



## Latest Software Testing Tools and Techniques: A Review

Sarika Chaudhary

Department of Computer Science and Engineering, Amity University,  
Haryana, IndiaDOI: [10.23956/ijarcse/SV7I5/0138](https://doi.org/10.23956/ijarcse/SV7I5/0138)

**Abstract**— Software Testing could be a method of finding errors while executing a program so we tend to get a zero defect software system. It is aimed at evaluating the potential or usability of a program. Software system testing is a vital suggests that of accessing quality of software system. Although plenty of advancements are wiped out formal ways and verification techniques, still we'd like software to be totally tested before it may well be handled to the client facet. So there area unit variety of testing techniques and tools created to accomplish the task. Software system testing is a vital space of analysis and plenty of development has been created during this field. During this paper, testing techniques and tools are represented. Some typical latest researches have been summarized. Software system testing is gaining additional and additional importance within the future.

**Keywords**— Software engineering, Testing Techniques, Testing Tools, Testing Principles, Verification, Validation.

### I. INTRODUCTION

Software Testing is an associate activity that's performed for evaluating software package quality and conjointly for raising it. Thus, the goal of testing is consistently and stepwise detection of various categories of errors among a minimum quantity of time provided and conjointly with a way less quantity of effort. Software package testing is additionally a vital element of software package quality assurance (SQA), and variety of software package organizations are defrayal up to four-hundredth of their resources on testing. There are four main objectives of testing i.e detection, prevention, demonstration and improving quality[1][4].

For real time software system, testing will be critical and expensive as risk analysis is additionally concerned. Risk analysis means that the likelihood by that a code project will expertise undesirable events, like delays, schedule, outright cancellation and value overruns and far additional. So, variety of test cases and test plans are created in testing which suggests that the behaviour of a program is inspected on a finite set of test cases i.e. inputs, execution preconditions, and additionally expected outcomes for a specific objective, like to follow a specific program path or to verify compliance with a particular demand, that valued inputs are created. Much, the set of test cases is taken into account to be infinite, so on paper there are plenty of test cases even for the tiniest and simplest program[3][4]. In that case, testing may take plenty of time even months and months to execute. So, the way to select a correct set of take a look at cases? Much, varied techniques are used, and a few of them are correlate with risk analysis, whereas others techniques correlate with take a look at engineering experience. The fundamental purpose of software system testing is verification, validation and error detection so as to search out varied errors and issues – and therefore the aim of finding those issues is to urge them mounted. Software system testing is over simply error detection[5]. Software system testing is completed underneath controlled conditions for:

- **Verification:** To verify if system behaves as fixed. It's the checking and testing of things, which incorporates software system, for conformity and consistency of software system by evaluating the results against pre-defined necessities. In verification we tend to raise a matter, area unit we tend to building the merchandise right?
- **Validation:** during this we tend to check the system correctness that is that the method of checking that what has been specified by user and what the user really wished [6]. In validation we tend to raise a question: are we tend to building the correct system?
- **Error Detection:** To find out unseen errors too. Variety of testings are carried out not to just find out what is going wrong but also what will be the output if software undergoes under negative testing i.e for what it is not intended for?

### II. SOFTWARE TESTING METHODOLOGIES

There are following methodologies for software testing:

#### A. White Box testing

In this testing, internal details and structure of system is created visible. Thus, it's extremely economical in detecting and resolving problems, as a result of bugs will typically be found before they cause hassle. we will so outline this methodology as testing software with the data of its internal structure and coding. White box testing is

additionally known as clear box testing, white box analysis or clear box analysis. It's a method for locating errors within which the tester has complete data of however the program parts act[1]. This methodology isn't used much for debugging in giant systems and networks, so used for net services applications. Different types of white box testing are basis path, loop testing, control structure testing etc.

### **B. Black Box testing**

A black box is any device whose internal details and workings aren't understood by or accessible to its user. This type of testing support specifications and output needs and with none data of internal structure within the program. The main aim is to check however well the system conforms to the desired needs for the system. Black box testing have very little or no data to the inner logical structure of the system. Thus, it solely examines the basic facet of the system. It ensures that every input is properly accepted and outputs are properly produced. Different types of black box testing are equivalence class partitioning, boundary value analysis and cause effect graphing[1][2].

### **C. Grey Box Testing**

In recent years, a 3rd testing technique has been conjointly thought about i.e. grey box testing. It's outlined as testing software package and conjointly having some data of its internal logic and underlying code. It uses internal information structures and algorithms for coming up with the take a look at cases over black box testing however a lot of but white box testing. This technique holds necessary once conducting integration testing between two or more modules of code written by totally different developers, wherever solely their interfaces area unit exposed for testing. This technique includes reverse engineering to work out boundary values[1]. Grey box testing is unbiased and non-intrusive as a result of it doesn't need that the tester have access to internal ASCII text file.

## **III. SOFTWARE TESTING TOOLS**

Now days we are able to get innumerable package Testing Tools within the market. Choice of tools is completely supported the project necessities & industrial (Proprietary/Commercial tools) or free tools (Open supply Tools) we are interested. Off Course, free Testing Tools might have some limitation within the options list of the merchandise, therefore it's whole supported what are we probing for & is that our demand fulfill in free version or choose paid package Testing Tools. The tools are divided into different categories as follows:

- Test Management tools
- Functional Testing Tools
- Load Testing Tools

### **A. Test Management Tools**

- TET(Test Environment Toolkit): TET produce a test driver that accommodated the current and anticipated future testing needs of the test development community. To achieve this goal, input from a wide sample of the community was used for the specification and development of TET's functionality and interfaces[7].
- TETware: The TETware is the Test Execution Management Systems which allows us to do the test administration, sequencing of test, reporting of the test result in the standard format (IEEE Std 1003.3 1991) and this tools supports both UNIX as well as 32-bit Microsoft Windows operating systems, so portability of this is with test cases you developed. The TETware tools allow testers to work on a single, standard, test harness, which helps you to deliver software projects on time.
- Test Manager: The Test Manager is an automated software testing tool is used in day to days testing activities. The Java programming language is used to develop this tool. Such Test Management tools are used to facilitate regular Software Development activities, automate & mange the testing activities.
- RTH: RTH is called as "Requirements and Testing Hub". This is an open source test management tool which we can use as requirement management tool along with this it also provides the bug tracking facilities[1][3].

### **B. Functional testing Tools**

- Ranorex: This is a simple, comprehensive and cost effective tool used for automatic testing. It is a better alternative to other testing tools because it tests applications from a user's perspective, using standard language and common programming techniques like C# and VB.net. It does not require understanding a scripting language, because it is coded in pure .net code. Any one of the three languages, VB.net, C# and Iron Python can be used. It is used by a lot of commercial software companies and enterprises around the globe. These simulation tools such can have same problems to the same record and playback methods, as the test plan and test cases are often tightly coupled to the code, and both methods still depend highly on experts to create the correct these tests to ensure full coverage[1]. Future work for ranorex involves creating an easily accessible, open and highly documented interface for the clients to write their own plug-ins, which provides the maximum of recognition for their own applications. Some of the features of this tool are:
  - The test automation modules can be created with a standard .NET compiler.
  - It provides the ability to do test automation in client's own environment.
  - It uses standard and modern programming techniques.
  - It allows testers with little programming knowledge to create professional test plans and cases and modules with Ranorex Recorder .

- It does image-based recognition .
- It contains Record-Replay functionality which is called Ranorex Recorder .
- It provides easy integration for 32 and 64 bit operating systems .
- It is built on the .NET Framework .
- Rational Function Tester: IBM developed this product in 1999. It is an object-oriented programming based automated testing tool. It includes regression and useful testing tools that author the results of recording equipment tests in an exceedingly well written format. Once captured, these scripts are dead against future script builds of any application to verify that new functionalities haven't disabled any previous practicality. With the assistance of this tool, recording equipment tests is run in addition as white box tests for code bottlenecks, memory leaks or measurement code coverage. In 2006, IBM created a significant transition to its package development platform to raised facilitate corporations build complicated package and applications[8]. The Baltic or IBM Rational 7 was developed in 2006. Some of the advantages of this tool are:
  - It enables regression testing
  - It frees up Quality Assurance departments from maintaining and executing basic tests plan and cases, and encourages the creation of additional, thorough tests
  - It automates other non-testing activities such as functional and test lab machine preparation.
  - It reduces the probability of human error that can occur during activities such as test step execution and also test result recording .

### **C. Load testing Tools**

- Load Tracer: Load tracer is one of the best web performance testing tools developed by Trace Technologies Pvt. Ltd. It is very user friendly tool used for web application load testing and performance testing. For internet application performance testing, watching and performance management additionally this tool will be used effectively. This tool uses varied techniques to make load on internet server as real user will. Then victimisation those performance watching counters and recorded logs we are able to analyse to work out the problems and bottlenecks within the internet application[7].

## **IV. CONCLUSION**

Quality is that the main focus of any software system engineering project. While not measurement, we have a tendency to can't be positive of the amount of quality during a software system. That the ways of measurement the quality area unit software system testing techniques. This paper relates various varieties of testing technique that we are able to apply in measuring numerous quality attributes. Software system testing research is that the driving part of development and application. During this era of recent and better demand of software testing, it's necessary to perpetually summarize new achievements, contemporary hotspots and propose completely different ideas so as to market the study on software system testing system engineering, to facilitate the speedy development on software testing field and trade.

## **REFERENCES**

- [1] R. Kaur Chauhan and I. Singh, *Latest Research and development on software testing techniques and tools*, International Journal of Current Engineering and Technology ,vol. 4, no. 4,2014.
- [2] S.Kaushal and J.K.Bajwa, *Analytical Review of User Perceived Testing Techniques*, International Journal of Advanced research in Computer Science and Software engineering, vol.2, issue 10,pp 213-216,2012.
- [3] Lessons Learned in Software Testing, by C. Kaner, J. Bach, and B. Pettichord
- [4] Testing Computer Software, by C. Kaner, J. Falk, and H. Nguyen
- [5] Effective Software Testing, by E. Dustin
- [6] Software testing, by Ron Patton
- [7] Software engineering, by Roger Pressman
- [8] P.Sharma, A.Wadhwa and Komal ,*Analysis of Selection Schemes for Solving an Optimization Problem In Genetic Algorithm*, International Journal of Computer Applications,vol 93,issue 11,2014.