

# A Review of Various Software Testing Techniques

Sukhjinder Singh<sup>1</sup>, Er.Sukhpreet Singh<sup>2</sup>, Dr.Madhuchanda Rakshit<sup>3</sup>

M.Phil- Student Computer Application Department/Guru Kashi University,  
Talwandi Sabo, Punjab, India

Assistant Professor, CSE Department /Guru Kashi University,  
Talwandi Sabo, Punjab, India

Assistant Professor, Applied Science Department/Guru Kashi University,  
Talwandi Sabo, Punjab, India

## Abstract

In software development any project, errors in which can be introduced at any stage during development. Though errors are detected after each phase by techniques like inspection, some errors remain undetected. Ultimately, these remaining errors will be removing in the coding .hence; the finally coding is like to have some requirements errors and design errors, in addition to errors introduced during the coding activity. Testing is the activity where the errors remaining from all the previous phases must be detected. Hence, testing perform a very critical role for software assurance quality.

## 1. Introduction

Correctness is the minimum requirement of software. Correctness testing will need some type of oracle, to tell the right behaviour from the wrong one. The tester may or may not know the inside details of the software module under test. [1] Therefore either white box testing or black box testing can be used. Correctness testing has following three forms:-

1) White box testing

2) Black box testing

3) Grey box Testing

1) White box testing White box testing is highly effective in detecting and resolving problems, because bugs can often be found before they cause trouble. [2] White box testing is the process of giving the input to the system and checking how the system processes that input to generate the required output. White box testing is also called white box analysis, clear box testing or clear box analysis. White box testing is applicable at integration, unit and system levels of the software testing process

2) Black box testing Black box testing is testing software based on output requirements and without any knowledge of the internal structure or coding in the program.[2] Basically Black box testing is an integral part of „Correctness testing“ but its ideas are not limited to correctness testing only. The goal is to test how well the component conforms to the published requirement for the component. Black box testing have little or no regard to the internal logical structure of the system, it only examines the fundamental aspect of the system. It makes sure that input is properly accepted and output is correctly produced.

3) Grey box testing The Gray box Testing Methodology is a software testing method used to test software applications. The methodology is platform and language independent. The current implementation of the Gray box methodology is heavily dependent on the use of a host platform debugger to execute and validate the software under test. Recent studies have confirmed that the Gray box method can be applied in real time using software executing on the target platform. Grey box testing techniques combined the testing methodology of white box and black box. Grey box testing technique is used for testing a piece of software against its specifications but using some knowledge of its internal working as well. The understanding of internals of the program in grey box testing is more than black box testing, but less than clear box testing. [1]

### 1.1. Need for Software testing

Software development involves developing software against a set of requirements. Software testing is needed to verify and validate that the software that has been built has been built to meet these specifications. If not we may probably lose our client. So in order to make it sure, that we provide our client a proper software solution, we go for testing. Testing ensures that what you get in the end is what you wanted to build. We check out if there is any problem, any error in the system, which can make software unusable by the client. This helps in the prevention of errors in a system [3]

### 1.2. Goals for software testing

Goals are the output of the software process. Software testing has following goals. 1) Verification and validation Testing can also be used for verifying that the product or the software works as desired and validate whether the software fulfills condition laid down 2) Priority Coverage Testing should be performed in efficient and effective manner within the budget and schedule limits. 3) Balanced Testing process must balance the requirements, technical limitation and user expectation. 4) Traceable Documents should be prepared of both success and failures of testing process. So no need to test same thing again. 5) Deterministic we should know what we are doing, what we are targeting, what will be the possible outcome [4]

### 1.3. Testing principles

Principle is the rule or method in action that has to be followed. Different testing principles are as follows: 1) Test a program to try to make it fail Testing is the process of executing a program with the intent of finding errors. We should expose failures to make testing process more effective. 2) Start testing early this helps in fixing enormous errors in early stages of development, reduces the rework of finding the errors in the initial stages. 3) Testing is context dependant Testing should be appropriate and different for different points of time. 4) Define Test Plan Test Plan usually describes test scope, test objectives, test strategy, test environment, deliverables of the test, risks and mitigation, schedule, levels of testing to be applied, methods techniques and tools to be used. Test plan should efficiently meet the needs of an organization and clients as well. 5) Design Effective Test case must be specified in a way that is measurable so that testing results are unambiguous. 6) Test for valid as well as invalid Conditions In addition to valid inputs, we should also test system for invalid and unexpected inputs/conditions 7) Testing must be done by different persons at different levels Different purpose addressed at different level of testing so different person should perform testing differently using different testing techniques at different level. 8) End of Testing has to be stopped somewhere. The testing can be stopped when risk is under some limit or if there limitation [4]

## 2. LITERATURE REVIEW

“Software Testing Methods and Techniques” Manuscript received May 26, 2008. I. M. Jovanovic is with the DIV Inzenjering,d.o.o.Belgrade according to this paper main testing methods and techniques are shortly described. General classification is outlined: two testing methods – black box testing and white box testing, and their frequently used techniques:

Black Box techniques Equivalent Partitioning, Boundary Value Analysis, Cause-Effect Graphing Techniques, and Comparison Testing; § White Box techniques: Basis Path Testing, Loop Testing, and Control Structure Testing. Also, the classification of the IEEE Computer Society is illustrated.[5]

“Selecting Software Testing Criterion based on Complexity Measurement” Wen C. Pai, Chun-Chia Wang and Ding-Rong Jiang Department of Information Management Kuang Wu Institute of Technology, PeiTou, Taipei 112, Taiwan, R.O.C (1999). According to Software testing is an important process in software quality assurance (SQA). Software testing consumes the majority of the software developers' time and cost of all the phases of system Development. An important problem in software testing is to determine whether a program has been tested enough with a testing criterion and when one could stop testing. On the other hand, software Measurement is used to estimate the quality of software. Complexity measures can be used to compare software quality and development cost. This paper proposes a new testing approach which choosing Testing path based on path complexity. The process is to use a set of testing paths that satisfying all-uses testing criterion, and then choose the testing path with the most complexity to test. This process is continued until the program can be accepted. The proposed formula is used to determine if it is possible to stop testing before satisfying all-du-paths testing criterion. The main contribution of the paper is that software developers can choose a proper testing action with lower testing effort.[6] “Software Testing Research and Software Engineering Education” Thomas J. Ostrand AT&T Labs - Research 180 Park Avenue Florham Park, NJ 07932 ostrand@research.att.com, Elaine J. Weyuker AT&T Labs - Research 180 Park Avenue Florham Park, NJ 07932 weyuker@research.att.com November 7–8, 2010, according to Software testing research has not kept up with modern software system designs and applications, and software engineering education falls short of providing students with the type of knowledge and training that other engineering specialties require. Testing researchers should pay more attention to areas that are currently relevant for practicing software developers, such as embedded systems, mobile devices, safety-critical systems and other modern paradigms, in order to provide usable results and techniques for practitioners.

We identify a number of skills that every software engineering student and faculty should have learned, and Also propose that education for future software engineers Should include significant exposure to real systems,

preferably Through hands-on training via internships at software producing Firms.[7]

“The Theory of Software Testing” Adtha. Lawanna Department of Information Technology, Faculty of Science and Technology Assumption University, Bangkok, Thailand (Jul. 2012)

according to Software testing is the process of testing bugs in lines of code of a program that can be performed by manual or automation testing. The theory of software testing involves problem definitions of testing such as test team, failure after testing, manual testing, uncertainty principle, participation, and incorrect test case selection. This article shows the details of a critical part of software testing, which is how to test the performance of new software and the entire system. The outcome of this article is the whole picture of three phases for software testing as follows: preliminary testing, testing and user acceptance testing.[8]

“Software Testing Techniques and Strategies” Abhijit A. Sawant<sup>1</sup>, Pranit H. Bari<sup>2</sup> and P. M. Chawan<sup>3</sup> Department of Computer Technology, VJTI, University of Mumbai, INDIA May-Jun 2012 according to This paper describes Software testing, need for software testing, Software testing goals and principles. Further it describe about different Software testing techniques and different software testing strategies. Finally it describes the difference between software testing and debugging [9]

“Software Quality Assurance and Maintenance for Outsourced Software Development” Nelly Maneva Institute of Mathematics and Informatics, BAS, 1113 Sofia, Bulgaria and American University in Bulgaria, 2700 Blagoevgrad, Bulgaria Email: nmaneva@aubg according to The paper describes the author’s experience, gained during a complex and long outsourcing project. The applied Software Quality Program is presented and some ideas for its improvement are given. The problems of project management, software configuration management and maintenance are clarified. A feasible approach to these problems –both incremental and measurable – has been suggested. In conclusion the author shares a few possible directions for further research.[10]

“Software Testing” Carnegie Mellon University 18-849b Dependable Embedded Systems Spring 1999 Authors: Jiantao Pan according to Software testing is any activity aimed at evaluating an attribute or capability of a program or system and determining that it meets its required results. [Hetzel88] Although crucial to software quality and widely deployed by programmers and testers, software testing still remains an art, due to limited understanding of the principles of software. The difficulty in software testing stems from the complexity of software: we cannot

completely test a program with moderate complexity. Testing is more than just debugging. The purpose of testing can be quality assurance, verification and validation, or reliability estimation. Testing can be used as a generic metric as well. Correctness testing and reliability testing are two major areas of testing. Software testing is a trade-off between budget, time and quality.[11]

### 3. CONCLUSIONS

This paper on Software testing describes in detail about software testing, need of software testing, Software testing goals and principles. . To perform testing effectively and efficiently, everyone involved with testing should be familiar with basic software testing goals, principles, and concepts. We further explains different Software testing techniques such as Correctness testing, Further we have discussed the basic principles of black box testing, white box testing and gray box testing. We have surveyed some of the strategies supporting these paradigms, and have discussed their pros and cons. This paper’s purpose is study of software testing techniques, so that it can provide some help to its readers.

### REFERENCES

- [1] Mohd. Ehmer Khan, “Different Forms of Software Testing Techniques for Finding Errors,” IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 3, No 1, May 2010.
- [2] Jovanovic and Irena, “Software Testing Methods and Techniques,” May 26, 2008.
- [3] Sahil Batra and Dr. Rahul Rishi, “IMPROVING QUALITY USING TESTING STRATEGIES,” Journal of Global Research in Computer Science, Volume 2, No.6, June 2011
- [4] S.M.K Quadri and Sheikh Umar Farooq, “Software Testing-Goals, Principles and Limitations,” International Journal of Computer Applications, Volume 6-No.9, September 2010
- [5] “Software Testing Methods and Techniques” Manuscript received May 26, 2008. I. M. Jovanovic is with the DIV Inzenjering, d.o.o. Belgrade
- [6] “Selecting Software Testing Criterion based on Complexity Measurement” Wen C. Pai, Chun-Chia Wang and Ding-Rong Jiang Department of Information Management Kuang Wu Institute of Technology, Pei Tou, Taipei 112, Taiwan, R.O.C .
- [7] “Software Testing Research and Software Engineering Education” Thomas J. Ostrand AT&T Labs - Research 180 Park Avenue Florham Park, NJ 07932 ostrand@research.att.com, Elaine J. Weyuker AT&T Labs - Research 180 Park Avenue Florham Park, NJ 07932 weyuker@research.att.com

[8] “The Theory of Software Testing” Adtha. Lawanna  
Department of Information Technology, Faculty of Science and  
Technology Assumption University, Bangkok, Thailand

[9] “Software Testing Techniques and Strategies” Abhijit A.  
Sawant<sup>1</sup>, Pranit H. Bari<sup>2</sup> and P. M. Chawan<sup>3</sup> Department of  
Computer Technology, VJTI, University of Mumbai, INDIA

[10] “Software Quality Assurance and Maintenance for  
Outsourced Software Development” Nelly Maneva Institute of  
Mathematics and Informatics, BAS, 1113 Sofia, Bulgaria and  
American University in Bulgaria, 2700 Blagoevgrad, Bulgaria  
Email: nmaneva@aubg

[11] “Software Testing” Carnegie Mellon University 18-849b  
Dependable Embedded Systems Spring 1999 Authors:  
Jiantao Pan

