Study of Software Testing Techniques and Tools
Department of School of Information Technology &Engineering
VIT University, Vellore, Tamil Nadu, India

Abstract:
Software testing is a method with the aim of finding software bug so we tend to get a zero defect software. It is aimed at figuring out the worth usability of a program. Software testing is an important means of getting to quality of software. Though a lot of advancements have been done in formal methods and verification techniques, still we need software to be fully tested before it could be handled to the customer side. This way there are some verification techniques and tools made to complete the task. Software testing is an important area of research and a lot of development has been made in this field. In this paper, focuses on testing techniques and tools have been described. Some typical latest researches have been summarized. Software testing is gaining more and more importance in the future.

Keywords: Software testing, Software testing strategies, Testing tools, research orientation

I. INTRODUCTION

Software Testing is an activity that is performed for evaluating software quality and also for improving it. So the goal of testing in an organized way and stepwise detection of different classes of errors within a minimum amount of time and also with a much less amount of effort. Software testing is also an important part of SQA (Software Quality Assurance) and some organizations are spending up to 40% of their resources on testing. Effectiveness and Efficiency plays a major role in software testing. Test efficiency is the relative cost of finding a bug in the software under test [1]. In the detection process, different errors, defects, and deficiencies are detected. System abilities and various limitations, quality of all components, the work products, and the overall system are calculated. In the prevention process which includes prevent or reduce the number of errors, to clarify system requirement specifications and system performance is given. Different ways to avoid risks and to tackle problems in the future are identified. Verification is to verify if system behaves as specified. Verification is done at the starting of the development process. It includes reviews and meetings, walkthroughs, inspection, etc. to evaluate documents, plans, code, requirements and specifications[2]. To verify if system behaves as specified. Validation is concerning crucial if the system complies with the necessities and performs functions for which it is intended for and meets the organization’s goals and user needs[2]. In this we check the system correctness which is the process of checking that what has been specified by user and what the user actually wanted. Software testing is more than just error detection.

Software Testing Strategies
A software testing strategy integrates various software test case design methods into a well-planned series of steps. This will result in successful testing of software. Testing and debugging are different activities. Debugging must be accommodated in any testing strategy. There are four software testing strategies as shown in figure.1

Figure 1. Four Testing Strategies [10]
There are four testing strategies. They are Unit testing, Integration testing, System testing and acceptance testing. Unit testing is a method by which individual units of source code are tested together with associated control data. Unit test run by software developers to ensure that code [3]. It is done at the lowest level. Basic unit of software is tested by unit testing, which can be a component. In unit testing, Interfaces tested for proper information flow, Local data are examined to ensure that integrity is maintained, Boundary conditions are tested. In Integration testing individual software modules are combined and tested as a group to verify they work together without errors. It is done by a specific integration tester or a test team [5]. Integration testing with the inclusion of Bottom - up testing (test harness), Top - down testing (stubs), Modified top - down testing – test levels independently, Big Bang and Sandwich testing. In System testing which investigates both functional and non-functional requirements of the software. In system testing there are four testing. They are Recovery testing, security testing, stress testing and performance testing. Recovery testing which checks system’s ability to recover from failures. Security testing verifies that system protection mechanism prevents
improper penetration or data alteration. Stress testing program is checked to see how well it deals with abnormal resource demands. Performance testing which tests the run-time performance of software. In the acceptance testing, Once all or most of the defects have been corrected after the system test, the system will be delivered to the user or customer for acceptance testing. It is done when the complete system is handed over to the customers or users from developer side. Making sure the computer code works properly for supposed user in his or her traditional work surroundings [2].In Alpha test where software is tested by customer at the developer’s site. In Beta test version of the complete software is tested by customer at his or her own site without the developer being present.

Software Testing Methodologies

Programming testing procedures are the diverse methodologies and methods for guaranteeing that a product application specifically is completely tried. Programming testing procedures incorporate everything from unit testing singular modules, combination testing a whole framework to specific types of testing, for example, security and execution. As programming applications get always mind boggling and entwined and with the substantial number of various stages and gadgets that need to get tried, it is more imperative than any time in recent memory to have a powerful testing strategy for ensuring that product items/frameworks being created have been completely tried to ensure they meet their predetermined prerequisites and can effectively work in all the foreseen situations with the required ease of use and security [12]. There are three testing types as shown in figure.2

![Software Testing Methodologies][3]

Figure.2Software Testing Methodologies[10]

White Box Testing, Clear Box Testing otherwise called White Box Testing, Open Box Testing, Glass Box Testing Transparent Box Testing, Code Based Testing, Structural Testing which is a product testing strategy where the inward structure of the thing will be known to the analyzer. The analyzer picks contributions to exercise ways with the code and decides the fitting yields Programming know and the structure information is fundamental Clear box testing will be trying past the UI and low down the framework. This strategy is named so on the grounds that the product program according to the analyzer resembles a white/straightforward box inside which one unmistakably observes White Box Testing technique is relevant to the accompanying levels of programming testing. The first level is Unit Testing. For testing ways inside a unit, second level is Integration Testing. For testing ways between units and final is System Testing For testing ways between subsystems. Example, In any case it is principal connected to the Unit Testing Illustration. An analyzer as a rule a designer too concentrates in the usage of the coding and a specific area in a website page which decides all lawful legitimate and which is not valid illicit information sources and confirms the yields against the normal results which is likewise dictated by concentrate the execution [9]. Black Box Testing, Discovery Testing Discovery Testing otherwise called Behavioral Testing which is a product testing strategy the inside structure/outline/execution of the thing being tried is not known to the analyzer. These testing are utilitarian or non-practical however generally useful as shown in figure.3. This strategy is on the grounds that the product program according to the analyzer which cannot be seen [9].

![Black Box Testing][4]

Figure.3. Black Box Testing[10]

This strategy endeavors to discover blunders in the accompanying classifications are not properly capacities, Interface blunders, Errors in information structures or outside database get to, Behavior or execution blunders and Initialization and end blunders Case Example, An analyzer without information of the internal body in which the website pages will test the site by utilizing a program giving data sources snaps keystrokes and checking the yields against the normal result. Discovery testing technique is relevant to the accompanying levels of programming testing [2]. There are three levels such are Reconciliation Testing, Framework Testing, and Acknowledgment Testing. Grey Box Testing is also known as Dim Box Testing which is a product testing technique that blends Black box Testing strategy and white box testing technique. Inner structure of the black box testing being tried which is obscure to the analyzer. Interior structure of white box testing will be known in gray box testing. This includes having admittance to interior information structures and calculations for reasons for outlining the experiments however testing in presence of client. Gray Box Testing strategy might be utilized as a part of different stages of testing which will be useful in Integration Testing [12].

II. PROGRAMMING TESTING TOOLS

There are various apparatuses accessible in vend for programming testing a few have been utilized from quite a while and little new devices have additionally been established with a great deal of new features and process Here we will talk about few apparatuses that are utilized for computerized testing [7]. A correlation of robotized programming testing devices Ranorex tool is a straightforward far reaching and practical
instrument utilized for programmed examining. It is a superior contrasting option to other examining instruments since it tests functions from a client's point of view utilizing standard dialect and regular programming procedures such C# programming and (visual basic)VB.net[10]. It doesn't need grasping a scripting dialect. Since it is arranged in unadulterated .NET code. In general of the three dialects Visual Basic.net, C# and Iron Python may be utilized. It is utilized by a ton of business programming organizations and undertakings around the world. These recreation instruments such may have look alike issues to a similar log and playback techniques, in the act of test plan and experiments(test cases) are frequently firmly united to the code and the two strategies still depend exceptionally on specialists to make the right these tests to guarantee full scope. Future job for Ranorex includes making an effectively available open and very archived collaboration for the customers to compose their own particular modules which gives the greatest of acknowledgment to their own function. A portion of the components of this apparatus are tool which will testrobotization programs can be made with a ideal .NET compiler. It gives the capacity to execute test robotization in customer's own particular condition. It utilizes standard and present day programming strategies. It permits analyzers with small programming learning to make proficient test arranges and cases and factors with Ranorex Recorder. It images based acknowledgment and contains log Replay usefulness which is known as Ranorex Recorder. It gives simple reconciliation to 32bitand 64 bit working frameworks. It is based on the .NET scheme. It offers a classic and adaptable test robotization interface. Ranorex Recorder gives client code activities which permits engineers to give unique approval or robotization strategies for their analyzers with less involvement in computing. It focuses to get all things adaptable and mechanized. It underpins every one of the advancements by means of Ranorex Plug Ins. It permits UI for overseeing experiments arrangements and setups. It bolsters the utilization of information factors Sane Functional monitor. Rational Functional Tester (RFT) tool where IBM built up in 1999. RFT is a question arranged programming based computerized testing apparatus. It incorporates relapse and utilitarian testing devices which make up the consequences of discovery tests in an all-around scripted configuration Once caught these setup can be executed against future setup works of any function to confirm new performance have not handicapped any past usefulness. [10].With the assistance of this device discovery verifications can be keep running and additionally white box tests as code obstruction memory breaks or experimenting code scope. In 2006, IBM formed a noteworthy move to its product advancement stage to better utility organizations assemble complex programming and applications. The Baltic or IBM Rational 7 was created in 2006 A portion of the upsides of this device are which empowers relapse testing. It arranges for Quality Assurance offices from keeping up and executing essential tests plan and test cases and supports the production of extra intensive tests. It computerizes other non-experimenting exercises for example useful and test lab machine readiness. It decreases the likelihood of human mistake that can happen amid exercises for example analyze step execution and furthermore test result recording. It perform with Web based, Java and Microsoft Visual Studio,.NET, SAP, terminal based, Siebel and Web 2 0 applications. This item additionally utilizes an Object Code Insertion (OCI) innovation where no programming code is utilized. This innovation takes a gander at the runnable records in a function. These instruments when incorporated with the product including clear Coverage and Purify Quantify work white box testing on an outsider code A portion of the benefits of these devices are which gives memory spill discovery and run time mistake. It logs the correct measure of duration an application spends in a given square of code with the end goal of discovering all wasteful code blocks. It identifies zones of use that have been and have not been executed. When performing relapse tests on an item if the function changes similar to pictures in various areas tests won't fall flat in light of the fact that the item utilizes hearty Janova apparatus is much like others for it empowers a few clients to robotize programming testing arrangements and by the assistance of this instrument it is complete in a cloud as well. This device does not depend upon either scripts to be composed (i.e) just basic English based apparatuses are utilized that rearrange the undertaking of programming execution with effective and simple to utilize instruments. There is no such programming to compute and in this manner no infrastructural speculation is required. Since it is utilized as a part of the cloud it obtain a snappy and simple setup that incorporates no introduce. This cloud placed programming has a simple route to landing page[10]. Latest Research And

III. DEVELOPMENT IN SOFTWARE TECHNIQUES AND TOOLS:

The main aim of this research is to bring out the latest development on software testing tools and techniques. They have issues too. Latest advancement in software testing new issues on testing in latest years they are test driven development, iterative and incremental testing, GUI automation test, testability of component software and international and domestic research hotspots. These areas the testing process should be improved [8]. Improvement Latest Results of Software Testing are test driven development before writing the code we should write the related test cases, test code and the cycle continues until the development has been completed. but latest popular XP [7]. Iterative And Incremental Testing if iteration is done the system will integrate some new functions until the entire system is completed. Mainly focuses on cumulative functions used in regression test and iterative is completed in two parts and this is one of IBM most widely used methods, GUI Automation Test, in this framework is based on OO capture for GUI. The automatic generation of test case based on antonym algorithm using bit code the algorithm is established.it is the model driven software code in MDA improved automation degree in software testing. Testability of Component Simulation this the test development MVC based software is designed not only ensured the successful development of ESTDE but also for improved the adaptability and repeatability of the system. International And Domestic Research hotspots from some of the researches different types of software features like embedded system and real time system are found. These research that direct to other new software development including researches that aimed on OO technology java language and software component. The automated testing helps to improve the degree of automation in all steps of testing and this ease the burden in test analysis such as automatic generation of test case, performance of regression tests and improved the interoperability and effectiveness of testing tools[9].

http://ijesc.org/
IV. CONCLUSION:

The main factor is quality of software engineering project without measuring we cannot be sure of the level of quality software. so these are done by software testing methods. This paper related to various kinds of testing technique that apply measuring various qualities attributes. Software testing is a driven element of development and application. in this era of new and higher demand of software testing it is most important to constantly summarize new achievements, fresh hotspots and propose of different ideas in order to improve the study of software testing system engineering to facilitate the rapid development on software testing field and industry.

V. REFERENCES


[2]. Classification of software testing tools based on the software testing methods, Mustafa, Khaleed M., Rafa E. Al-Qutaish, and Mohammad I. Muhairat, 2009.


[6]. "Black-box system testing of real-time embedded systems using random and search-based testing.

[7]. Nancy Bordelon, A comparison of automated software testing tools.


[10]. RasneetKaurChauhan and Iqbal Singh, Latest research on software testing techniques and tools, Aug 2014.


