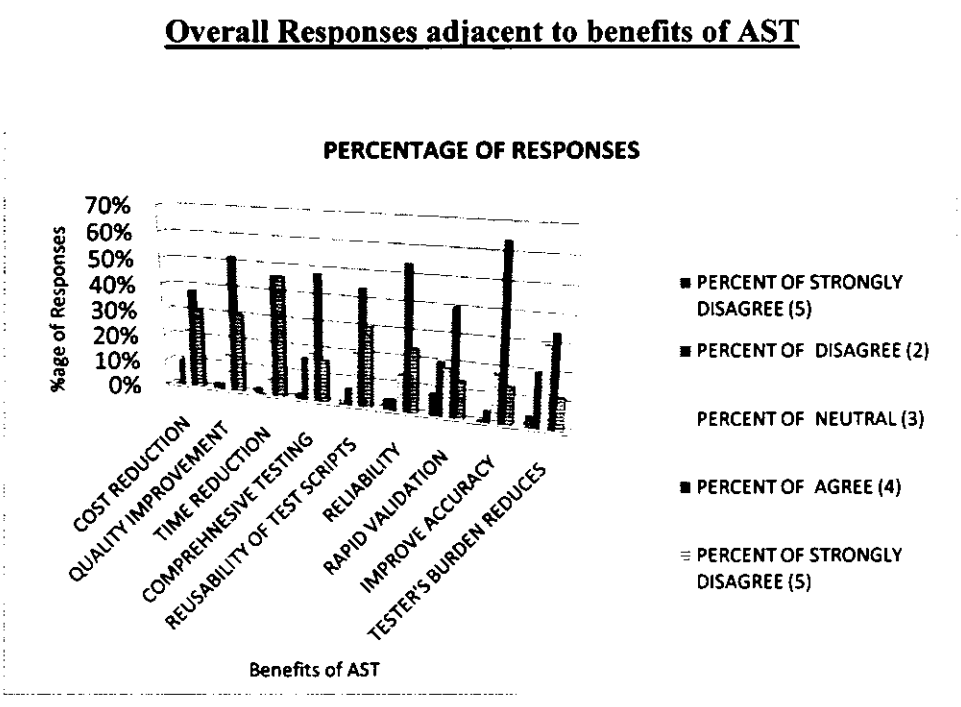


Figure 5-2

The percentages of benefits of Automated Software Testing according to respondents are as follows:

Agreed responses in favour of benefits of AST

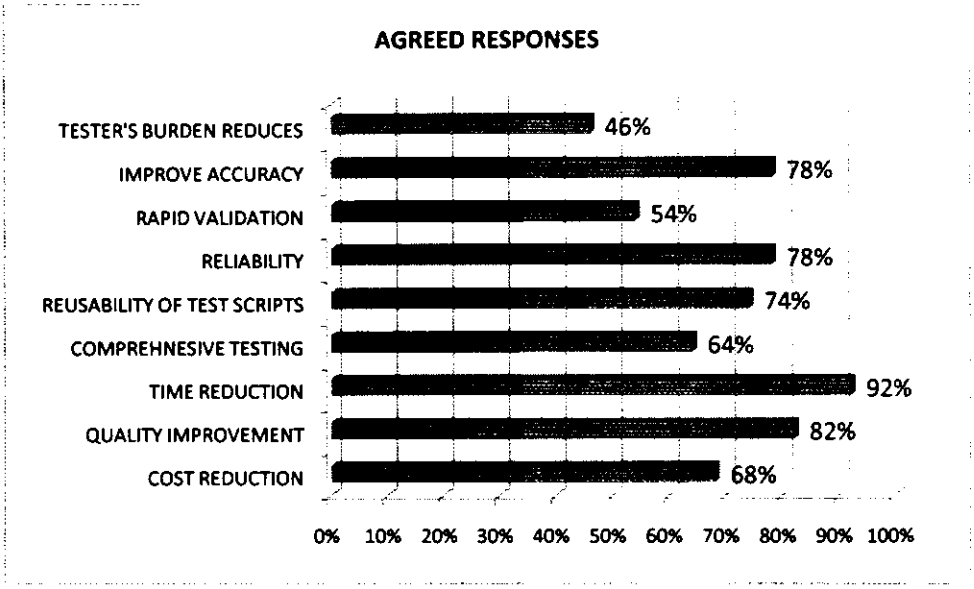


Figure 5-3

The figure 5-3 shows the proportion of agreed responses related to benefits of AST. 68% respondents agreed on the cost reduction of AST, 82% on quality improvement, 92% agreed on time reduction, comprehensive testing has agreed response of 64%, 74% on reusability of test scripts, reliability has value of 78%, rapid validation 54%, accuracy improvement 78% and 46% respondents agreed that tester's burden reduces with the usage of AST.

The result shows that most of the benefits of AST have greater percentage to agreement, which is above 70%. But a few has low percentage which is less than 70%. These are cost reduction, comprehensive testing, rapid validation and tester's burden reduction.

The respondents were asked to rate the benefits of AST with the options given in the Likert scale. The average of the responses is shown in the figure 5-4; Benefits of AST i.e.

Computation of Geometric mean of items of benefits of AST

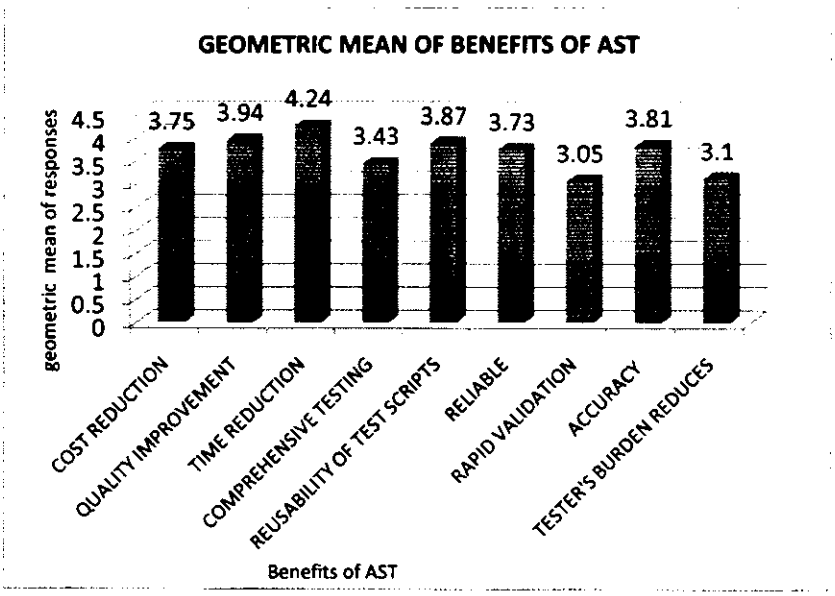


Figure 5-4

cost Reduction has value 3.75, quality improvement 3.94, time reduction 4.24, comprehensive testing has value 3.43, reusability of test scripts has value 3.87, reliability of tests got 3.73, Accuracy of results got 3.81 and tester's burden reduction has value 3.1. These values show that all the benefits of AST tend to agreement of the respondents except the comprehensive testing, rapid validation and tester's burden reduction. These values tend to somewhat disagreement or neutral response of the respondents.

After taking average of responses we chose 60% as threshold. The responses less than 60% depict the gap between literature and industry practices. So from the above figure it is depicted that rapid validation of changes , tester burden reduction using AST and

comprehensive software testing have percentage less than 60%, accordingly there is gap between literature and industry practices.

Gap identification in Benefits of AST

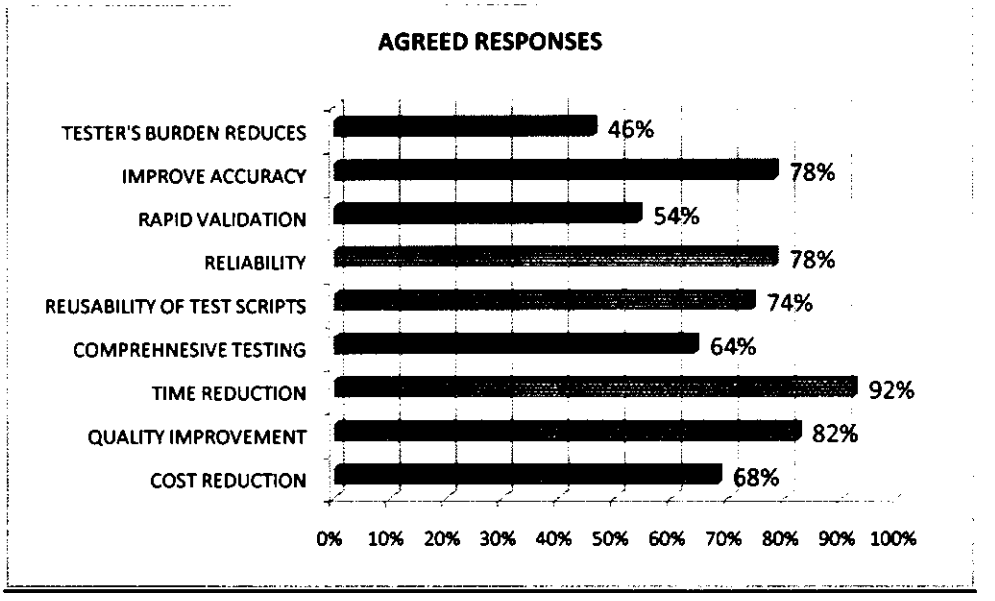


Figure 5-5

Same gaps can be proved through the geometric mean of responses of benefits of AST. This is shown in figure 5-6. Same gaps are identified through geometric mean of responses of benefits of AST. As comprehensive software testing, rapid validation and tester's burden reduction due to AST, have values less than 3.5.These go to disagreement with benefits of AST.

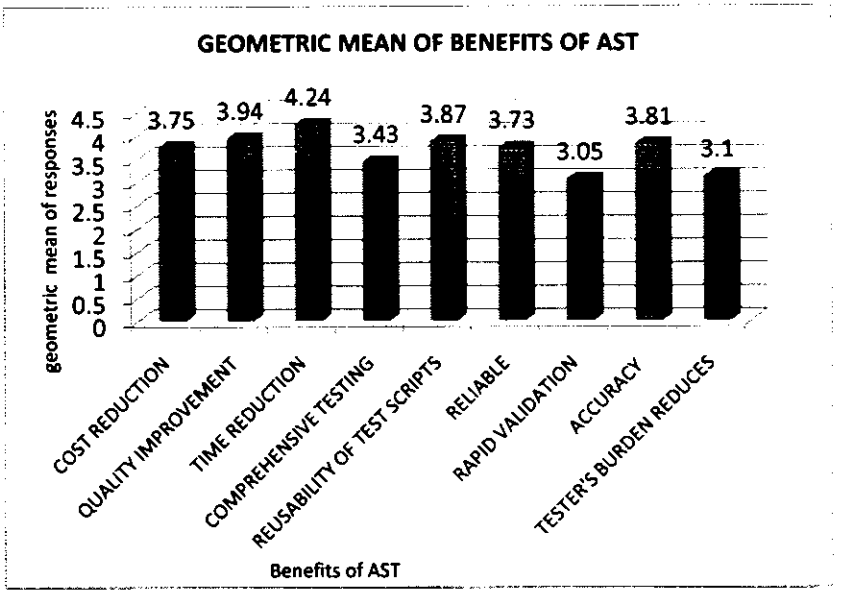


Figure 5-6

5.3 Results Related to Challenges of AST

Figure 5-6 shows the overall responses of the challenges of Automated Software Testing.

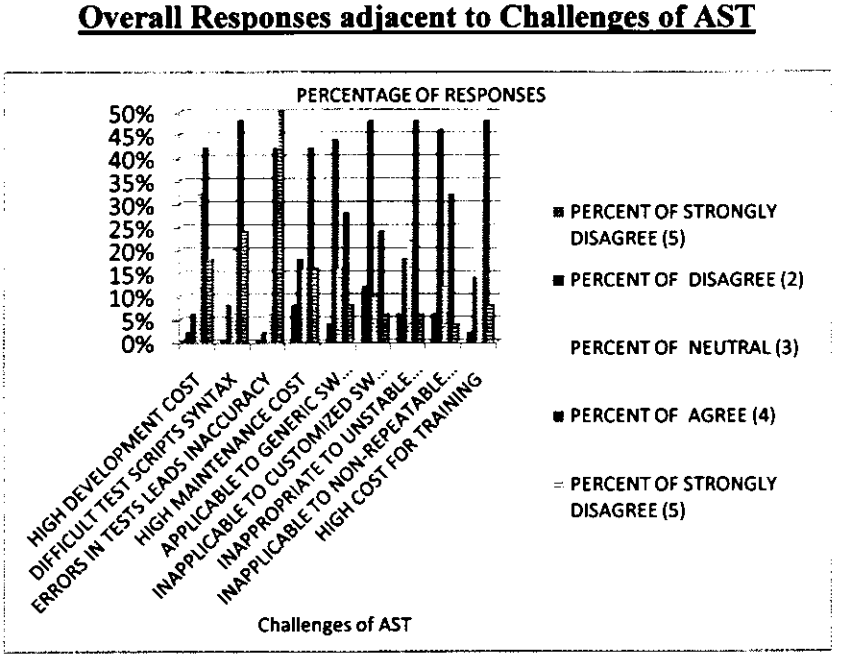


Figure 5-7

Figure 5-8 depicts the percentage of agreed responses against the challenges of Automated Software Testing. In the view of the respondents, 60% of the respondents viewed that AST has high development cost, 72% agreed that test scripts have complex syntax, 92% agreed that error in test script leads to inaccuracy in results, AST has high maintenance cost 58% respondents have the same opinion, 36% and 30% has consent that AST is appropriate to generic software products and not suitable to customized software products respectively. It is unfit for unstable underlying technology of the products that have to be tested, 54% respondents have the same opinion about this.36% and 56% of the respondents agreed that AST is not applicable to non-repeatable testing tasks and high training cost requirement respectively.

Agreed responses in favour of Challenges of AST

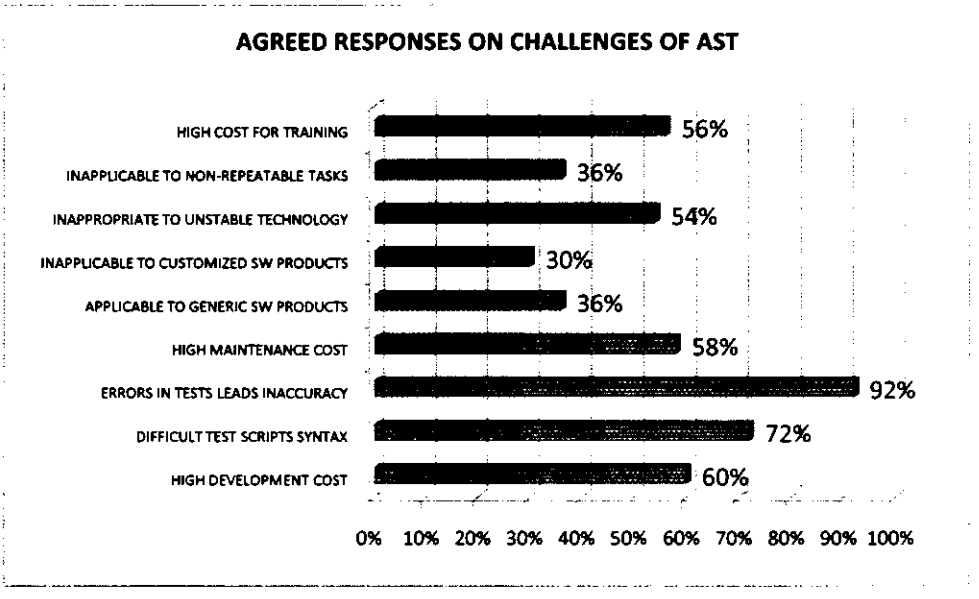


Figure 5-8

The following figure shows the weighted mean of responses of challenges of Automated Software Testing. The respondents were asked to rate the challenges of AST with the options given in the Likert scale. High development cost of AST has value 3.54, difficult test scripts syntax 3.77, errors in test scripts leads to inaccuracy has value 4.34, high maintenance cost 3.13, applicable to generic products has 2.71, inapplicability to customized software products 2.39, unsuitable to underlying unstable technology has value 3.9, non-repeatable testing tasks and high cost requirement for training has value 2.62 and 3.56 respectively.

Computation of Geometric mean of items of challenges of AST

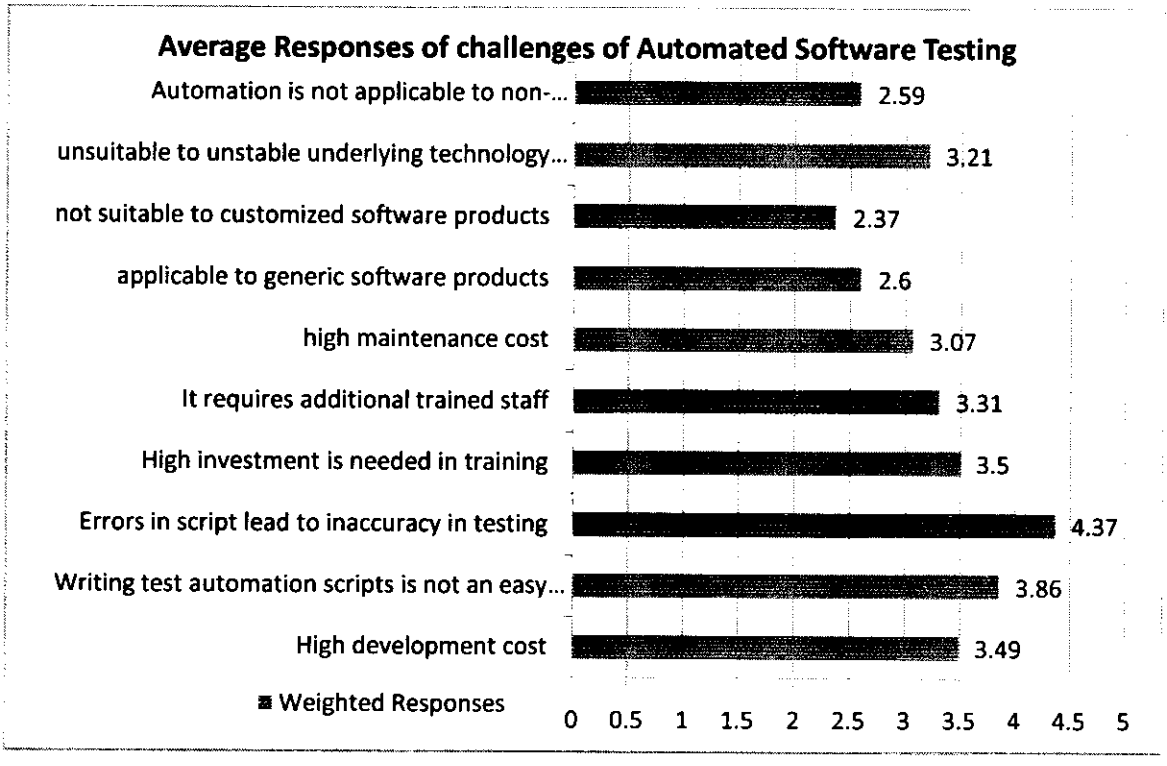


Figure 5-9

These values show that high maintenance cost, appropriation to generic products, not-suitable to customized software products , inapplicable to non-repeatable testing tasks has vales less that 3.5, which lie between disagreement and neutral responses of the respondents.

Gap identification in Challenges of AST

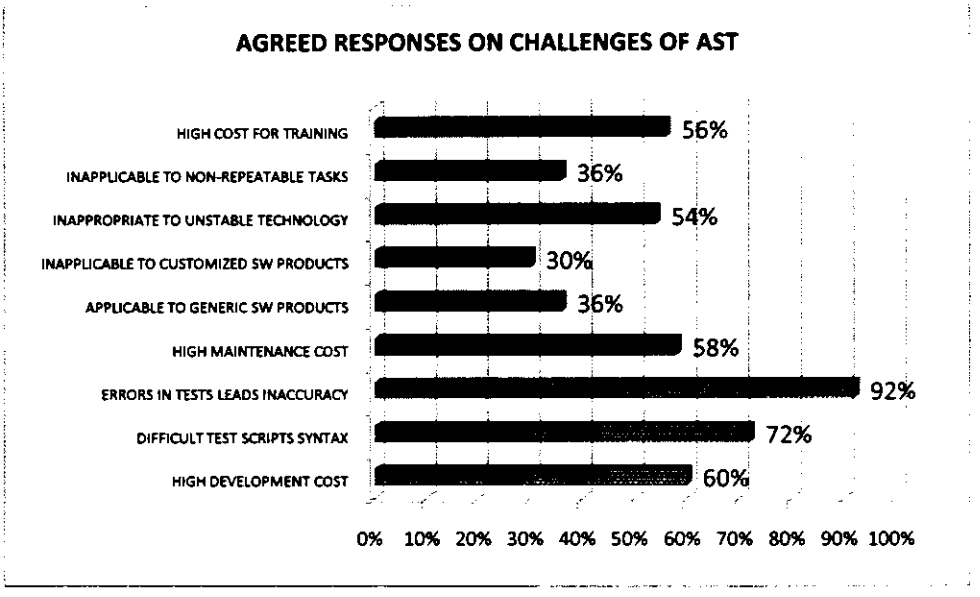


Figure 5-10

After taking average of responses we chose 60% as threshold. The responses less than 60% depict the gap between literature and industry practices. So from the above figure it is depicted that High cost for training, applicability of AST to non-repeatable tasks, inappropriate to unstable technology, inapplicable to customized software products, only applicable to generic software products and high maintenance cost have values less than 60%.

Same gaps can be identified through the geometric mean of responses of challenges of AST as shown in the figure below.

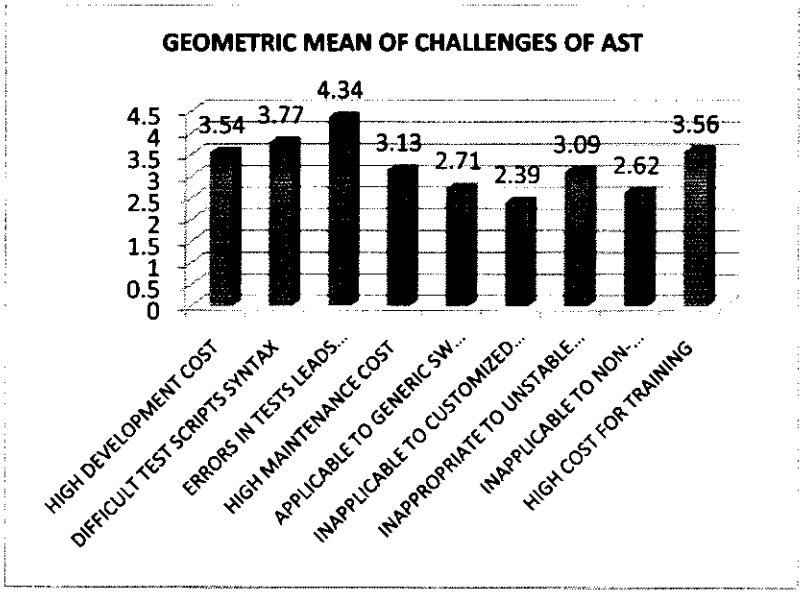


Figure 5-11

5.4 Automated Software Testing Tools

It was observed through results that most recurrent testing tool used in the local industry of Pakistan is HP QuickTest Professional; Selenium an open source tool is also used in most of the organizations. The testing tools used in the industry are mostly related to functional, performance, load and regression testing. It is depicted that both open source and proprietary licensed tool are being used in the industry.

The Figure 5-13 demonstrates that in the local industry of Pakistan, automated testing is established for last few years. The organizations that are currently using automated testing have experience ranging from three months to two years. Results depict that 60% of the testing work is still manual.

AST tools used in industry

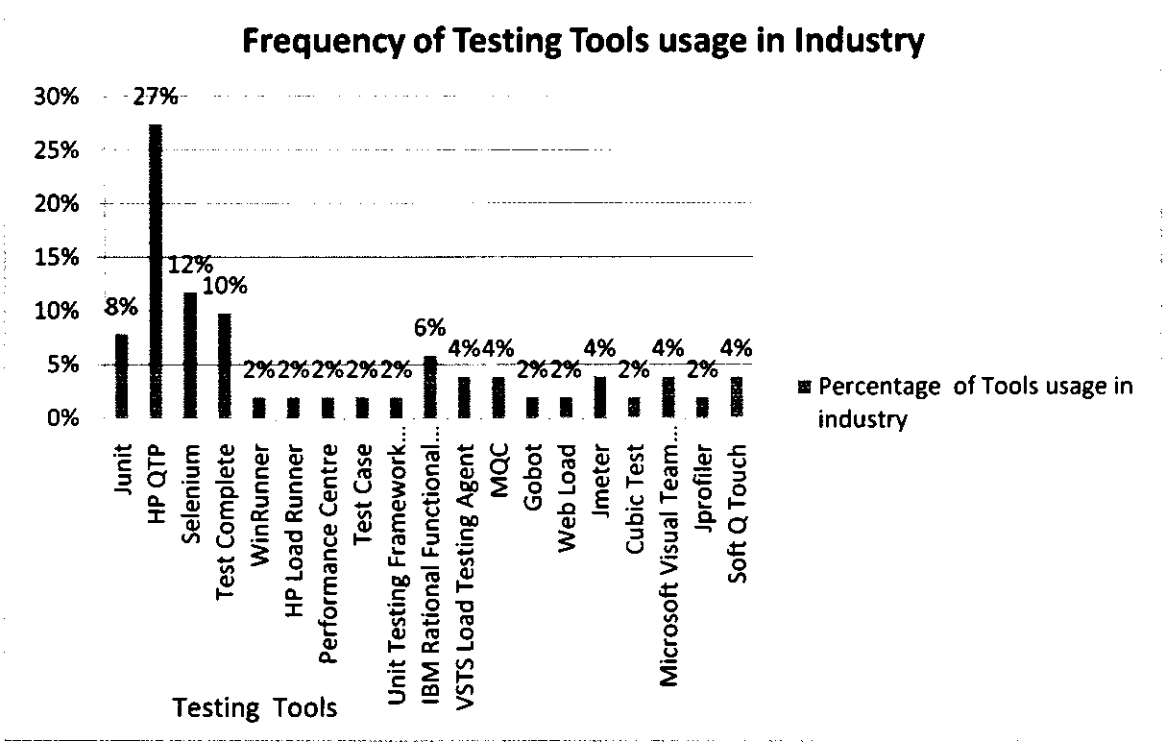


Figure 5-12

Experience of Industry in AST

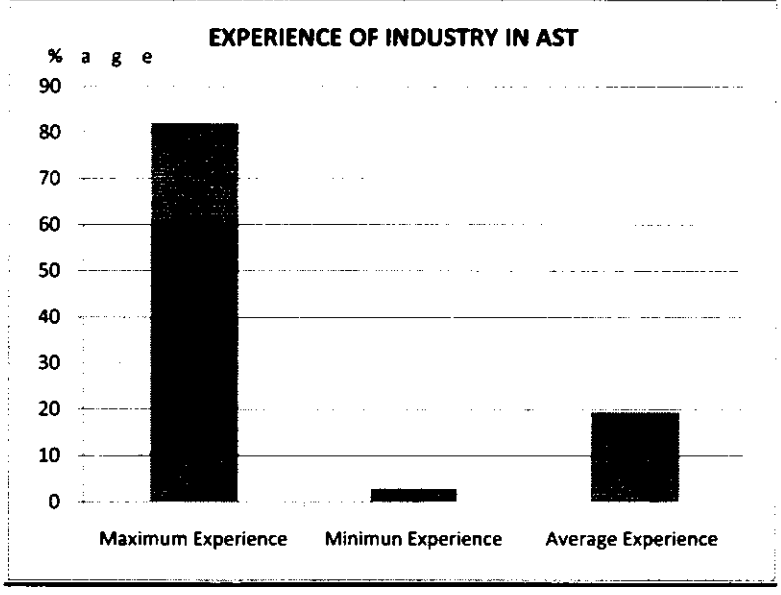


Figure 5-13

Respondents found that user interface testing is bit complicated and testers must have programming skills to write the test scripts. They notified that some tools are language dependent. It becomes a time consuming and tedious task: to manage automated test cases with dynamically changing requirements and continuous clients' feedback, especially when you are following agile development methodology e.g. scrum.

Chapter 6

CONCLUSION

6.1 Conclusion

Automated Software Testing is very valuable for the software industry. It has lot of benefits regarding the time, cost and quality of the software product as well as challenges with respect to its usage and automation.

It is observed from the research study that in local industry of Pakistan "Automated Software Testing (AST) "has been introduced for the last few years. The organizations that are currently using automated testing have experience ranging from three months to two years. From this study some gaps have been identified between literature and practices. Literature considers 'rapid validation & tester's burden reduction' as benefits of AST whereas the local industry does not agree with it. Literature consider "high development cost, trained staff, product type and unstable underlying technology" as challenges where as the local industry does not agree with it. It may happen that either Pakistan software industry professionals have lack of knowledge due to very short experience of automation or literature had not correctly report the benefits and challenges of AST.

There is need of further research in automated software testing. The above mentioned challenges should be solved to make AST process more effective. The world of automated software testing is one of the most challenging and gratifying jobs in the IT

industry. When done appropriately, automated testing is a deliberate challenge and a match of intellect between the manual tester and an application to find out hidden bugs, usability issues, and so forth.

REFERENCES

[1] Praveen Ranjan Srivastava, Km Baby, " Automated Software Testing Using Metaheuristic Technique Based on An Ant Colony Optimization", in proceedings of International Symposium on Electronic System Design, 2010.

[2] E.Kit, Software Testing in the Real World: Improving the process, Addison-Wesley, Reading, Mass, USA, 1995.

[3] E. Dustin, J.Rashka, ansd J.Paul, "Automated Software Testing: Introduction, Management and Performance", Addison-Wesley, Bostan, Mas, 1999.

[4] L.J. Whittaker, "What is software testing and why is it so hard?" IEEE Software, vol.17, no. 1, 2000.

[5] Jussi Kasurinen, Ossi Taipale, and Kari Smolander, "Software Test Automation in Practice: Empirical Observations", Hindawi Publishing Corporation Advances in Software Engineering, Volume 2010, 2010.

[6] S. Berner, R. Weber, and R.K. Keller, "Observations and lessons learned from automated testing", in Proceedings of the 27th International Conference on Software Engineering (ICSE'05), PP.579, May 2005.

[7] Katja Karhu, Tiina Repo, Ossi Taipale, Kari Smolander, "Empirical Observations on Software Testing Automation", in proceedings of International Conference on Software Testing Verification and Validation, 2009.

[8] L.S. Chin, D.J. Worth, and C. Greenough, "A Survey of Software Testing Tools for Computational Science", Software Engineering Group Computational Science & Engineering Department, June 29, 2007.

[9] Khaled M. Mustafa, Rafa E. Al-Qutaish, Mohammad I. Muhairat, "Classification of Software Testing Tools Based on the Software Testing Methods", ICCEE, IEEE, 2009.

[10] Mark Fewster Grove Consultants Llwyncynhwyra, "Common Mistakes in Test Automation", Grove Consultants, 2001.

[11] C. Persson and N. Yilmazt¨urk, "Establishment of automated regression testing at ABB: industrial experience report on 'Avoiding the pitfalls'", in Proceedings of the 19th International Conference on Automated Software Engineering (ASE '04), pp.112–121, Linz, Austria, September 2004.

[12] Leckraj Nagowah and Purmanand Roopnah, "AsT -A Simple Automated System Testing Tool", IEEE, 2010.

[13] Tibor R´ep´asi, "Software Testing – State of the Art and Current Research Challenges", 5th International Symposium on Applied Computational Intelligence and Informatics, May 28, 2009.

[14] Antonia Bertolino, "Software Testing Research: Achievements, Challenges, Dreams", FOSE'07, 2007.

[15] J. Bach, "Test automation snake oil," in Proceedings of the 14th International Conference and Exposition on Testing Computer Software (TCS '99),Washington, June 1999.

[16] Marcantonio Catelani, Lorenzo Ciani , Valeria L. Scarano, Alessandro Bacioccola, "Software automated testing: A solution to maximize the test plan coverage and to increase software reliability and quality in use", Computer Standards & Interfaces, science Direct, 2011.

[17] Aaron B. Brown and Joseph L. Hellerstein, "Reducing the Cost of IT Operations.Is Automation Always the Answer?", In proceedings of 10th Workshop on Hot Topics in Operating Systems, 2005.

[18] Cheng-hui Huang, Huo Yan Chen, Jinan Univ., Guangzhou, "A tool to support automated testing for web application", IEEE International Conference on Systems, Man, and Cybernetics, October, 2006.

[19] T. Parveen, S. Tilly, and G. Gonzalez, "A Case Study in Test Management", in ACM Southeast Regional Conference 2007, pp. 82-87.

[20] Kolawa, Adam; Huizinga, Dorota, "Management", Wiley-IEEE Computer Society Press 2007, pp. 74. ISBN 0470042125.

[21] R. Ramler and K. Wolfmair, "Economic Perspectives in Test Automation: Balancing Automated and manual testing with opportunity cost", in AST'06 Shanghai China: ACM, 2006.

[22] Cheng-hui Huang, Huo Yan Chen, "A Tool to Support Automated Testing for Web Application Scenario", Systems, Man and Cybernetics, 2006. SMC '06. IEEE International Conference on 16 July 2007.

[23] Ruffo, G., Schifanella, R., Sereno, M.and Politi, R., "WALTy: a user behavior tailored tool for evaluating Web application performance", Network Computing and Applications, 2004. (NCA 2004). Proceedings. Third IEEE International Symposium on 8 November 2004.

[24] Andrica, S., Candea, G., "WaRR: A tool for high-fidelity web application record and replay", Dependable Systems & Networks (DSN), 2011 IEEE/IFIP 41st International Conference, 18 July 2011.

[25] Ji-Tzay Yang, Jiun-Long Huang, Feng-Jian Wang, Chu, W.C, "An object-oriented architecture supporting Web application testing", Computer Software and Applications Conference,. The Twenty-Third Annual International, 6 August 2002.

[26] Krizanic, J., Grguric, A., Mosmondor, M., Lazarevski, P., "Load testing and performance monitoring tools in use with AJAX based web applications", MIPRO, 2010 Proceedings of the 33rd International Convention, 29 July 2010.

[27] Sofian, Hasliza, Saidi, Raihana Md, Yunos, Rozita, Ahmad, Siti Arpah, "Analyzing server response time using Testing Power Web Stress tool", : Science and Social Research (CSSR), International Conference on 27 May 2011.

[28] Bagnato, A., Raiteri, F., Mallouli, W., Wehbi, B., "Practical Experience Gained from Passive Testing of Web Based Systems", Software Testing, Verification, and Validation Workshops (ICSTW), 2010 Third International Conference on 2010, pp 394-402.

[29] http://jakarta.apache.org/jmeter/

[30] http://www.badboy.com.au/

[31] http://grinder.sourceforge.net/

[32] http://www.webload.org/

[33] http://www.neotys.com/

[34] N.E.Fenton et al," Predicting software defect as in varying development lifecycles using Bayesian nets", Information and software technology, 2007.

[35] Xing et al, "A novel method for early software quality prediction based on support vector machine", In Sixteenth IEEE international symposium on software reliability engineering, 2005.

[36] A.Carvalho et al, "A symbolic fault prediction model based on multi-objective practical swarm Optimization", The Journal of Systems and Software, 2010.

[37] I. Gondra, "Applying machine learning to software fault-proneness prediction", The Journal of Systems and Software, 2008.

[38] R. Torker and S. mankefors, "A Survey on testing and reuse", in Proceedings of IEEE International Conference on Software Science, Technology and Engineering, Herzlia, Israel, November 2003.

[39] M. Blackburn, R. Busser and A. Nauman, "Why model-based test automation is different and what you should know to get started", in Proceedings of the International Conference on Practical Software Quality, Braunschweig, Germany, September 2004.

[40] D. J. Mosley and B. A. Posey, "Just Enough Software Test Automation", Prentice-hall, Upper Saddle River NJ, USA, 2002.

[41] Timothy C. Lethbridge, "A Survey of the Relevance of Computer Science and Software Engineering Education", Proceedings of CSEE&T'98, February 22-25, 1998 in Atlanta, Georgia.

[42] Shad Lawrence Pfleeger Barbara A. Kitchenham, "Principles of Survey Research Part 1 Turning Lemons into Lemonade", ACM SIGSOFT Software Engineering Notes Volume 26, 6 November 2001.

[43] Barbara A. Kitchenham, Shad Lawrence Pfleeger, "Principles of Survey Research Part 2: Designing a Survey", ACM SIGSOFT Software Engineering Notes Volume 27 No 1, January 2002.

[44] Barbara A. Kitchenham, Shad Lawrence Pfleeger, "Principles of Survey Research Part 3: Constructing a Survey Instrument", ACM SIGSOFT Software Engineering Notes Volume 27 No 2, March 2002.

[45] Barbara A. Kitchenham, Shad Lawrence Pfleeger, "Principles of Survey Research Part 4: Questionnaire Evaluation", ACM SIGSOFT Software Engineering Notes Volume 27 No 3, May 2002.

[46] Barbara A. Kitchenham, Shad Lawrence Pfleeger, "Principles of Survey Research Part 5: Populations and Samples", ACM SIGSOFT Software Engineering Notes Volume 27 No 5, September 2002.

[47] Barbara A. Kitchenham, Shad Lawrence Pfleeger, "Principles of Survey Research Part 6: Data Analysis", ACM SIGSOFT Software Engineering Notes Volume 28 No 2, March 2003.

[48] Busha, Charles H., and Stephen P. Harter, "Research Methods in Librarianship: Techniques and Interpretation. Orlando, FL: Academic Press, Inc., 1980.

[49] Babbie, Earl R., "Survey research methods", Belmont, CA: Wadsworth Pub. Co, 1973.

[50] C.Kaner, "Improving the maintainability of automated test suites", Software QA, Vol. 4, 1997.

[51] S.P. Ng, T. Murnane, AK. Reed, D. Grant, and T.Y. Chen, "A preliminary survey on software testing practices in Australia", in Proceeding of the Australian Software Engineering Conference, pp. 11-12, Melbourne, Australia , April 2004

[52] T. Dyba, "An instrument for measuring the key factors of success process improvement", empirical Software Engineering, vol.5, no. 4, pp. 357-390, 2000

APPENDIX

Appendix -Questionnaire

AUTOMATED SOFTWARE TESTING: AN INSIGHT INTO LOCAL INDUSTRY

The purpose of this survey is to validate the issues, benefits and tools of automated software testing in software development organizations of Pakistan. Based on the frequency of occurrences, the benefits, issues and tools of automated software testing will be prioritized. The prioritized lists are beneficial while adopting it.

Role:	**Company Name:**
Experience in Automated Software Testing: **___YEARS____MONTHS**	**Automated Software Testing followed for all projects: Yes/No**
Automated Testing Tools used:	

Scale:

1 = Strongly disagree
2 = Disagree
3 = Neutral or Do not use
4 = Agree
5 = Strongly agree

For each item identified below, mark the number to the right that best fits your judgment. Use the scale above to select the number. If you don't know the answer, please leave it blank.

Benefits of Automated Software Testing	**1**	**2**	**3**	**4**	**5**
It reduces the cost of software testing	☐	☐	☐	☐	☐
Software quality improves when we use the testing tools	☐	☐	☐	☐	☐
It helps in time reduction	☐	☐	☐	☐	☐
Comprehensive and thorough software testing is possible	☐	☐	☐	☐	☐
It facilitates Reuse of test scripts	☐	☐	☐	☐	☐
It is more reliable (Tests perform precisely the same operations each time they are run, thereby eliminating human error)	☐	☐	☐	☐	☐
Rapid validation of software changes with each new release of application is possible.	☐	☐	☐	☐	☐
It increases the accuracy of testing	☐	☐	☐	☐	☐
It helps the testers to focus on creative tasks	☐	☐	☐	☐	☐

It requires less human knowledge for testing the product	☐	☐	☐	☐	☐
It decreases the burden of the tester	☐	☐	☐	☐	☐
Challenges of Automated Software Testing	**1**	**2**	**3**	**4**	**5**
Automated software testing has high development cost	☐	☐	☐	☐	☐
Writing test automation scripts is not an easy task	☐	☐	☐	☐	☐
Errors in script lead to inaccuracy in testing	☐	☐	☐	☐	☐
The more functionality covered by automation, the more complicated the test programming	☐	☐	☐	☐	☐
High investment is needed in training	☐	☐	☐	☐	☐
A lot of testing areas are left uncovered	☐	☐	☐	☐	☐
Automation is too burdensome	☐	☐	☐	☐	☐
It requires additional trained staff	☐	☐	☐	☐	☐
Automated testing cannot replace human judgement	☐	☐	☐	☐	☐
It has high maintenance cost	☐	☐	☐	☐	☐
It is only applicable to generic software products	☐	☐	☐	☐	☐
It is not suitable to customized software products	☐	☐	☐	☐	☐
It is not applicable to the products which have unstable underlying technology	☐	☐	☐	☐	☐
Automation is not applicable to non-repeatable testing tasks	☐	☐	☐	☐	☐
It is mainly applicable to unit and regression testing	☐	☐	☐	☐	☐

Please inform about the Testing Tools that you are using in your Organization:

Testing Tool Name:	**In-house/Outsource/Open Source Testing Tool:**
Duration in using the Testing Tool:	**Type of the Projects (i.e. Web application, Database, Desktop etc):**

Automated Software Testing Tools in an Organization	Response
What benefits does it offer?	
What is the cost of the automated testing tool?	
What are the features of automated testing tool?	
What features of testing tool are you using?	
Are you using all the features of the testing tool? If not then why?	
What parts of the testing process have you automated or provided by tool (i.e. testcase generation, test execution etc)?	
What percentage of the testing work is purely manual?	
Does it affect the quality of software testing?	
What experiences do you have regarding the usage of testing tools?	
What are the limitations of the testing tool?