



INTERNATIONAL JOURNAL OF PURE AND APPLIED RESEARCH IN ENGINEERING AND TECHNOLOGY

A PATH FOR HORIZING YOUR INNOVATIVE WORK

THE ROLE OF SOFTWARE TESTING IN SOFTWARE ENGINEERING: A SYSTEMATIC SURVEY

NIRMAL KUMAR A ¹, DR. B. G. GEETHA ²

1. Assistant Professor, Department of Computer Science and Engineering, Christian College of Engineering and Technology, Oddanchatram, Dindigul, Tamilnadu - 624619, India.

2. Professor & Head, Department of Computer Science and Engineering, K. S. Rangasamy College of Technology, Thiruchengode, Namakkal, Tamilnadu - 637215, India.

Accepted Date: 13/01/2014 ; Published Date: 01/02/2014

Abstract: The systematic study of developing the software is known as software engineering. This software engineering process is used to develop the software with higher quality. Even though, the software development life cycle consists of several steps like requirements gathering, feasibility analysis, design, coding, testing, implementation and maintenance, among these steps, only the software testing process can ensure the quality of the particular software. It is not efficient that to test the software after the complete development of the software product. Hence the software should be tested while the development phases itself. Any product before coming to the real world that should be tested and evaluated. That much importance should be given to the software testing. The survey about the role and importance of the software testing has been described in this paper.

Keywords: Software Engineering, Software Testing, Test Cases

Corresponding Author: Mr. NIRMAL KUMAR A



PAPER-QR CODE

Access Online On:

www.ijpret.com

How to Cite This Article:

Nirmal Kumar A, IJPRET, 2014; Volume 2 (6): 93-97

INTRODUCTION

Testing is generally described as a group of procedures carried out to evaluate some aspect of a piece of software. The testing can be classified into two types namely manual testing and automated testing [1]. The automated testing is more efficient than manual testing. The testing process is used to detect the defects and reduce the defects. The test cases are generated in testing process. Figure 1: illustrates the various steps involved in Test Life Cycle.

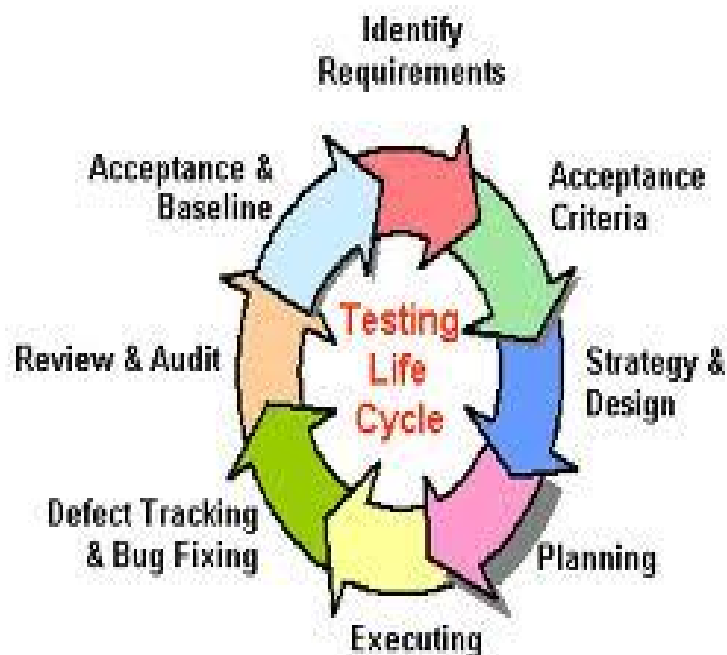


Fig 1: Test Life Cycle

1. RELATED WORKS

Sira Vegas et al proposed a new technique for classification of a set of unite testing techniques. There are many levels of testing available. The proposed classification has been proven useful for maturing knowledge about the testing techniques. The proposed the classification in three ways: providing a unifying constructs, understanding interrelationships and identifying knowledge gaps. The properties of classification were also checked [2].

Vegas S [3] suggested a technique for software testing selection. The solution for how to get a suitable set of test cases to test a software system is explained. The results of developing an artifact to assist with testing technique selection were implemented.

Engin et al [4] made a research on incremental test generation for software product lines. They presented about a novel approach for efficient test generation by combining ideas from software product lines and specification based testing. They also presented a further optimization using dedicated integer solvers for feature properties that introduce integer constraints. Figure 2 illustrates about the levels of testing.

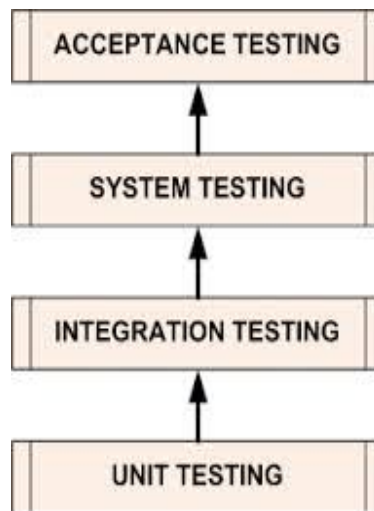


Fig 2: levels of testing

Kaushal et al [5] describes about how the system testing will be used in Test Life Cycle. The process of testing the system or software as a whole is called system testing.

Brad Long et al [6] proposed a method for selecting the testing tool for java components. Clementine Nebut et al [7] suggested a technique for automatic test generation. Sebastian E et al [8] differentiated between unit test cases and system test cases.

Ammar Masood et al [9] made a research for effective test generation for role based access control systems. Roberto et al [10] proposed a method for software reliability and testing time allocation. Paven Kumar et al [11] have given a new proposal for recomputing coverage information to assist regression testing.

2. CONCLUSION

Thus the literature survey is made to prove the role and importance of software testing in software development process. In this paper, the steps involved in testing process, testing techniques, testing levels, automated software testing tools were also discussed. The various research proposals are analyzed for the effective implementation of software testing.

3. REFERENCES

1. Manjit, K. "Comparative Study of Automated Testing Tools", International Journal of Computer Applications (0975 – 8887), Vol 24– No.1, 2011.
2. Sira Vegas, Natalia Juristo and Victor R. Basili,, "A case study on unit testing technique", IEEE Transactions on Software Engineering, Vol. 35, Issue 4, pp. 551 – 565, 2009.
3. Vegas S, "identifying the relevant information for software testing technique selection", International symposium on Empirical Software Engineering, 2004.
4. Engin U, Sarfraz K and Don B., "Incremental Test Generation for software product lines", IEEE Transactions on Software Engineering, Vol. 36, Issue 3, pp. 309 – 322, 2010.
5. Kaushal Chari and Alen Henver, "System Test Planning of Software", IEEE Transactions on Software Engineering, Vol. 32, Issue 7, pp. 503 – 509, 2006.
6. Brad Long, Daniel Hoffman and Paul Strooper, "Tools support for testing concurrent Java Components", IEEE Transactions on Software Engineering, Vol. 29, Issue 6, pp. 555– 566, 2003.
7. Clementine Nebut, Frank Fleurey, Yves Le Traon and Jean Marc, "Automatic test generation: a use case driven approach", IEEE Transactions on Software Engineering, Vol. 32, Issue 3, pp. 140– 155, 2006.
8. Sebastian E, Hui Nee, Mathew B and Mathew Jorde, "Carving and replaying differential unit test cases from system test cases", IEEE Transactions on Software Engineering, Vol. 35, Issue 1, pp. 29– 45, 2009.
9. Ammar Masood, Rafae Bhatti, Arif Ghafoor and Adithya Mathur, "Scalable and effective test generation for role based access control systems", IEEE Transactions on Software Engineering, Vol. 35, Issue 5, pp. 654– 668, 2009.

10. Roberto P, Stefano R and Kishore S Trivedi, "Software reliability and testing time allocation: An architecture based approach", IEEE Transactions on Software Engineering, Vol. 36, Issue 3, pp. 323– 337, 2010.

11. Pavan Kumar Chittimalli and Mary Jean Harrold, "Recomputing coverage information to assist regression testing ", IEEE Transactions on Software Engineering, Vol. 35, Issue 4, pp. 452– 469, 2009.