WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

A Review on Software Testing & Techniques

Manpreet Kaur[1], Suman Lata[2], Harpreet Kaur[3]

[1,2,3] Guru Kashi University, Talwandi Sabo

ABSTRACT

Software testing is any activity aimed at evaluating an attribute of a program that it meets its required results. Testing is more than just debugging. The purpose of testing can be quality assurance, verification and validation. 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 of between budget, time and quality [1]. Software testing strives for achieving its goals but it does have certain limitations, still testing can be done for more effectively if certain established principles are followed. Software testing continuous to dominate other verification techniques like static analysis, model checking, and proofs. So, it is indispensable to understand the goals, principles and limitations of software testing so that the effectiveness of software testing could be maximized [2].

Keywords: Software testing, debugging, assurance, verification, validation, correctness, reliability, effectiveness.

Introduction

Testing is the process of executing a program with the aim of finding errors. To make our software perform well it should be error free. If testing the done successfully it will remove all the errors from the software. If finding the bug in the document software for testing software. Software testing is a process or a series of processes, designed to make sure computer code does what it was designed to do and that it does not do anything unintended. Software should be predictable and consistent, offering no surprises to users. software testing, defines testing as the process of applying a few well defined, general-purpose test criteria to a structure or model of the software. The formation of the text directly reflects the educational approach and incorporates the latest innovation in testing, including modern types of software such as object oriented, web applications, and embedded software [3].

For the beginning of the Testing system, the initial step is to create experiments. The experiments are created utilizing different testing methods, for the viable and exact testing. The significant testing methods are Black box testing, White Box testing and Gray Box testing

White Box testing

White box testing is a software testing technique in which internal structure design and coding of software are tester to verify flow of input-output and to improve design, usability and security. White box testing refers to the test methods that rely on the internal structure of the software. White box methods are based on executing or "covering" specific elements of the code [4].

Black Box Testing

Black box testing is a type of software testing in which the functionality of the software is not known. The testing is done without the internal knowledge of the product. One of the software testing techniques which I have explain in my paper is black box testing, it is a method of generating test cases that are independent of software internal structure, I have also briefly explored various different approaches to black box testing technique for finding errors [5].

Grey Box Testing

Grey box testing is a software testing technique to test a software product or application with partial knowledge of internal structure of the application. Grey box testing is a software testing method, which is a combination of both white box testing and grey box testing method. In view of this situation, this paper proposes a grey box testing method for availability simulation based on event tree model [6].

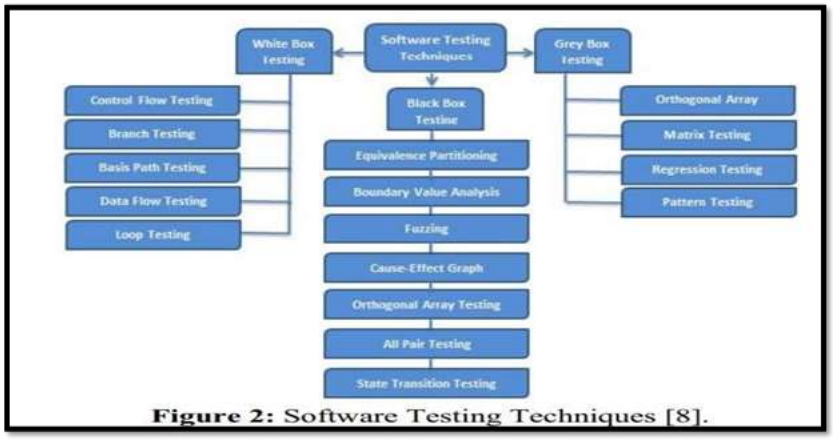

Figure 2: Software Testing Techniques [8].

Comparative Analysis of Cloud Simulation tools

Table 1: Comparison between the three testing types

Sr.no.	Black box testing	Grey box testing	White box testing
1.	The internal workings of an application are not required to be known.	Somewhat knowledge of internal working is known	Tester has full knowledge of the internal workings of the applications
2.	Also known as close box testing, data driven testing and functional testing.	Another term for grey box testing is translucent testing as the tester has limited knowledge of the insides of the application.	Also known as clear box testing, structural testing or code-based testing.
3.	Performed by end users and also by testers and developers.	Performed by end users and also by testers and developers.	Normally done by testers and developers.
4.	Testing is based on external expectations internal behavior of the application is unknown.	Testing is done on the basis of high-level database diagrams and data flow diagrams.	Internal workings are fully known and the tester can design test data accordingly.
5.	This is the least time consuming and exhaustive.	Partly time consuming and exhaustive.	The most exhaustive and time-consuming type of testing.
6.	Not suited to algorithm testing.	Not suited to algorithm testing.	Suited for algorithm testing.
7.	This can only be done by trial-and-error method.	Data domains and internal boundaries can be tested, if known	Data domains and internal boundaries can be better tested

Software testing is highly complex and time-consuming activity- it is even difficult to say when testing is complete. The combination of black box (external) and white box(internal) testing is known as grey box testing. Grey box testing is not black box testing, because the tester does know some of the internal working of the software under test [7-10].

Result and discussion

Test results are the outcome of the whole process of software testing life cycle. The results thus produced, offer an insight into the deliverables of a software project, significant in representing the status of the stakeholders. This paper presents a method for optimizing software testing identifying the most critical path. We do this by developing variable length genetic algorithms that select the software path clusters. Software testing is rarely possible because it becomes intractable or even medium size software. Typically, only parts of a program can be tested, but these parts are not necessarily the most error prone. By identifying the most critical paths, the testing efficiency can be increased [8].

Conclusion

To close our review, we might want to find that Software testing is a fundamental movement of the Programming Development Life Cycle (SDLC). We can never say that an item is "Great". Testing is an endless process. Testing shows the presence of mistakes, rather than the nonattendance. The time has come consuming and a concentrated process, along these lines, overhauled strategies and inventive techniques are important to keep up with the nature of the product. This prompt performing Automated Testing execution previously and during the testing process. This paper expects to depict exhaustively

different programming testing procedures, the requirement for programming testing, programming testing objectives, and targets. Programming testing is frequently less formal and careful than it ought to. To perform programming testing actually furthermore, proficiently, the people who are associated with testing should be acquainted with the fundamental idea, objectives and guideline of programming testing. The stage over which the product advancement and testing dwell proceeds to advance and remains really famous. Nonetheless, something so urgent and basic like Testing comes frequently very late during the time spent Software Development. There ought to be a most extreme communication between determination journalists and Testers for better getting it and early audit, which might fix uncertainty issues and thusly result in saving the expense of later fixing of the product. Analyzers subsequent to being clear about the details and necessities ought to surrender designers a specific lightweight test model, so they ensure the essential determination are met prior to taking care of the task for official testing.

Future scope

Based on the research made by Gartner, the costs of developing the IT sector will increase to 3.76 trillion dollars towards the end of 2019 and by 2020 it would have increases to 3.87, this study implies that IT plays an important role in our lives.

In the later part of the developing process, guaranteeing the quality of a product is not an easy task, even with the entire part of the techniques that determine the fate of the application. The means of developing software have undergone a vast change. According to the latest software testing trends in 2022. A future career in software testing will give you more opportunities and success.

References

1. Jiantao pan. Software testing. Dependable Embedded systems 5,2006,1999
2. SMK Quadri Sheikh Umar Farooq. Software testing- goals, principles and limitations. International journal of computer application 6(9),1,2010
3. Sanjay Kumar Singh, Amarjeet Singh. Software testing. Vandana publications, 2012
4. Thomas Ostrand. White box testing. Encyclopedia of software engineering,2002
5. Mohd khan. Different approaches to black box testing techniques for finding errors. International Journal of software engineering and applications (IJSEA) 2 (4), 2011
6. Changhao song, linhan Guo, Naichao wang, lin Ma. A grey box testing method for availability simulation software based on event tree model. 2014 Prognostics and system health management conference (PHM-2014 Hunan), 368-372, 2014
7. Shivani Acharya, vidh pandya. Bridge between black box and white boxgray box testing technique. International Journal of electronics and computer science engineering 2(1), 175-185, 2012
8. Parveen Ranjan Srivastava, Tai-hoon kim. Application of genetic algorithm in software testing. International Journal of software engineering and its application 3 (4), 87-96, 2009
9. Lata, S., and R. Kumar. “A Hybrid Approach for ECG Signal Analysis.” Proceedings - IEEE 2018 International Conference on Advances in Computing, Communication Control and Networking, ICACCCN 2018, 2018, doi:10.1109/ICACCCN.2018.8748858.
10. Lata, Suman, and Rakesh Kumar. “Disease Classification Using ECG Signals Based on R-Peak Analysis with ABC and ANN.” International Journal of Electronics, Communications, and Measurement Engineering, vol. 8, no. 2, July 2019, pp. 67–86, doi:10.4018/IJECME.2019070105.
11. Lata, Suman, and Dheerendra Singh. "A Hybrid Approach for Cloud Load Balancing." In 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), pp. 548-552. IEEE, 2022.