# Guidelines for the Development of Automated Test Case Execution Tool(s)

**Name: Asma Habib**

**Registration No: 360-FBAS/MSSE/F12**

**Supervisor: Dr. Rizwan Bin Faiz**

**Co-Supervisor: Mr. Muhammad Nasir**

Department of Computer Science Faculty of Basic and Applied Sciences International Islamic University Islamabad

2015

**Department of Computer Science & Software Engineering**

International Islamic University Islamabad

## Final Approval

**Dated:** 4th Sep, 2015

It is certified that we have read the project report submitted by Miss Asma Habib, Registration no. 360-FBAS/MSSE/F12.It is our judgment that this research thesis is of sufficient standard to warrant its acceptance by the International Islamic University, Islamabad for the MS Degree in Software Engineering.

## Committee

### External Examiner

Dr. M. Zohaib Zafar Iqbal
Assistant Professor,
FAST University, Islamabad
zohaib.iqbal@nu.edu.pk

### Internal Examiner

Mr. Zulqarnain Hashmi
Lecturer.
Department of Computer Sciences,
IIUI, Islamabad.
zulqarnain@iiu.edu.pk

### Supervisor

Dr. Rizwan-Bin-Faiz
Assistant Professor
RIU, Islamabad
rizwan.faiz@riu.edu.pk

**A dissertation Submitted To**

**Department of Computer Science,**

**Faculty of Basic and Applied Sciences,**

**International Islamic University, Islamabad**

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

**Degree of MS Software Engineering**

# Declaration

We hereby declare that this Thesis "Framework for the development of Automated Test Case Execution tool(s)" neither as a whole nor as a part has been copied out from any source. It is further declared that we have done this research with the accompanied report entirely on the basis of our personal efforts, under the proficient guidance of our teachers especially our supervisor Dr. Rizwan-Bin-Faiz and Co-supervisor Muhammad Nasir. 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, we shall stand by the consequences.

Asma

**Asma Habib**
**360-FBAS/MSSE-F12**

# Acknowledgement

All praises to Almighty Allah the most merciful and the most Beneficent for granting me the courage and knowledge to complete this work. Without the blessings of Allah, I would not be able to complete my work. Praise to our Holy Prophet Hazrat Muhammad (May peace & blessing of Allah SWT be upon him) whose teachings enable me to recognize our creator Allah Almighty and path of guidance.

First I would like express my sincere gratitude to my honorable supervisor Dr. Rizwan Bin-Faiz for his continuous support, patience, motivation, enthusiasm and immense knowledge during the entire research period. I would like to thank Mr. Muhammad Nasir for his valuable time and suggestions who monitor my work and took effort in reading and providing me with valuable comments.

Specially, I want to acknowledge the prayers, endless love, efforts and support of my family without which I was unable to reach this stage and my special thanks goes to my sister Arsa Habib, my father for their well wishes, moral support and continues encouragement. I am especially indebted to my friends Samia Rafique and Asma Habib for their moral support.

Asma

**Asma Habib**

**360-FBAS/MSSE-F12**

# Abstract

Test cases are executed to find defects and improve the quality of software product. Test case execution is the most observable activity as it reduces testing cost, effort. It therefore has a strong impact on effectiveness and efficiency of the whole testing. In general automated software testing as adapted by many industries is well known method to improve the quality of software product [11].

Aim of current research is to provide a framework for the development of automated test case execution tool(s). Although there are many commercial and open source test case execution tools [21-35] available but their parameters/features (functionality) is not aligned with current requirements of software industry. Therefore there exists a real need to have a test case execution framework which categorizes most comprehensive set of features in to core phases of test case execution. Proposed framework provides architectural foundation for test case execution tool(s) in order to define their scope with respect to phases and its feature set instead of implementing features which may belong to various phases. Proposed framework is rigorously evaluated through well reputed international organizations i.e. Net sole, Siemens, IBM systems, Microsoft etc, and by organizations which are CMMI level 5 and ISO 9001, 90003 certified and also by highly experienced i.e. over 15 years, software and quality engineers.

# Table of Contents

# LIST OF TABLE

# LIST OF FIGURE

# LIST OF ABBREVIATIONS

| | |
|---|---|
| **ATCE** | Automated Test Case Execution |
| **SQA** | Software Quality Assurance |
| **CMMI** | Capability Maturity Model Integration |
| **ISO** | International Organizations for Standardizations |

# Chapter 1
# Introduction

# 1. Introduction

This chapter presents an overview of automated test case execution, research problem, research aims and objectives. The research methodology is be briefly introduced followed by an outline for the rest of the report.

## 1.1. Background

In general automated software testing is well known method which is adapted by many industries to improve the quality of software product. These companies make use of test automation tools to ensure that software delivered is of high quality [11].Testing tools are the software products which are used to automate or semi automate the software testing methods or process, such as test case designing, execution and incident reporting [12,13]. Test case execution is the most observable activity as it consumes 40% of testing effort therefore it has to be concluded as quickly as possible [14].Test case are executed to find defects and increase the confidence in quality [17][11]. In automated test case execution generated and modified test cases are automatically executed, compare actual and expected result and record test result in automated test environment on a system under test, [11].

Automated test case execution is useful to perform regression testing more efficiently, execute more test cases in less time ,execute same tests on different hardware configuration using different operating system[17].Test case execution has an important characteristic therefore different tools are used to decrease the time spent during the entire testing process [16] There are a number of commercials and open source test case execution tools [21-30] which implement a verity of features and particularly used in industry practice [12] [15]. However, feature set of current tools are misalign with current requirement of software industry. The reason current tools do not have any framework or a set of guidelines validated as per current needs of software industry. Therefore, there is a need to bridge the gap by providing a framework for test case execution tool(s) whose phases and their feature set are designed based upon current industrial feedback.

The goal is, therefore, to develop such test case execution framework which bridge the gap between academia and industry and provide architectural foundation for test case execution tools. This study will help those companies and practitioners who implement automated test case execution tools in future.

## 1.2. Motivation

Many survey are conducted on software testing practice which indicate that test case execution tools need improvement [1]. However, currently there are large number of automated test case execution tools available in software industry [21-30].The current tools implement variety of features but they are still misaligned with the current requirement of software industry [1].The existing tools have no consensus in their parameter because everyone develops tool according to their requirement having different scope with in development house. The reason that they do not have any common set of guidelines not even single tool know which parameters should be implemented to execute test cases. These motivate us to provide framework for all those people who implement test case execution tool and have clarity that according to this framework what set of feature should be must implemented in the software industry

## 1.3. Aim/Objectives

The aim of this research is to design framework for development of automated test case execution tool which provides foundation to those who develop them. The objective of this research are primarily describe in the list below:

- First we perform literature survey to identify list of open source and commercial test case execution tools, their common set of features and tool evaluation criteria which include standard presented by IEEE, ISO, IEC etc.
- Secondly we conduct industrial survey to identify currently implemented features, and set of desired feature that should be implemented by automated test case execution tool.
- Finally we will conduct detailed multi-dimensional analysis of industry survey and tool elevation criteria. This will lead us to establish framework for development of automated test case execution tool.

## 1.4. Research question

RQ.1.What is maximum set of feature(s) and core phases of test case execution tool(s) and their classification of feature set in to core phases?
RQ.2.How can we evaluate proposed framework?

## 1.5. Expected outcome

i. Framework for the development of automated test case execution tools

## 1.6. Research Methodology

In order to get a sufficient amount of information the study has been dividing in to the following parts

### 1.6.1. Literature review

An exhaustive literature survey is conducted which was taken for the identification of most common referred tool used for automated test case execution, their feature set and for the identification of different framework, methods, approaches of test case execution tools are considered as state of the art.

### 1.6.2. Research method

The proposed research is a large scale survey research. The empirical investigation through survey research was intended to propose framework for automated test case execution tool .we performed web-based international survey.

### 1.6.3. Data Collection

Data is collected from personal of quality and testing department, in global software organization. The targeted software organization are CMMI certified or have separate Software Quality Assurance department or medium scale software house. We use social network and personal contacts to elicit data from respondent using face book, linked in and through emails.

### 1.6.4. Data Analysis Method

Initially in this thesis qualitative and quantitative approach are used and multidimensional analysis is performed. The result combine from this survey consist of sufficient information for the construction of framework for the development of automated test case execution tools.

## 1.8. Thesis Outline:

The thesis report is divide into a number of chapters which are explained below:

**Chapter 2:** chapter provide the literature review of automated test case execution identified from academic literature survey. This chapter present the recommended practices, gap analysis for future research of test case execution.

**Chapter 3**: discusses about research methodology and method used to collect data from industry.

**Chapter 4:** discussed the detail analysis of results extracted from questionnaire.

**Chapter 5**: is about a discussion on the validity of results with regard to provide framework for test case execution.

**Chapter 6:** is about the conclusion and future work that can be done on this thesis further.

# Chapter 2

# Literature Review

# 2. Introduction

From the past few decades' large number of work on test case execution tools had been done. In order to obtain information about test case execution framework, approaches or feature set detail literature survey was performed.

This chapter discuss major frameworks and approaches as referred in literature. The purpose was to investigate how comprehensively framework were proposed and identify gaps and cluster for future research and improvement are suggested.

## 2.1. Literature Review:

J.LEE et al [1] conducted a survey on software testing activities Respondents were asked about their desire improvements in testing activities and process. Survey revealed the limitations in software testing process and tools. However proposed framework is limited to few phases of test case execution. Besides association of set features against core phase of test case execution is not established.

T.E.J.Vos et al [2] presented a generic framework for evaluating software testing tools nature of the tool it contain prerequisites, performed operations, results and tool license; subject who was using the tool, subject is basically worker of the company; object is the function or program under test and final part define the variables which shows the efficiency, effectiveness and satisfaction of the tools. The proposed framework was evaluated through three case studies but more importantly it lacks feedback from current software industry.

LeckrajNagowah et al. [3] develop web based tool for test case execution. The tool was developed based upon existing limitation of common automated testing tools (IBM functional tester, Selenium IDE, Quick test professional, Sahi). Result showed that proposed tool perform test execution more rapidly and with minimum human interference, support web application language and enable user to support regression testing. Although proposed framework was evaluated through experimental results, however it does not involve feedback from software industry.

Fei Wang et al. [4] proposed a web-based test automation framework for test execution. The framework was design based upon two test execution tool Selenium and Jmeter. Result showed framework support multiple browser, operating system and convenient to switching different type of testing and perform test execution. Similarly Vera Stoyanova et al. [5] presented web based framework for automated test case execution. The framework was designed based upon the limitations of existing five SOA (Service Oriented Architecture) testing tools to perform test case execution. An experiment was conducted to evaluate the proposed framework. Results showed definition of assertions at different level and execution of test case. Although proposed framework was evaluated through experimental results, however scope of the proposed tool limited small feature set, also it lack software industry feedback.

J. Tang [6] proposed test automation model for Software Testing Lifecycle (STLC) activities, including test case design, test case execution, and test case incident reporting. The model was categorize in to; requirement agent, construct agent, execution agent and report agent. Execution agent is responsible for execution of test cases under specific test environment through specific test tool. Since this study only reviewed academia it therefore does consider current requirements of software industry.

E.Ha Kim et al. [7] proposed a framework for test automation tools. The framework was design based upon two test automation framework STAF *(software test automation framework)* and Fit *(Framework for integrating testing).*Result showed that STAF fixture was used to control flow of execution using keywords that redirect the path of execution, it automate test execution and result analysis. Although proposed framework was evaluated through case study, however scope of the proposed tool limited and lack feedback from current software industry.

G.Jing et.al [8] presented "an agent based distributed automated test execution framework" for different type of testing. The framework was design based upon two framework OSGI technology (Open Service Gateway Initiative) and JADE (Java Agent Development Framework). The framework was categorized in to three parts; GUI test console, test execution server and communication system in which test case execution server is further categorized into; test master, monitor test execution, test execution agent, test result collection agent. The framework performed complete test execution on different type of testing by integrating different type of test case. Even though proposed framework was evaluated through different type of testing but it yet does not associate set features against few phases of test case execution.

Tauhida Parveen et al. [9] proposed "a distributed execution framework (Hadoop unit) "for Junit test case to reduce test execution time. The framework was design based upon map reduce algorithm. An experiment was conducted and result shows that to execute single test case was take more time as compare to run the whole test suite. Although proposed framework was evaluated through Junit test case and test suite but it is not evaluated by software industry.

T.Abdou et al. [10] proposed a framework which defines the testing process of open source software. Testing process of open source software is compared with ISO/IEC standard software testing process. Result shows that open source software testing is similar and improvements are highlighted in the task related to each activity. Proposed frameworks lack industrial feedback and more importantly framework is instead of suggesting comprehensive feature set against phases, features are randomly proposed which may belong to any phase.

Critical analysis of above literature emphasis upon the need to develop such framework which not only involve industrial feedback but can also relate features against core phases of framework for test case execution tool(s).

## 2.2. Discussion/Gap

By analyzing the existing literature we can see that features set of common referred automated test case execution tools is not organized into phases. Besides proposed frameworks are based upon academic reviews, not validated from software industry and domain specific. Since choice of feature set to be implemented by automated test case execution tool are not validated by software industry, therefore desired set of features to be implemented by current tools may not be aligned with current requirements of software industry.

To overcome the existing gaps our study proposed a test case execution framework in which industrial feedback are involve. We will proposed a framework which categorizes most comprehensive set of features in to core phases of test case execution. The framework will provides architectural foundation for test case execution tool(s) in order to define their scope with respect to phases and its feature set instead of implementing features which may belong to various phases.

# Chapter 3

# Research Methodology

## 3. Introduction:

This chapter emphasizes on the research methodology including research approach, research method, survey design technique, data collection method, structure and explanation of our questionnaire used in the development of framework of automated test case execution tools. The purpose of this section is to explain how we have conducted our research process. This empirical investigation through survey research aimed to provide framework for the development of test case execution tools

### 3.1. Research Approach:

For empirical research two approaches are used by researcher that may be inductive and deductive approach. We use inductive approach for our research work because inductive approach are used when researcher explore an unfamiliar phenomena where little theory exist [19].This approach is start from set of observation [19].Systematic tools are used in inductive approach like questionnaire, interview and theories. [18]

### 3.2. Research Method:

In literature there are three different empirical strategies are discussed survey, case study, experiment [18]. Researchers used these methods according to their research objective. We used survey method to collect data. The reason behind using survey method was that, it is conducted when the tool or techniques already has in used [18].

### 3.3. Research Design

For gathering information we used online survey questionnaire to elicit data from international level software organization. Questionnaires result can be easily analyzed quantitatively and qualitatively [18]. Our survey includes various phases of test case execution, each of which was further explained through comprehensive feature(s) set. Phases and feature(s) set were identified through literature review which includes ISO/IEC 29119 standard and common referred open source and commercial tools.

## 3.4. Data collection Sources and Methods

Data collection is usually divided in to two parts that are primary and secondary data. The primary data was collected from the industry to address the problem while the secondary data was collected through literature survey already available in research journals, conference proceeding papers. For secondary data we performed literature survey which is discussed below in detail.

### 3.4.1. Literature Survey

An extensive literature survey was conducted in the first part of the study which resulted in the relevant test case execution background for use in the industry survey. The primary intention of this survey is to find out the most common referred tool used for automated test case execution and their set of feature .And to find out that research which empirically evaluated different tools features, techniques, approaches, framework, and guidelines. The purpose of this review was to identify the recommended practices of test case execution. The importance is derived for the fact all that tools perform test case execution in different ways the reason is that they do not follow a common set of guidelines according to which they perform test case execution. Different research paper are review and the study was included if it is relevant to our thesis background otherwise discarded. From each included research paper a summary of different view of author related to our topic was chosen and discussed in chapter 2.

#### 3.4.1.1. *Literature Search Strategy:*

In order to get the current state of research with in the field of automated test case execution tools, the literature survey was initiated as the first phase of research thesis. This study is iterative where carefully selection of each research paper was evaluated on the basis of relevance and quality aspects. An acknowledge literature data bases as well as searching journals, conference proceeding was used to increase the chance of sufficient research quality for this purpose following research engine are used.

- IEEE Explore
- Springer link
- Science Direct
- ACM digital library

- CiteSeerX
- Google scholar

The search was performed with combination of many key terms related to our study. For each paper significant data was extracted and references at end of each papers were also quickly examined to find more relevant papers related to our interest.

### *3.4.1.2. Analysis of Common referred tools*

Several tools have been proposed in literature for test case execution. Among these most common referred tools are selected which are physically accessible i.e. either commercial or open source, whose versions are regularly updated and their latest versions are since 2010 to onward. Table 3.1 shows gaps and clusters in the feature set of tools. Such gaps and clusters in feature set were used to design the questionnaire.

| **Common Referred Tools** | | **Feature Of Test case Execution Tool** | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | **Execute** | | **Compare Actual & expected Result** | **Record** | | **Approach/ Framework** | |
| **Tools** | **License** | **Test cases** | **Test Suite** | | **Detail Log** | **Short Log** | **Code-driven** | **GUI-driven/Record-Playback** |
| Pex [30-32]<br>2010 | Commercial | Yes | Yes | Yes | - | - | Yes | - |
| Junit [21-24]<br>2012 | Open source | Yes | Yes | Yes | - | - | Yes | - |
| Selenium [25-27]<br>2013 | Open source | Yes | - | Yes | - | Yes | - | Yes |
| FitNesse [33-34]<br>2013 | Open source | Yes | - | Yes | - | - | Yes | - |
| JCrasher [25][35]<br>2010 | Open source | Yes | - | Yes | - | - | Yes | - |
| Rational Robot [29]<br>2011 | Commercial | Yes | - | Yes | Yes | - | - | Yes |

**Table 3.1 Analysis of Common referred Tools**

### 3.4.2. Industrial survey

The primary data was collected from industrial survey. This part actually included the development of framework for automated test case execution tools. The primary data was collected form national and international software organization through questionnaire. Industrial survey consists of following steps.

#### 3.4.2.1. *Population and Sample*

The primary objective of this survey is to provide framework for automated test case execution tool used in organization. The scope of our study was to collect data from QA department of software organizations in different countries and in-house software organization. Population of our survey was software quality department people.

## 3.5. Preparation and validation of questionnaire

We prepared a well design questionnaire which meet the research objective it consist of following general steps:

- The questionnaire was design using "Google doc" because we performed online survey to meet our objectives
- The length of the questionnaire is short and wording we used is so simple which was equally understandable by every respondent
- All technical terms were explained with in questionnaire.
- The questionnaire was based on close-ended question to take precise answer and some open-ended question to get their opinion and for additional knowledge
- The questionnaire was sent through e-mail, social network like linked-in so it was convenient for respondent to reply easily.

### 3.5.1. Questionnaire Structure

Our Questionnaire consist of three parts that are:

#### 3.5.1.1. *Demographic Detail*

This section consist of respondent information to assure the respondents name, experience, designation and email address.

#### 3.5.1.2. *Organizational Detail*

This section contain the information about organization CMMI level, ISO certification, country in which organization are located and number of people working in organization.

#### 3.5.1.3. *Test case execution Detail*

This section emphasis on information of test case execution to assure the current feature of test case execution tools.

### 3.5.2. Questionnaire Management

The scope of the study was to elicit data from global organization; therefore, data was collected by a web based survey using "Google Doc service". Google doc provide a web survey platform. Approximately 20-30 minutes were required to facilitate survey completion. The respondent were allowed to save their survey responses. The save and continue feature was conducted internally in survey website.

In the first step, web hyperlink link was send to the targeted respondent with the pre-notification included survey purpose and description, researcher's contact information, a statement about confidentiality of the respondent's response. The web hyperlink of online questionnaire was send out using social networks like email, Facebook, linked-in to the target participant. A reminder email was send after two weeks to those who had not answered.

The survey was conducted from July 7, 2014 to Sept 28, 2014.The total number of submitted responses was 133.

## 3.6. Objective of research survey:

We designed questionnaire to conduct our research survey which consist of closed-ended and open ended question to meet our objectives that are

- To identify set of parameter which are mostly used in industry for automated test case execution tools
- To investigate the desire feature of tools according to software industry
- To provide minimum set of feature which must be implement during development of automated test case execution

- To provide the framework for the development of automated test case execution tools.

# CHAPTER 4

# Result Analysis

# 4. Introduction

This chapter presents a detailed analysis of the survey which was conducted to investigate different companies at national and international level. The survey analysis is performed based upon on the classification of test case execution activities. In first section demographic analysis of all companies and respondent is presented. Secondly analysis of test case execution is presented in graphical form.

## 4.1. Demographic Analysis

This section presented the demographic analysis of all companies and respondent involve in our survey. We send our survey questionnaire online through social networks and personal contacts in which we received 133 responses. The following countries participate including Pakistan, India, USA (Washington, New York, San Francisco) UK, Italy, Egypt, Switzerland, Spain, Brazil, Singapore, Belarus, Germany, France and Belgium. The participated companies in our survey is Net sol Technologies, Aaj technologies, polycom, IBM System, Crestech, Cap Gemini, Belagcom, Quasus, Siemens.. The respondent designation is software quality assurance engineer, Test Managers, Test engineer, Quality analyst, Test automation engineers etc.

## 4.2. Quality Criteria

To establish credibility of feature and core phases we apply quality criteria basis upon of two variables which includes certifications of organizations and experience of respondent.

### 4.2.1. Certification of Organization

Among all responded organizations those which were either certified by CMMI (Capability Maturity Model Integration) at level 1, 2, 3 and 4, 5 and by ISO (International Organizations for Standardizations) were identifies. We received 17 responses from Level 1, 22 responses from level 2, 24 responses from level 3, 9 responses from level 4 and 16 responses from level 5 organization while from ISO

9001 certified organization we received 43 responses and 90003 we received 23 responses.

### 4.2.2. Experience

All respondents mentioned their years of experience which was then categorized in to low ($1 \leq \text{low} \geq 5$), Medium ($6 \leq \text{Medium} \geq 10$) and High ($\text{High} \geq 11$) based upon on response range. We received 47 responses have 1-5 year experience, 55 response have 6-10 year experience and 26 response have 11-30 year experience.

## 4.3. Identification of Feature Set & Core Phase

| Classification of feature set in to Core phases | Phase I | | | Phase II | | Phase III | | Phase IV | |
|---|---|---|---|---|---|---|---|---|---|
| | Test Execution N | | | Test Results Observation and Recording | | Comparison between Actual &Expected Result | | Test Result Logging | |
| Feature Set | Test Case | Test Set | Test Procedure | Observe Actual result | Record Actual result | Compare Actual & Expected Result | Determine Test Result | Create Detail Test Execution Log | Create Detail Test execution Log |

Identification of Feature(s) Set & Core Phases

Table 4.3 answers RQ1 through identification of both feature set and core phases and categorization of features set against core phases. We then evaluate the proposed framework through quality criteria so as to establish credibility of feature and core phases in proposed framework. In below graphs on vertical axis expressed the % of respondent of each feature set against % of quality attribute on horizontal axis.

## 4.4. Test execution

In this phase tests are executing which may be to perform either through test case, through Test set or through Test procedure

### 4.4.1. Through Test Case Execution

Test case contain preconditions, inputs, and expected results, developed to drive the execution of a test item to meet test objectives.

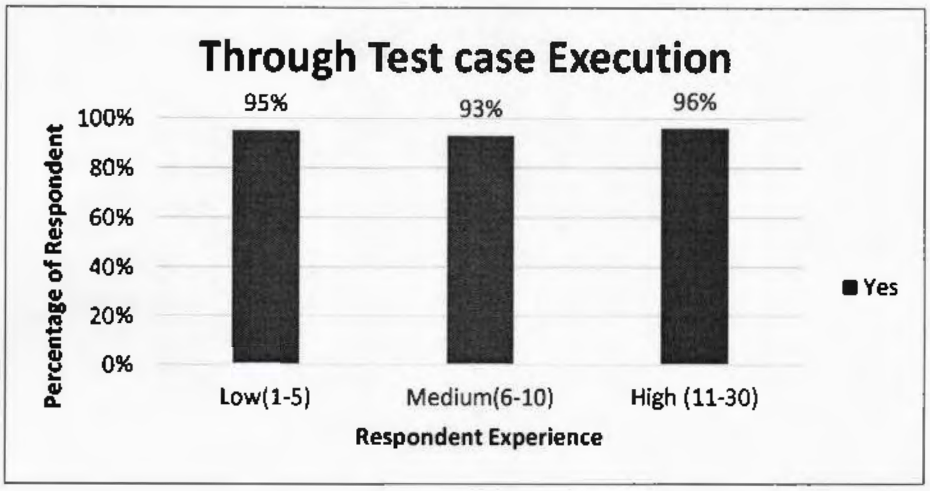

**Figure 4-1 Test case execution-experience**

Fig 4.1 explains that respondents with low medium and high experience support test case execution up to 95%, 93%, and 96% respectively.

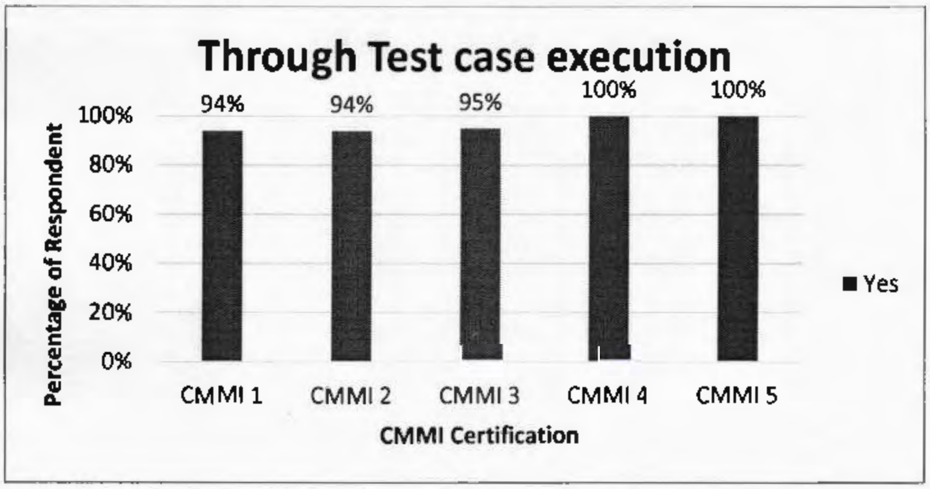

**Figure 4-2 Test case execution-CMMI certification**

Fig 4.2 explain that CMMI4, CMMI5 100% support Test case execution while CMMI2, CMMI3 support 94% and 95%.

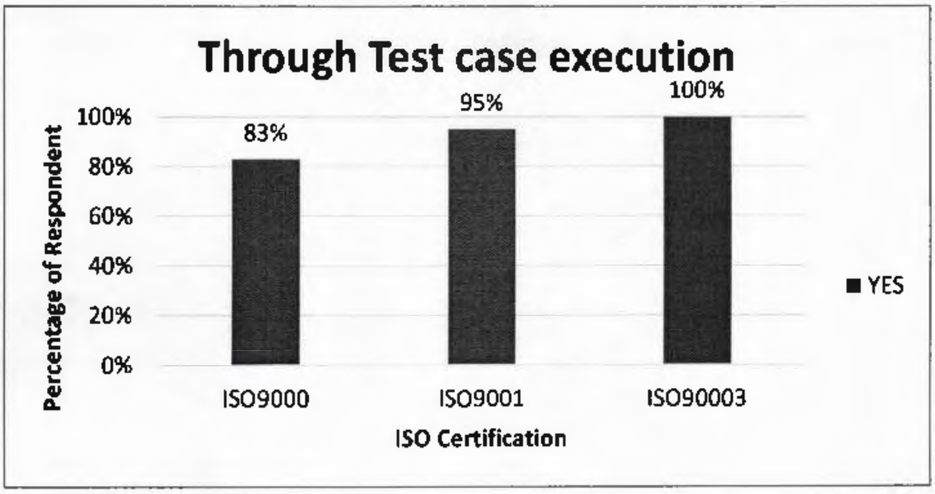

Figure 4-3 Test case execution-ISO certification

Fig 4.3 explain that ISO90003 100% support Test case execution while ISO9001, ISO9000 support 95% and 83% .

### 4.4.2. Through Test Suite/Test Set Execution

Test suite/Test set is collection of test cases have specific test objective

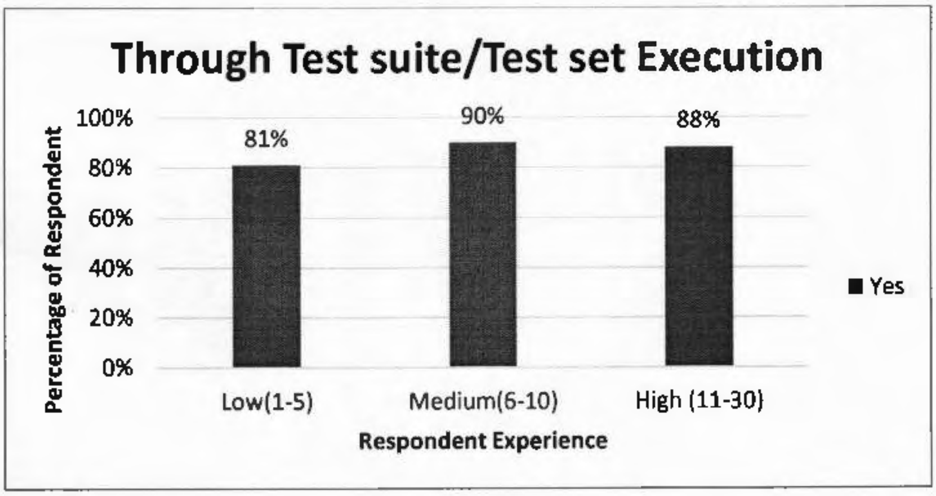

Figure 4-4 Test suite/Test set execution-Experience

Fig 4.4 shows that respondents with Medium experience support 90% test suite execution while Low and High experience support up to 81%, 88% respectively.

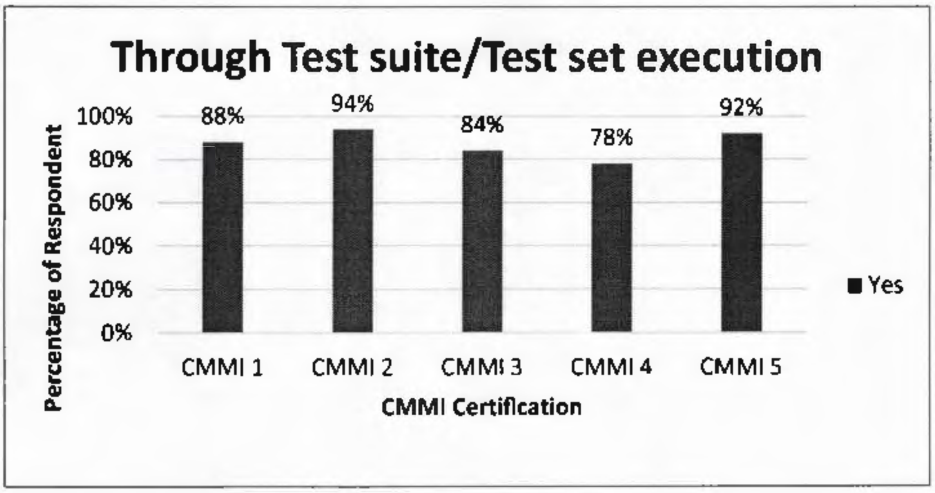

Figure 4-5 Test suite/Test set execution-CMMI certification

Fig 4.5 explain that CMMI2, CMMI5 94%,92%, support test suite/test set execution while CMMI1,CMMI3,CMMI4 support 88%,84% and 78%.

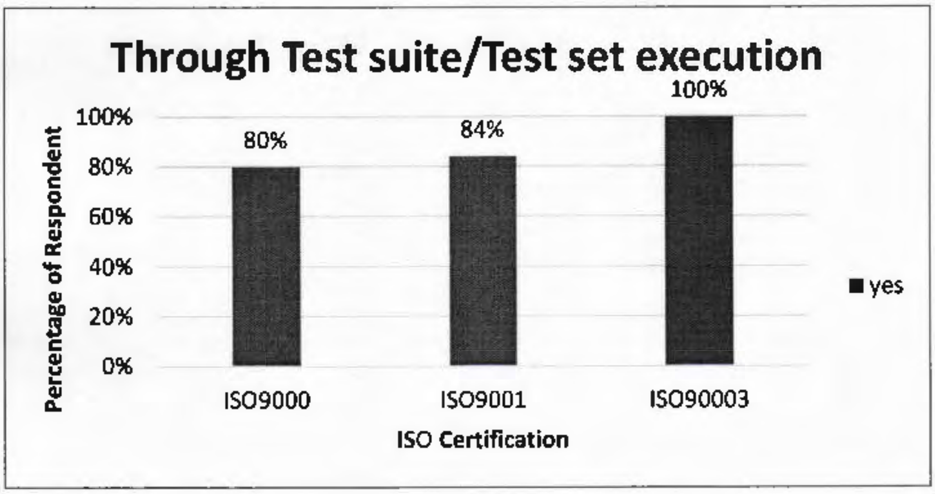

Figure 4-6 Test suite/Test set execution-ISO certification

Fig 4.6 that ISO9003 100%, support test suite/test set execution while ISO9000, ISO9001 support 80% and 84%.

### 4.4.3. Through Test Procedure Execution

Ordering test case with in a test set in accordance to dependencies set/describe in precondition, post conditions and other testing requirement

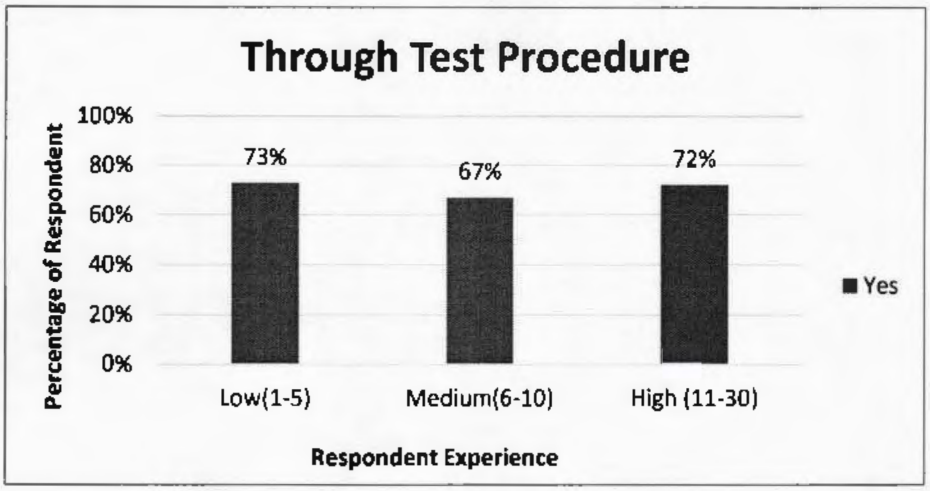

Figure 4-7 Test Procedure-Experience

Fig 4.7 describe that respondents with low, high experience support test procedure execution up to 73%, 72 % while high experience respondent support up to 67%.

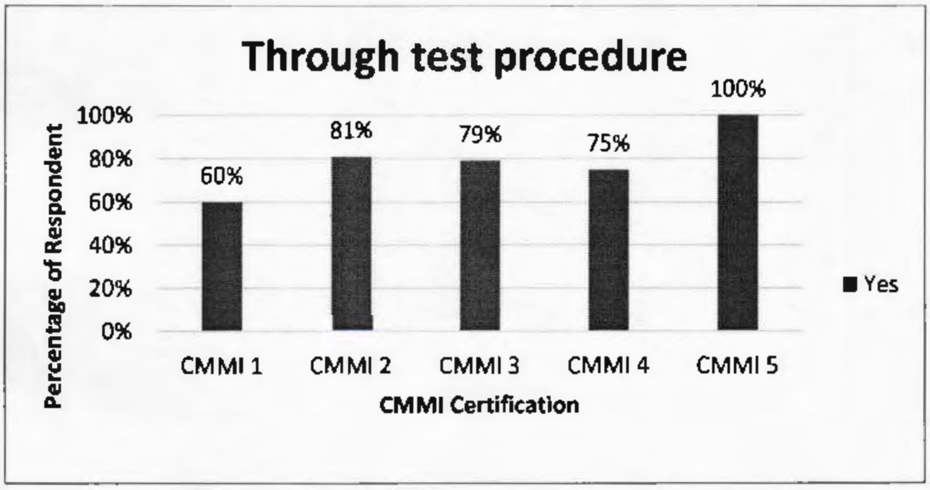

Figure 4-8 Test Procedure-CMMI Certification

Fig 4.8 explain that CMMI 5 100% support test procedure execution while CMMI2 support 81%, CMMI3 and CMMI4 support 79%, 75% .

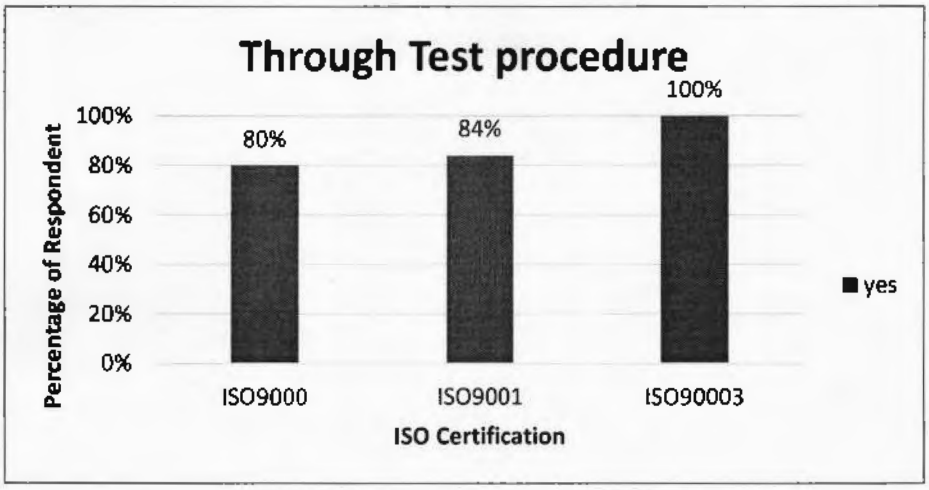

Figure 4-9 Test procedure-ISO certification

Fig 4.9 explain that ISO 90003 100% support test execution is perform through test procedure while ISO9000, ISO9001 support 80% and 84%.

## 4.5. Test Results Observation and Recording

During test case execution actual result are observe and recorded

### 4.5.1. Observe Actual Result

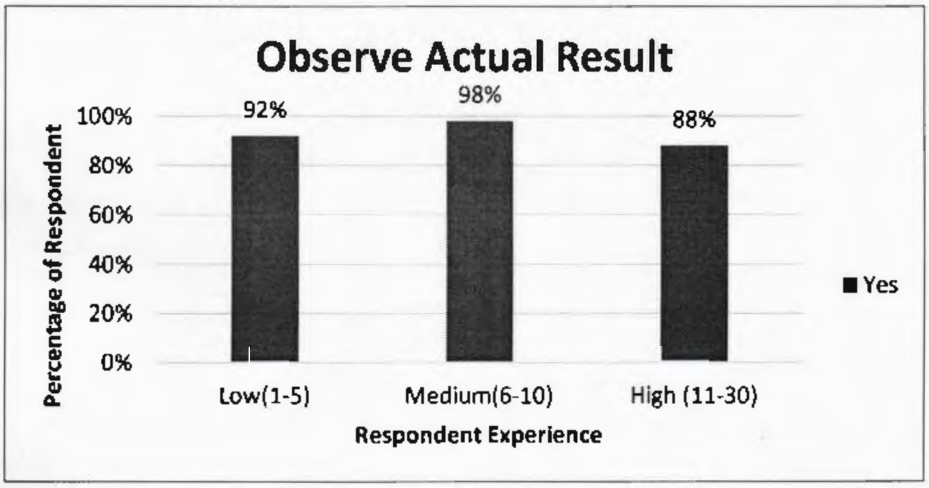

Figure 4-10 Observe actual result-Experience

Fig 4.10 shows that respondents with Medium experience support 98% that actual result are observed. While Low and High experience support up to 92 % and 88% respectively.

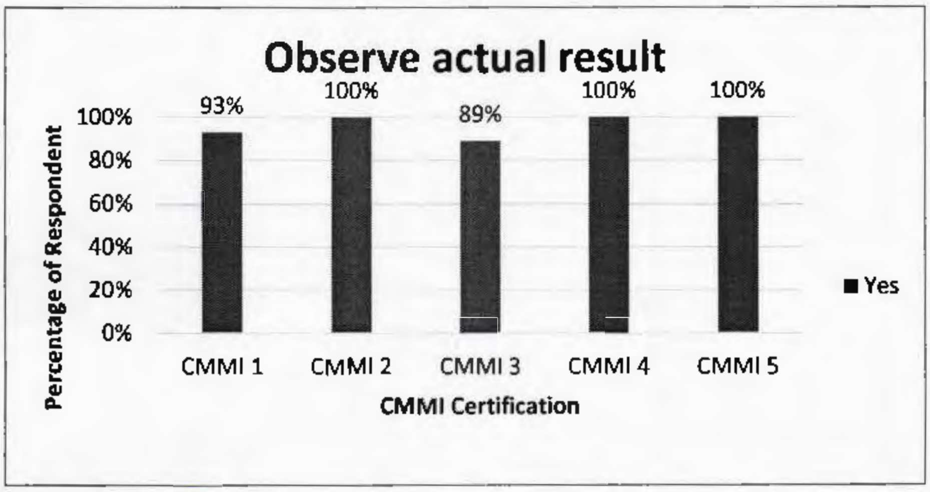

Figure 4-11 Observe actual result-CMMI certification

Fig 4.11 shows that CMMI2, CMMI4, CMMI5 100% support to observe actual result while CMMI 1 and CMMI3 support to 93% and 89%.

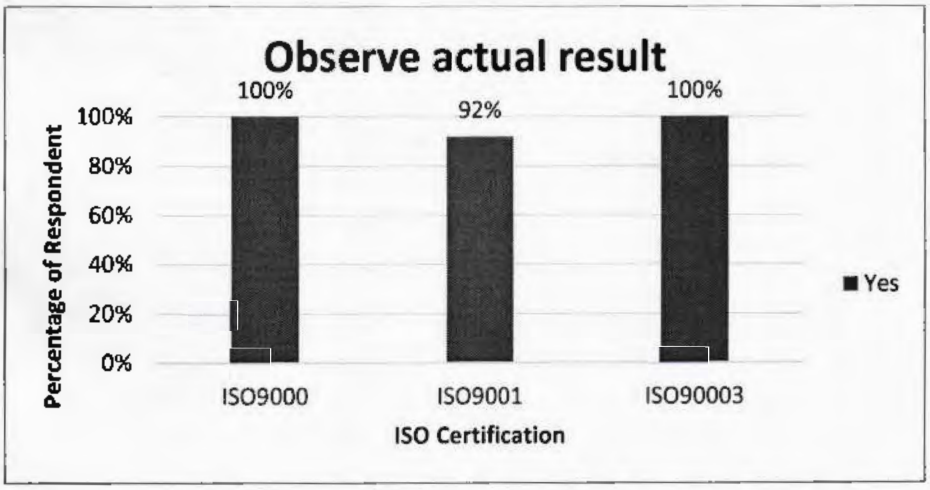

Figure 4-12 Observe actual result-ISO certification

Fig 4.12 explain that ISO 9000, ISO 90003 100% support to observe actual result while ISO9001 support 92%.

### 4.5.2. Record Actual result

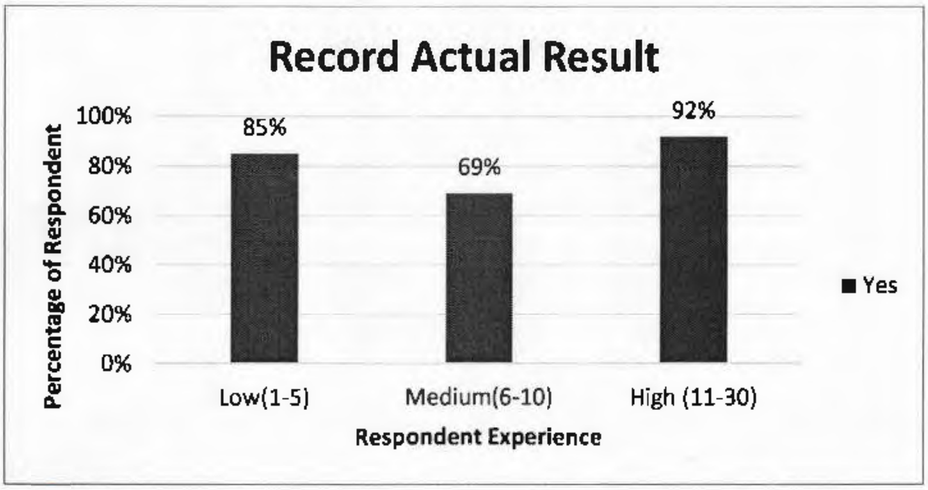

Figure 4-13 Record actual result-Experience

Fig 4.13 illustrates that respondents with large experience support 92% record actual result while Low experience support 85% and Medium experience support up to 69%.

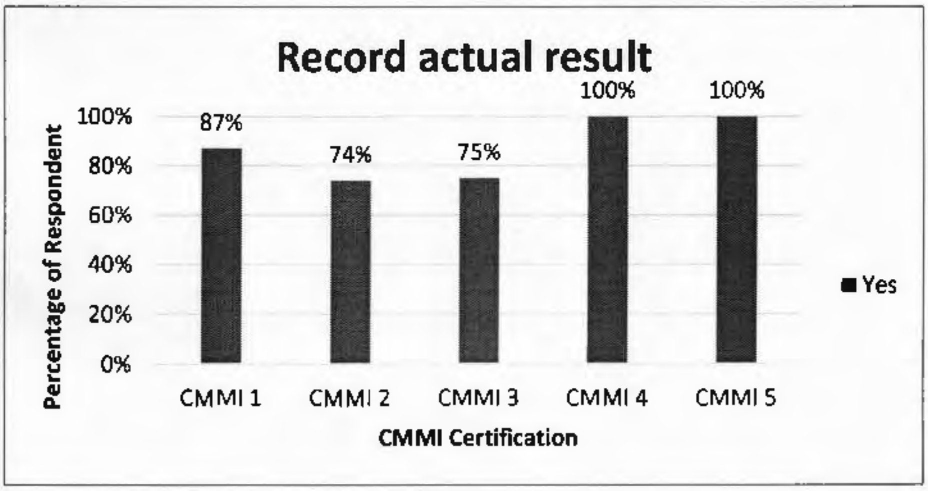

Figure 4-14 Record actual result-CMMI Certification

Fig 4.14 shows that CMMI4, CMMI 5 100%, support to record actual result while CMMI1, CMMI2, CMMI3 support 87%, 74% and 75%.

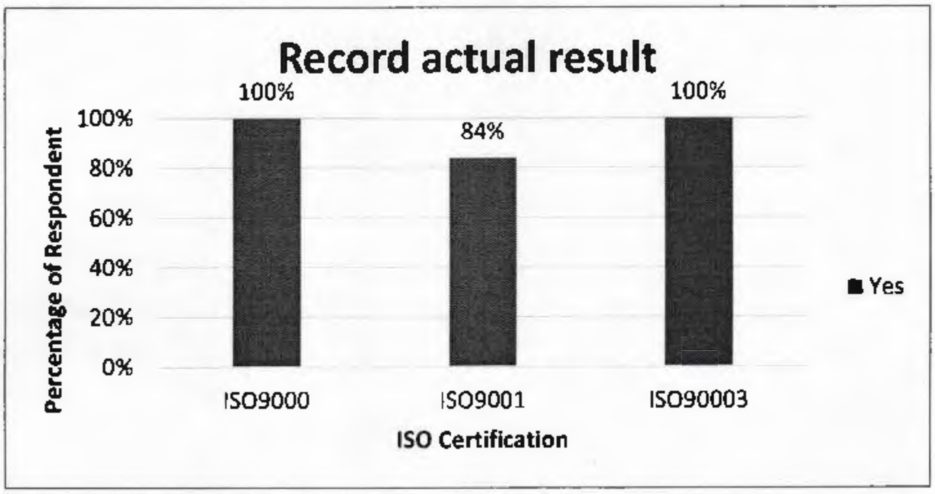

Figure 4-15 Record actual result-ISO certification

Fig 4.15 illustrates that ISO9000, ISO90003 100%, support to record actual result while ISO9001 support 84%.

## 4.6. Comparison between Actual &Expected Result:

In this phase the actual and expected results are compare and then determine either the retest will be required or not.

### 4.6.1. Compare Actual and Expected Result

The result we obtain after test execution are actual result. These actual result are compared with expected result

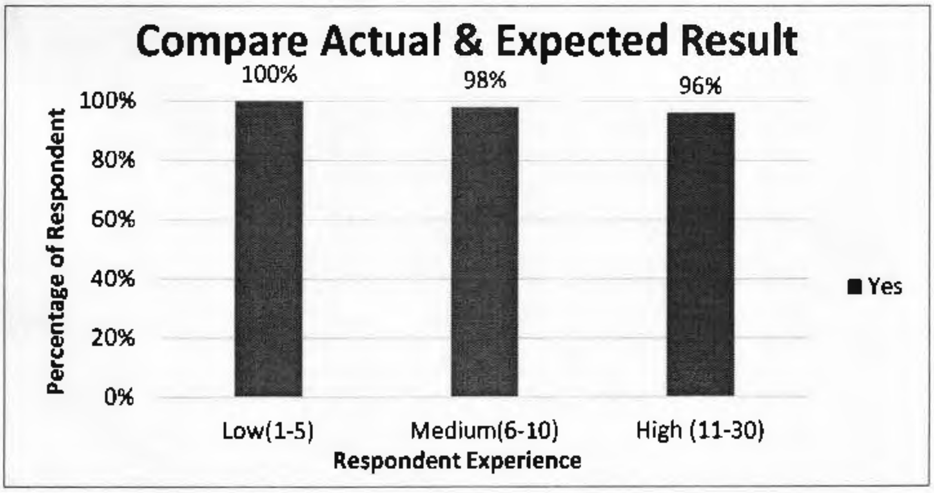

Figure 4-16 Compare actual & expected result-Experience

Fig 4.16 show that respondents with low experience 100% support compare actual and expected result while medium and high experienced respondent support up to 98%, 96% respectively.

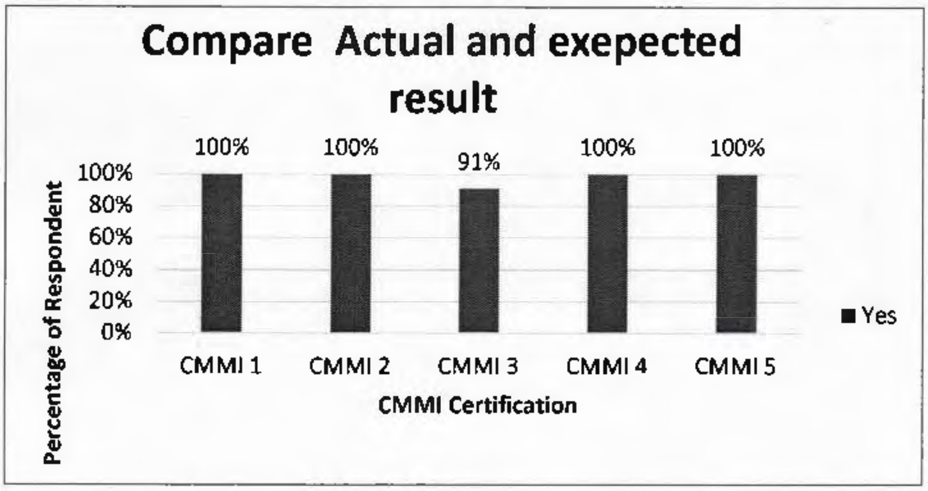

Figure 4-17 Compare actual & expected result-CMMI certification

Fig 4.17 shows that CMMI 1, CMMI 2, CMMI 4, CMMI 5 100% support to compare actual and expected result while CMMI 3 support 91%.

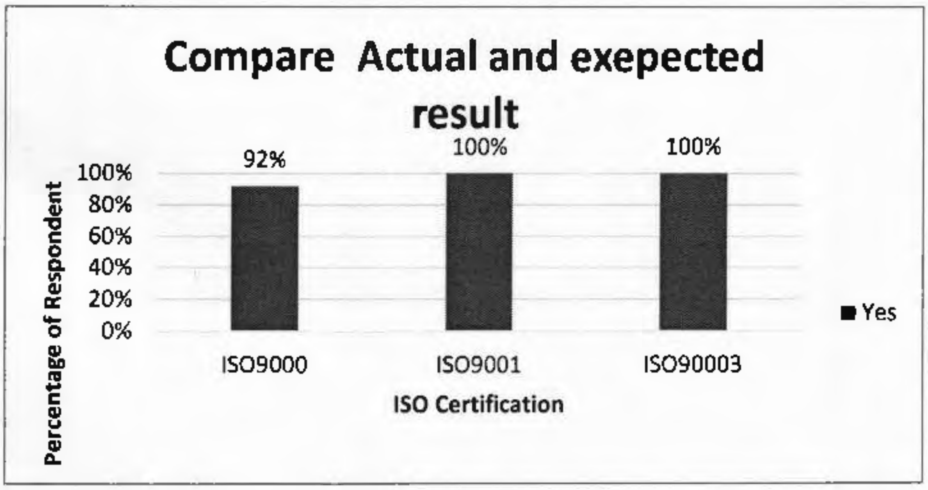

Figure 4-18 Compare actual & expected result-ISO certification

Fig 4.18 illustrates that ISO 9001, ISO 90003 100% support to compare actual and expected result while ISO 9000 support 92%.

### 4.6.2. Determine Test Result

To check the test case execution test result that either the retest will be required or not

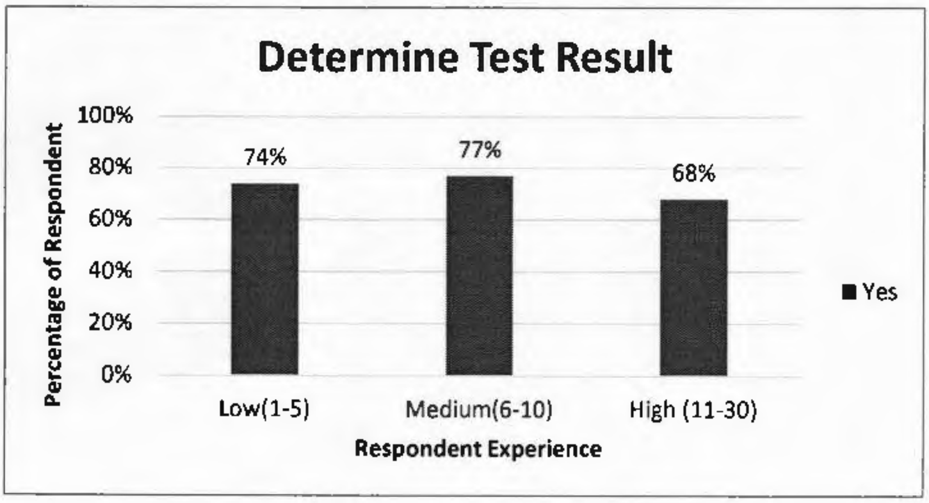

Figure 4-19 Determine test result-Experience

Fig 4.19 shows that respondents with medium experience support 77% Determine test result while low experience support 74% and high experienced respondent support up to 68%.

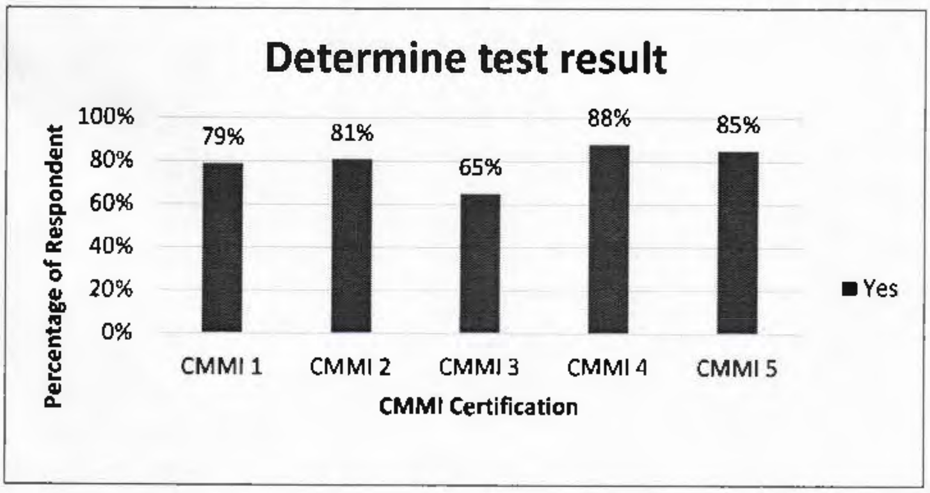

Figure 4-20 Determine test result-CMMI certification

Fig 4.20 shows that CMMI4 88%, CMMI 5 85%, CMMI2 81%, CMMI 1 79%, CMMI 3 65% support to determine test result.

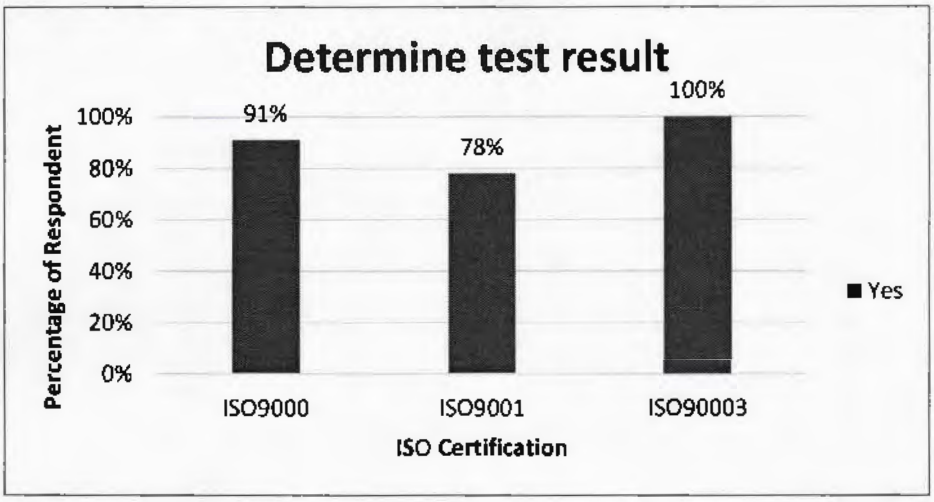

Figure 4-21 Determine test result-ISO certification

Fig 4.21 explains that ISO90003 100% support to determine test result while ISO9000 support 91% and ISO9001 support 78%.

## 4.7. Test Results Logging

In this phase the test case execution result are recorded which may be detail test execution logs or short test execution logs.

### 4.7.1. Create Detail Test Execution logs

Detail test execution logs are those which provide log/result with detail description

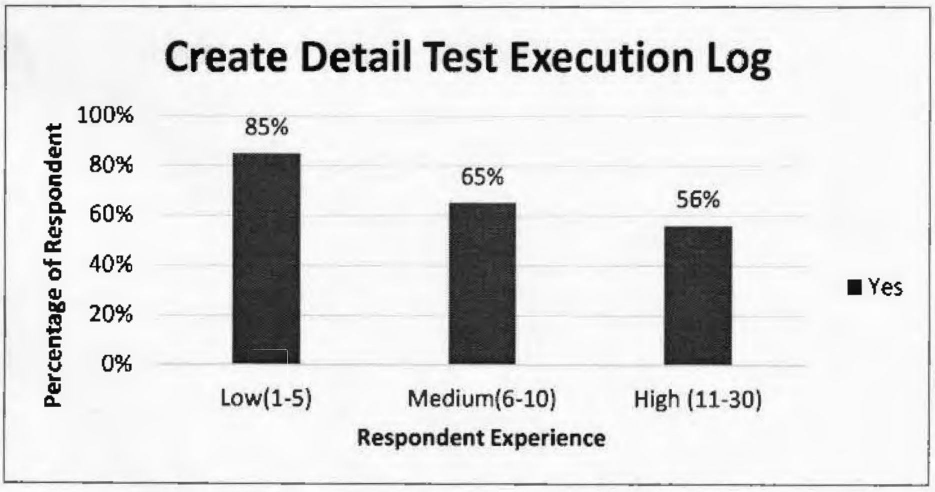

Figure 4-22 Create detail test execution log-Experience

Fig 4.22 illustrates that respondents with low experience support 85% to create detail execution log test suite while medium experience support 65% and high experience support up to 56%.

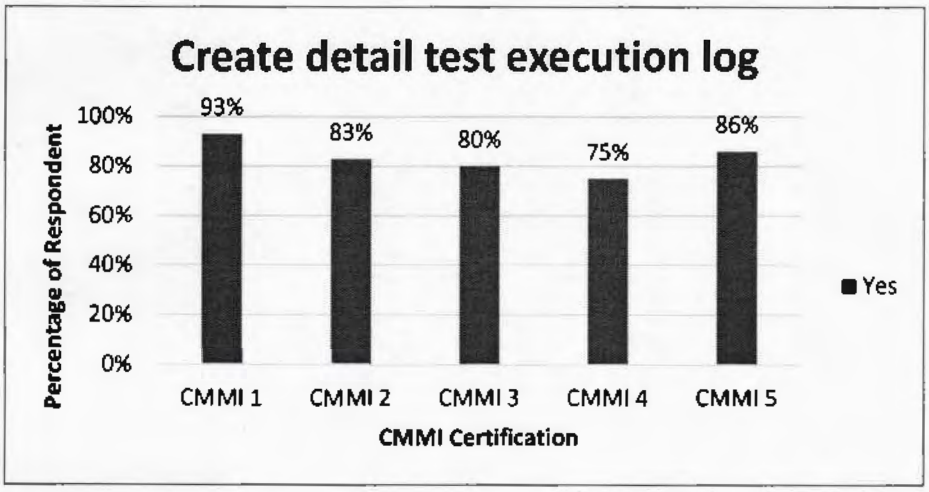

Figure 4-23 Create detail test execution log-CMMI certification

Fig 4.23 shows that CMMI 1 93%, CMMI2 83%, CMMI 3 80%, CMMI 5 86% and CMMI4 75% support to create detail test execution log.

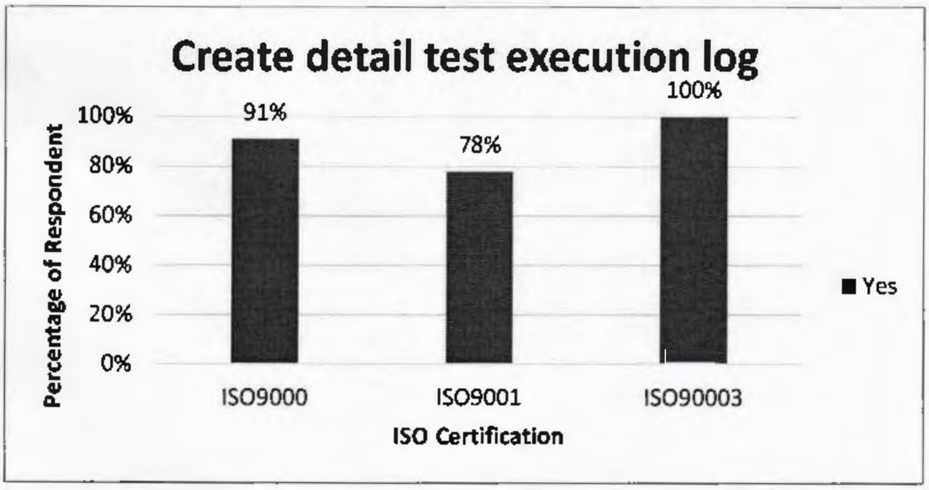

Figure 4-24 Create detail test execution log-ISO certification

Fig 4.24 explain that ISO90003 100% support to create detail test execution log while ISO9000 support up to 91% and ISO9001 support 78%.

### 4.7.2. Create Short Test Execution log

Short test execution logs only provide brief result not detail description.

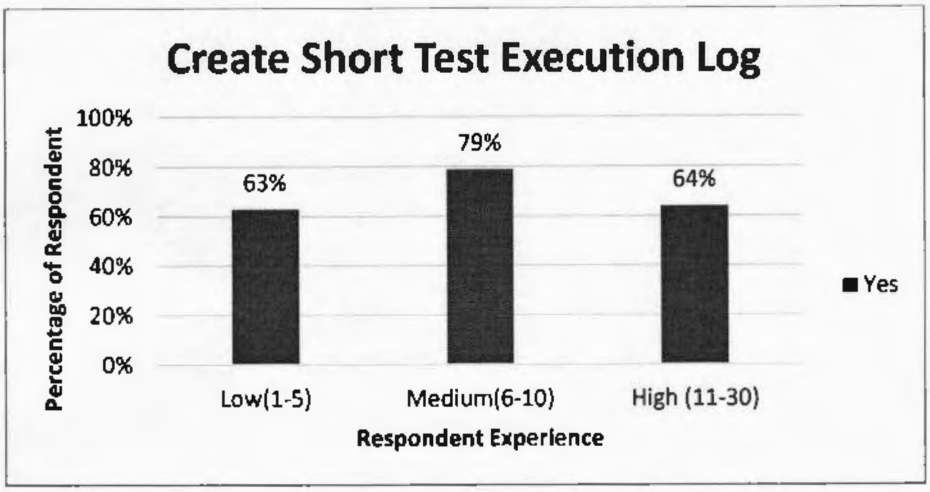

Figure 4-25 Create short test execution logs-Experience

Fig 4.25 explains that respondents with medium experience support 79% create short execution log while low and high experienced respondent support up to 63%, 64% respectively.

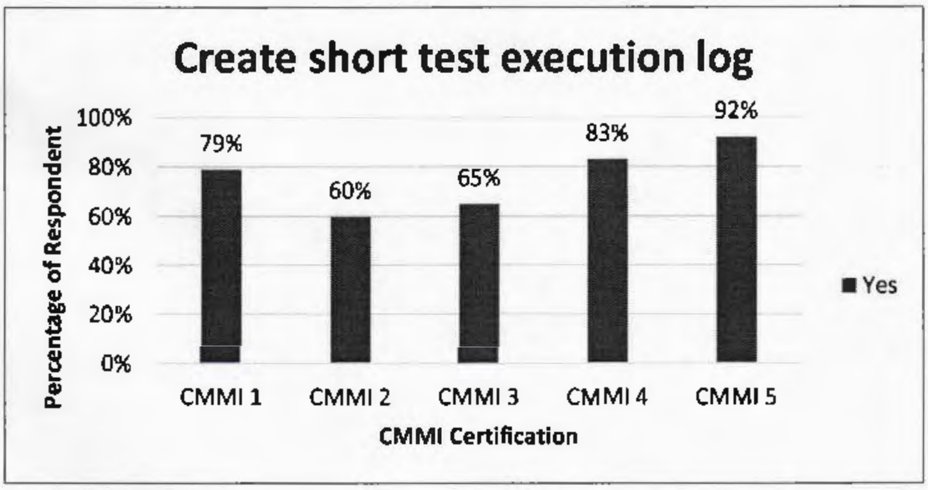

Figure 4-26 Create short test execution logs-CMMI Certification

Fig 4.26 explain that CMMI5 92%, CMMI4 83%, CMMI1 79%, CMMI3 65% and CMMI 2 60% support to create short test execution log

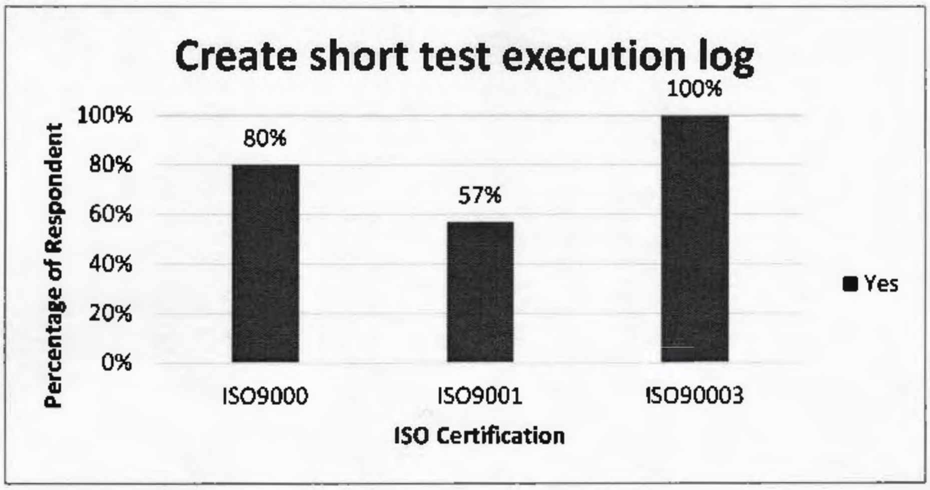

Figure 4-27 Create short test execution logs-ISO certification

Fig 4.27 explain that ISO 90003 100%support to create short test execution log while ISO9000 support 80% and ISO9001 support up to 57%.

## 4.8. How test case execution perform in your organization

Some tools used code driven testing approach and some record & play back test approach.

### 4.8.1. Code driven testing

Code driven testing, tester/developer have to write code to test different units of software

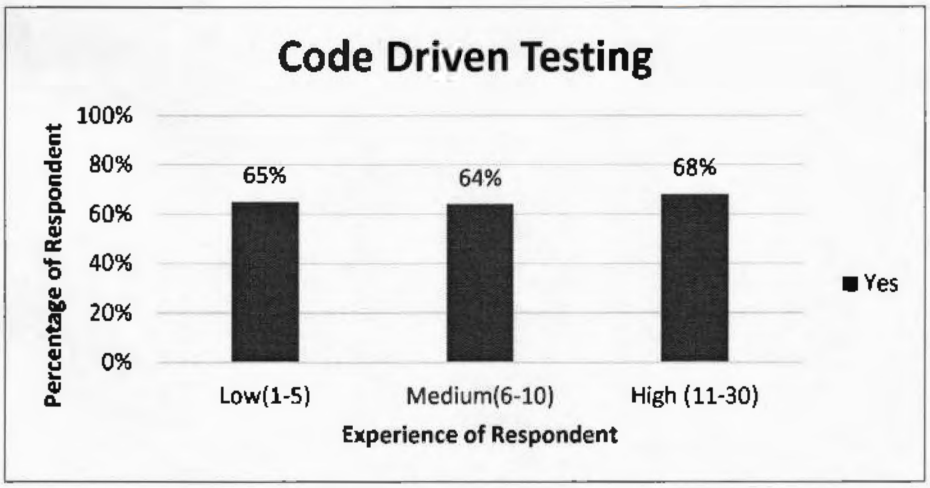

Figure 4-28 Code driven testing-Experience

Fig 4.28 explains that respondents with low experience, medium experience and high experience support Code driven testing up to65%, 64%, and 68% respectively.

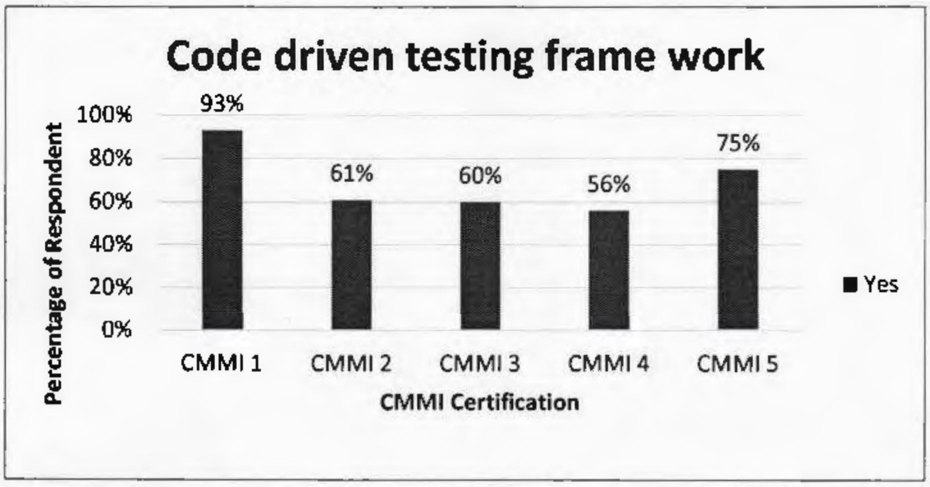

Figure 4-29 Code driven testing-CMMI certification

Fig 4.29 illustrates that CMMI 1 93%, CMMI 5 75%, CMMI 2 61%, CMMI 3 60% support to perform code driven testing.

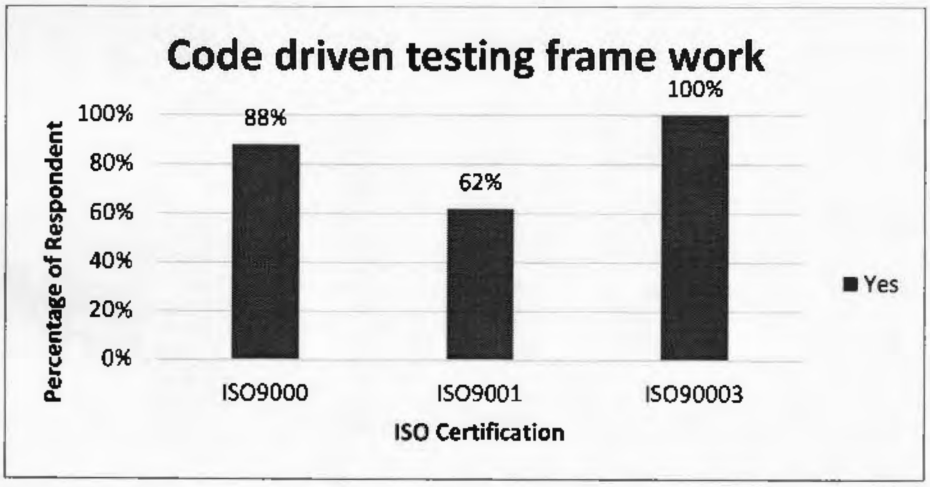

Figure 4-30 Code driven testing-ISO certification

Fig 4.30 explain that ISO 90003 100%, support to perform code driven testing while ISO 9000 support 88% and ISO 9001 support 62%

### 4.8.2. Record &playback testing

Generates user interface events such as keystrokes, observe the changes, to validate that the observable behavior of the program is correct

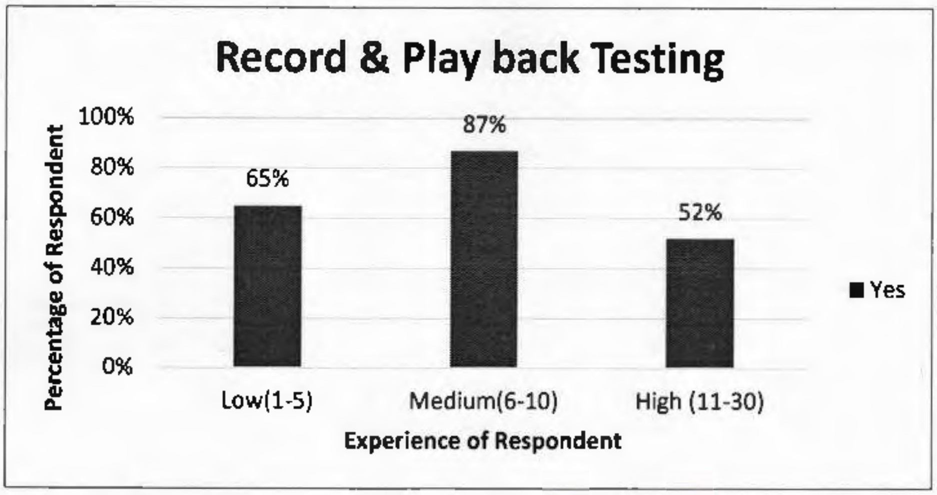

Figure 4-31 Record & play back testing-Experience

Fig 4.31 shows that respondents with medium experience support 87% record & play back testing while low experience and high experience support up to 65%, 52% respectively.

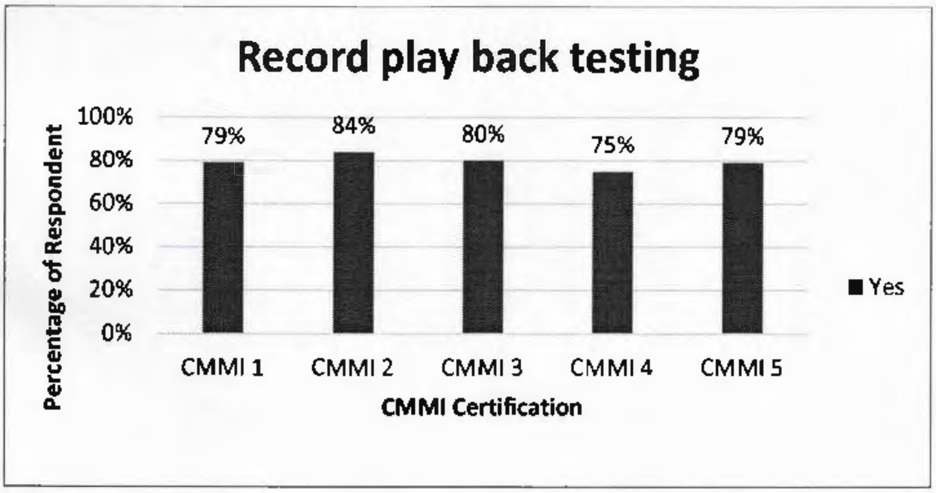

Figure 4-32 Record & play back testing-CMMI certification

Fig 4.32 illustrates that CMMI 2 84%, CMMI 3 80%, CMMI 1, CMMI 5 79%, CMMI 4 75% support to perform record play back testing.

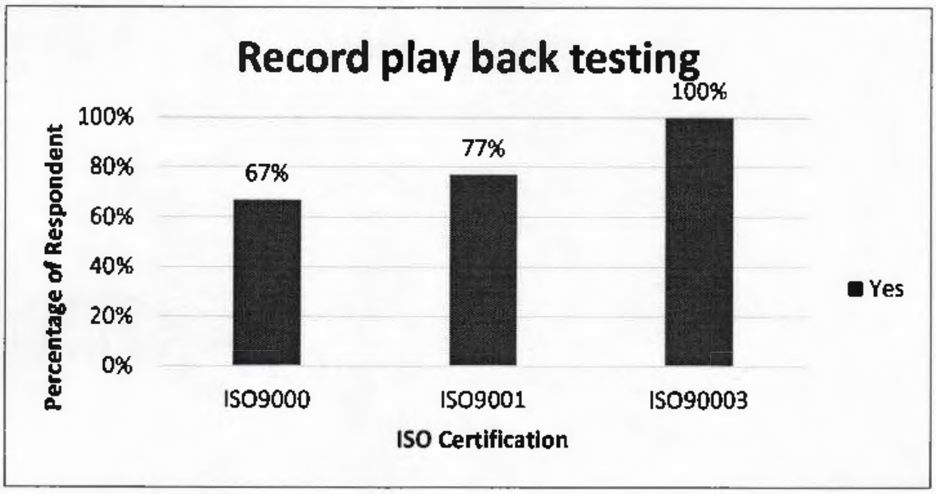

**Figure 4-33 Record & play back testing-ISO certification**

Fig 4.33 explains that ISO 90003 100%, support to perform record play back testing while ISO 9001 support 77% and ISO 9000 support 67%.

## 5. Introduction:

In this chapter we have discussed our proposed framework for automated test case execution tools development. Our industrial feedback ranges from 40% to 100% which we categorized our data in to High, Medium and Low.

100 < High> 80

80 < Medium>60

60 < low> 40

### 5.1. Identification of Feature Set & Core Phase:

| **RQ1.What is maximum set of feature(s) and core phases of test case execution tool(s) and their classification of feature set in to core phases?** | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Classification of feature set in to Core phases** | **Phase I** | | | **Phase II** | | **Phase III** | | **Phase IV** | |
| | Test Execution n | | | Test Results Observation and Recording | | Comparison between Actual &Expected Result | | Test Result Logging | |
| **Feature Set** | Test Case | Test Set | Test Procedure | Observe Actual result | Record Actual result | Compare Actual & Expected Result | Determine Test Result | Create Detail Test Execution Log | Create Detail Test execution Log |

Table 5.1 Feature(s) set and Core Phases of test case execution tools

After identification of both feature set and core phases, quality criteria is then applied so as to establish credibility of feature and core phases. In below tables High, Medium, Low shows the level of requirement of different organization for test case execution feature(s) set.

### 5.2. Test Execution

This phase suggests that a comprehensive feature(s) set is mandatory for test cases execution.

| **Test Execution** | **RQ2.How can we evaluate proposed framework?** | | | |
|---|---|---|---|---|
| | **ISO Certification** | | **CMMI** | **Experience** |
| | **Iso9001** | **Iso90003** | **Level** | **High** |

| | | | 5 | |
|---|---|---|---|---|
| **Through Test case** | High | High | High | High |
| **Through Test suite/ Test set** | High | High | High | High |
| **Through Test procedure** | High | High | High | Medium |

Table 5.2 Test Execution

Table 5.1 shows that above phase and its feature(s) set was highly endorsed by CMMI level 5 and ISO 9001 and 90003 certified organizations. Besides it was endorsed by highly experiences quality engineers. Thus above phase and its feature(s) set integrated in the proposed framework.

## 5.3. Test Results Observation and Recording

This phase suggests that a comprehensive feature(s) set is mandatory for test cases execution.

| **Test Results Observation and Recording** | RQ2.How can we evaluate proposed framework? | | | |
|---|---|---|---|---|
| | **ISO Certification** | | **CMMI** | **Experience** |
| | **Iso9001** | **Iso90003** | **Level 5** | **High** |
| **Observe Actual Result** | High | High | High | High |
| **Record Actual result** | High | High | High | High |

Table 5.3Observe/Record Actual result

Table 5.2 shows that above phase and its feature(s) set was highly endorsed by CMMI level 5 and ISO 9001 and 90003 certified organizations and by highly experiences quality engineers. Thus above phase and its feature(s) set integrated in the proposed framework.

## 5.4. Comparison between Actual &Expected Result:

This phase suggests that a comprehensive feature(s) set is mandatory for test cases execution.

| **Comparison between Actual & expected result** | RQ2.How can we evaluate proposed framework? | | | |
|---|---|---|---|---|
| | **ISO Certification** | | **CMMI** | **Experience** |
| | **Iso9001** | **Iso90003** | **Level 5** | **High** |
| **Compare Actual & expected result** | High | High | High | High |

| | | | | |
|---|---|---|---|---|
| **Determine test result** | High | High | High | Medium |

Table 5.4 Compare Actual & Expected result

Table 5.4 shows that above phase and its feature(s) set was highly endorsed by CMMI level 5, ISO 9001 and 90003 certified organizations. Besides it was endorsed by highly experiences quality engineers. Thus above phase and its feature(s) set integrated in the proposed framework.

## 5.5. Test Results Logging

This phase suggests that a comprehensive feature(s) set is mandatory for test cases execution.

| **Test Results Logging** | RQ2.How can we evaluate proposed framework? | | | |
|---|---|---|---|---|
| | **ISO Certification** | | **CMMI** | **Experience** |
| | **Iso9001** | **Iso90003** | **Level 5** | **High** |
| **Create detail**<br>**Test execution log** | Medium | High | High | Low |
| **Create short execution log** | Low | High | High | Medium |

Table 5.5Test case execution Results

Table 5.5 shows that above phase and its feature(s) set was highly endorsed by CMMI level 5 and 90003 certified organizations. Besides it was endorsed by highly experiences quality engineers. Thus above phase and its feature(s) set integrated in the proposed framework.

## 5.2. Proposed Framework

Several number of test case execution tools, which implement a verity of feature, are used in industry to improve the quality of software product. However, feature set of current tools are misalign with current requirement of software industry. This is because current tools do not any framework or a set of guidelines validated as per current needs of software industry. We therefore propose such a framework for test case execution tool(s) whose phases and their feature set are designed based upon current industrial feedback.

Framework for test case execution tool(s) as illustrated in the figure below has core phases which are further explained through feature set. Besides they are endorsed through rigorous quality criteria i.e. highly experienced software and quality engineers and organizations which are CMMI level 5 and ISO certified. Using this framework the developers and organization will choose suitable phase(s) which are further explained through comprehensive feature set for the development of test case execution tools. The proposed framework consists of following four phases. The arrows denote the relationship between each phase.

- First phase describe show is Test case execution performed e.g. test case. Test suite/test set, test procedure.
- Actual Result Phase. Observes and record actual results obtain from test execution phase.
- Comparison Phase. It draws a comparison between actual and expected result will determine whether retesting is required or not.
- Record Test result Phase. It record detail and short information generated in the process of test execution.
- Code driven. This is applicable in scenarios where tester/developer have to write code to test different units of software.
- Replay & play back. This is applicable in scenarios where tester/developer interactively record user actions and replay them back any number of times.

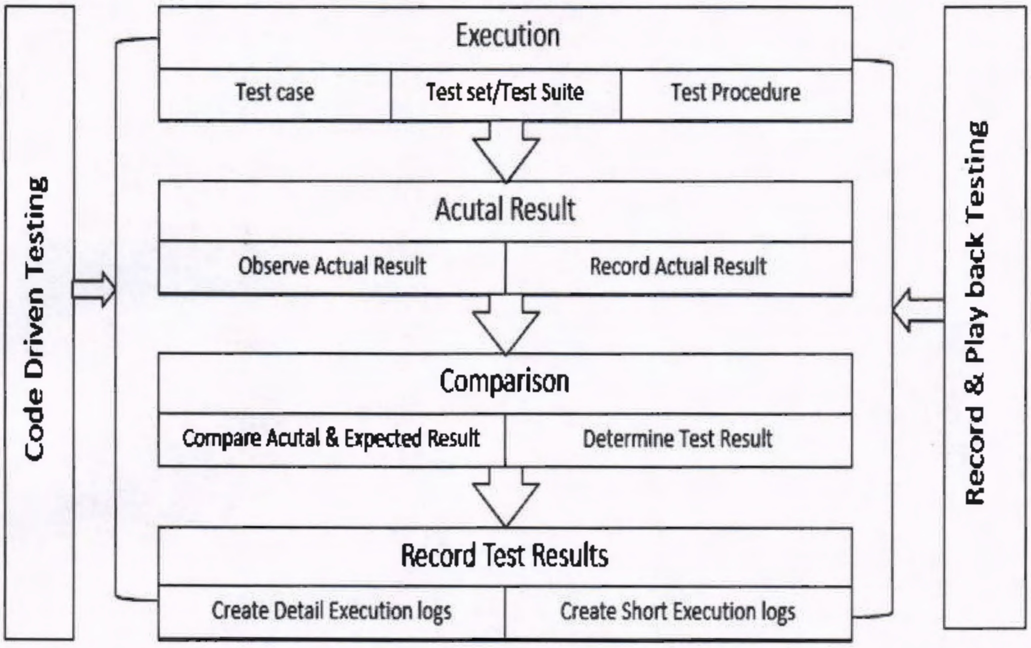

**Figure 5.2-1 Proposed Framework for test case execution tools**

Proposed framework provides architectural foundation for both code driven and record & play back test execution tools in order to define their scope with respect to phases and its feature set instead of implementing features which may belong to various phases. Proposed framework is also advantageous as it provides a layout to developer and organization to choose suitable feature for the development of tools. This research can be extended through development of a rigorous research instrument which involves such commercial tools that are not published in academia. This will bridge the gap between academia and industry through development of test case execution tool(s).

# Chapter 6

# Conclusion

### 6.1. Conclusion

Several number of test case execution tools, which implement a verity of feature, are used in industry to improve the quality of software product. However, feature set of current tools are misalign with current requirement of software industry. This is because current tools do not any framework or a set of guidelines validated as per current needs of software industry. We therefore propose such a framework for test case execution tool(s) whose phases and their feature set are designed based upon current industrial feedback.
Such framework will bridge the gap between academia and industry. it provide architectural foundation for test case execution tools in order to define their scope with respect to phases and its feature set instead of implementing features which may belong to various phases. The main advantage of proposed framework is that it provides a layout to developer and organization to choose suitable feature for the development of tools. This will improve the development of test case execution tools.

### 6.2. Future work

For more rigorous research instrument, study commercial and open source tools which are not published in academia. A correlational analysis of phases and its set of parameters against level of organization i.e. small, medium or large, type of certification i.e. CMMI or ISO and respondents experience.

# References

**References:**

[1] Lee, J., S. Kang, and D. Lee. "*Survey on software testing practices.*" IET software 6, no. 3, 275-282, 2012.

[2] Vos, Tanja EJ, Beatriz Marin, Maria Jose Escalona, and Alessandro Marchetto. "*A methodological framework for evaluating software testing techniques and tools.*"12th IEEE International Conference in Quality Software (QSIC), pp. 230-239, 2012.

[3] Leckraj Nagowah and Purmanand Roopnah. "*AsT -A Simple Automated System Testing Tool.* "IEEE, Vol: 9, pp.301 – 306, 2010.

[4] Wang, Fei, and Wencai Du. *"A test automation framework based on web."* Published In Computer and Information Science (ICIS), IEEE/ACIS, pp. 683-687, 2012.

[5] Stoyanova, Vera, Dessislava Petrova-Antonova, and Sylvia Ilieva. "*Automation of Test Case Generation and Execution for Testing Web Service Orchestrations."* Published In Service Oriented System Engineering (SOSE), pp. 274-279, 2013.

[6] Tang, Jingfan. *"Towards Automation in Software Test Life Cycle Based on Multi-Agent."* Published In Computational Intelligence and Software Engineering (CiSE), IEEE, pp. 1-4, 2010.

[7]Kim, Eun Ha, Jong Chae Na, and Seok Moon Ryoo. *"Test automation Framework for implementing continuous integration."* Published In Information Technology: New Generations, IEEE, pp. 784-789, 2009.

[8] Parveen, Tauhida, Scott Tilley, Nigel Daley, and Pedro Morales. *"Towards a distributed execution framework for JUnit test cases."* Published In Software Maintenance, ICSM IEEE,pp. 425-428. 2009.

[9]Gao Jing ; Lan Yuqing *"Agent Based Distributed Automated Testing Executing Framework."* Published in Computational Intelligence and Software Engineering, IEEE, pp.1-5, 2009.

[10] Abdou, Tamer, Peter Grogono, and Pankaj Kamthan. *"A Conceptual Framework for Open Source Software Test Process.* "Published In Computer Software and Applications Conference Workshops (COMPSACW), pp. 458-463, 2012.

[11]Maria Christakis, Peter Müller, and *ValentinWüstholz "Collaborative Verification and Testing with Explicit Assumptions."* International Symposium Formal Methods,pp. 27-31 2012

[12]S.J.Galler and B.K.Aicherning. *"Survey on test data generation tools: An evaluation of white- and gray-box testing tools for C#, C++, Eiffel, and Java."* International Journal on Software Tools for Technology Transfer, pp.1-25.2013.

[13]N.Tillmann and J.de Halleux. *"Pex–White Box Test Generation for .NET."* In the Proceedings of the 2nd international conference on Tests and proofs, pp.134-153, 2008.

[14] Nikolai Tillmann and Jonathan de Halleux *"Pex–White Box Test Generation for .NET"*, Proceedings of the 2nd international conference on Tests and proofs Springer, pp.134-153, 2008.

[15] K.Pan, X.Wu and T.Xie, *"Generating Program Inputs for Database Application Testing"* published in Automated Software Engineering, Washington.Pp.73-82, 2011

[16]Nikolai Tillmann, Jonathan de Halleux *"White-Box Testing of Behavioral Web Service Contracts with Pex."*International Symposium on Software Testing and Analysis, pp. 47-48, 2008.

[17] Fewster, Mark, and Dorothy Graham." *Software test automation: effective use of test execution tools"*. Reading: Addison-Wesley, ACM, 1999.

[18] Wohlin, Claes, et al. *"Experimentation in software engineering."* Springer, 2012.

[19] Runeson, per, et al. "*Case study research in software engineering: Guidelines and examples.*" John Wiley & Sons, 2012.

[20]Garousi, Vahid, and Junji Zhi. "*A survey of software testing practices in Canada.* "Journal of Systems and Software, Vol. 86, Issue no.5, pp.1354-1376, 2013

[21] Kumar, Ravinder, and Mr. Karambir Singh. *"Enhancing Component Based Testing Using JUnit Tool in Net Beans Environment."* Journal, Vol.2, Issue no. 7, 2012.

[22]Wei, Shi, and Tang Feilong. *"Customization of Test Case Selection Based on Filtering Mechanism in JUnit.* "International Forum on Computer Science-Technology and Applications, IFCSTA, IEEE, Vol. 2, pp. 57-60, 2009.

[23]Wahid, M., and A. Almalaise. *"JUnit framework: An interactive approach for basic unit testing learning in Software Engineering."*, Published In Engineering Education (ICEED), IEEE, pp. 159-164, 2011.

[24]Yuhei Otani, Hiroaki Hashiura, *"A Software Testing Tool with the Function to Restore the State at Program Execution of a Program under Test. "*, Published in Journal Education and Information Technologies Vol.6,Issue no.1,2012

[25]Csallner, Christoph, and Yannis Smaragdakis. *"JCrasher: an automatic robustness tester for Java."* Published in Software: Practice and Experience, Vol.34, Issue no. 11, pp. 1025-1050, 2004.

[26] de Castro, Andreza MFV, Gisele A. Macedo, Eliane F. Collins, and Arilo C. Dias-Neto. "*Extension of Selenium RC tool to perform automated testing with databases in web applications.* "Proceedings of the 8th International Workshop on Automation of Software Test, IEEE, pp. 125-131, 2013.

[27] Razak, Rosnisa Abdull, and Fairul Rizal Fahrurazi. *"Agile testing with Selenium.* "Published inSoftware Engineering (MySEC), IEEE, pp. 217-219. 2011.

[28]Massimiliano Di,Simon Poulding."Using search-based algorithms for Ajax event sequence generation during testing. "Springer, pp.103-140, 2011.

[29]Salama, Roberto. *"A regression testing framework for financial time-series databases: an effective combination of fitnesse."* In Proceedings of the ACM international conference companion on Object oriented programming systems languages and applications companion, ACM, pp. 149-154, 2011.

[30] Smith, Michael, James Miller, and Steve Daeninck. *"A test-oriented embedded system production methodology."* Journal of Signal Processing Systems, Vol.56, Issue no. 1, pp: 69-89, 2009.

[31]Tillmann, Nikolai, and Jonathan de Halleux. *"Parameterized Unit Testing with Microsoft PeX."* Long Tutorial in Cite seer, 2010.

[32] Louridas, Panagiotis.*"JUnit: unit testing and coiling in tandem."* Journal, IEEE, Vol.22, Issue no. 4, pp: 12-15, 2005.

[33]S. Hammond, and D. Umphress, "Test driven development: the state of the practice," in Proceedings of the 50th Annual Southeast Regional Conference, Tuscaloosa, Alabama, pp. 158-163, 2012.

[34] Smith, Michael, James Miller, and Steve Daeninck. "A test-oriented embedded system production methodology." Journal of Signal Processing Systems, Vol.56, no. 1, pp: 69-89, 2009.

[35]Csallner, Christoph, Yannis Smaragdakis, and Tao Xie. "DSD-Crasher: A hybrid analysis tool for bug finding." ACM Transactions on Software Engineering and Methodology (TOSEM), Vol.17, no. 2, 2008.

[36] "Software and systems engineering Software testing Part 2: Test processes," ISO/IEC/IEEE 29119-2, pp. 1-68, 2013

# APPENDIX

## Appendix A

## Survey to find the Guidelines for the development of Automated Test Case Execution Tool(s)

I am Asma Habib, Reg # 360-FBAS/MSSE/F12, asma.msse360@iiu.edu.pk, from International Islamic University Islamabad.

This questionnaire is part of my MS research thesis in (Software Engineering) under the supervision of Dr. Rizwan Bin Faiz, rizwan.faiz@riphah.edu.pk, who is a Higher Education Commission (HEC) Approved Supervisor. He is currently working as an Assistant Professor in Faculty of Computing in RIPHAH International University, Islamabad.

As part of our research this questionnaire is requested to be filled up by various software development organizations across the globe. We confirm that all the information provided by respondents will only be used for research purposes and will not be disclosed or shared with any individual or organization under all circumstances.

The objective of research is to propose minimum set of parameters that should be implemented by Automated Test Case Execution Tool. Automated Test Case Execution is a process that execute test cases automatically documented in the test specification(s) in the specified test environment.

**Respondent Information**

1. Name of Respondent

2. Gender of Respondent*Required

○ Male

○ Female

3. Designation of Respondent:*Required

□

4. Experience of Respondent (In year)*Required

□

5. Country*Required

□

6. E-mail of Respondent

□

**Organization Information**

7. Name of your Organization

□

8. Number of employees working in your organization:*Required

Choose an item.

9. Number of people working in quality department:* Required

Choose an item.

10. Which among below mentioned quality standards does your organization belongs to?

10. i. CMMI process maturity level

Choose an item.

10. ii. ISO standard

Choose an item.

10. iii Any other Please specify

□

**Automated Test Case Execution**

11. What automated tool(s) do you use for test case execution?

□ 1. Pex

□ 1.1. Pex VS 2008

□ 1.2. Pex VS 2010

□ 1.3. Pex VS 2013

□ 2. Selenium

□ 2.1. Selenium 2.30

□ 2.2. Selenium 2.40

□ 2.3. Selenium 2.50

□ 3. JUnit

□ 3.1. JUnit 4.8.1

□ 3.2. Junit 4.11

□ 3.3. Junit 4.0

□ 4. JCrasher

□ 5. FitNesse

□ 6. Rational Robot

□ 6.1Rational Robot 7.0.3.7

□ 6.1Rational Robot 7.0.3.9

**Automated Test Case execution**

12. Please specify any other tool and its version used for test case execution in your organization?

12. i What is type of automated test case execution tool(s) as specified in Q.12?

Choose an item.

12. ii..At which level of testing your organization used that tool(s) as specified in Q12?

Choose an item.

12. iii.Which type of testing is performed using that tool(s) as specified in Q.12.?

□ 1. Functional Testing

□ 2. Performance Testing

□ 2.1. Load testing

□ 2.2. Stress testing

□ 2.3. Volume testing

□ 3. Regression Testing

□ 4. Acceptance Testing

□ 4.1. Alpha testing

□ 4.2. Beta testing

□ 5. Security Testing

□ 6. Usability Testing

□ Other: □

**Test Case Execution**

Test Case execution execute test cases documented in the test specification(s) in the specified test environment.

**13. How are test case executed in your organization?**

| | Yes | No |
|---|---|---|
| 1 Through Test Cases Execution (i.e. set of test case preconditions, inputs, and expected results, developed to drive the execution of a test item to meet test objectives) | ○ | ○ |
| 2 Through Test Suite/Test Set Execution (i.e. collection of test cases for the purpose of testing a specific test objective | ○ | ○ |
| 3 Through Test Procedure Execution (i.e. ordering test | ○ | ○ |

| | | |
|---|---|---|
| cases within a test set in accordance to dependencies set/described in pre-conditions, post-conditions and other testing requirements) | | |

**13. i During test case execution do you**

| | Yes | NO |
|---|---|---|
| Observe Actual Result | ○ | ○ |
| Record Actual Result | ○ | ○ |

**13.ii How is test case execution perform in your organization?**

| | Yes | NO |
|---|---|---|
| Code Driven Testing/Framework (i.e. In code driven testing, tester/developer have to write code to test different units of software.) | ○ | ○ |
| Record-Playback Testing/Framework (i.e. generates user interface events such as keystrokes, observes the changes, to validate that the observable behavior of the program is correct. | ○ | ○ |

Please specify any other way of executing Test cases?

**Compare Test Result**

(Test result is Indication of whether or not a specific test case has passed or failed)

14. How are test result compared in your Organization?

| | Yes | No |
|---|---|---|
| Compare Actual and Expected result | ○ | ○ |
| Determine Test Result | ○ | ○ |

Please specify any other way of comparing test results?

**Test Execution Results logs**

(Test execution logs document that records details of the test execution of one or more test procedures)

**15. How is test case execution results recorded in your organization?**

| | Yes | No |
|---|---|---|
| Create Detail Test Execution Logs(i.e. Provide log/result with detail description) | ○ | ○ |
| Create Short Test execution logs (i.e. only provide logs/brief result not detail description) | ○ | ○ |

Please specify any other way of recording test execution results?