

A Brief Survey of Software Testing Methods and Tools

Dr. Kusum Lata Bharti ^[1], Dr. Tanvir Ahmad Abbasi ^[2], Dr. Varun Tiwari ^[3]

Associate Professor ^[1], Comm-IT Career Academy, (Affiliated to GGSIP University)

Professor ^[2], Comm-IT Career Academy, (Affiliated to GGSIP University)

Associate Professor ^[3], Comm-IT Career Academy, (Affiliated to GGSIP University)

ABSTRACT

Software Testing is a method of implement a program or application with the aim of finding the software bugs or software testing is the mixture of software verification and software validation. It is projected on estimate the capability or utilizable of a program. In spite of the fact that a lot of enhancement have been done in prescribed methods and verification method, still we require software to be entirely tested prior to be handle to the client side. Hence there are a number of testing methods and tools made to achieve the job. Software testing is a vital part of explore and a lot of improvement has been made in this field. In this paper, we have tried testing techniques and tools have been described. A few typical most recent researches have been review.

Keywords:- Risk Analysis, Test Case, Testing Strategies, Testing Techniques

I. INTRODUCTION

Software Testing is an activity that is execute for assess software class or quality as well for advancing it. Therefore, the objective of testing is thoroughly and stepwise finding of different classes of inaccuracy within a smallest amount of time as well with a large amount of effort. Software testing is also with a much less amount of effort. Software testing is also an essential component of software quality assurance. Testing can be much costly as well risk analysis also concerned. Risk analysis is the process of classify risks in application and prioritizing them to test. Risk Analysis effort to recognize all the risks and then measure the severity of the risks. A number of test cases and test plans are made in testing which means that the performance of a program is examined on a predetermined set of test cases. Basically, the set of test cases is considered to be endless, therefore there are a lot of test cases even for the smallest and simplest program. In that case, testing might take a lot of time to execute. Accordingly, how to decide an appropriate set of test cases? Practically, various techniques are used, and several of them are associated with risk analysis, whereas others are linked with test engineering expertise. The purpose of software testing is verification, validation and error detection in order to find various errors and problems.

There are different levels of testing:

- ✓ **Debug:** The successful rectification of a failure.
- ✓ **Validate:** The method of finding as many faults in requirements, design and application under test (AUT).
- ✓ **Demonstrate:** The process of viewing that most important features work with input.
- ✓ **Verify:** The method of finding several faults in the application under test (AUT) as possible.
- ✓ **Prevent:** To keep away from errors in growth of requirements, design and implementation by self-checking techniques.

Software Testing Terminology concerns with the following:

- **ault or Defect:** A missing or inaccurate declaration in a program resulting from an error is a fault.
- **rror or Bugs:** While coding, when people build error we call these mistakes bugs.
- **ailure:** Failures arise when a fault executes.
- **est Case:** A test case has individuality and is related with program performance. A test

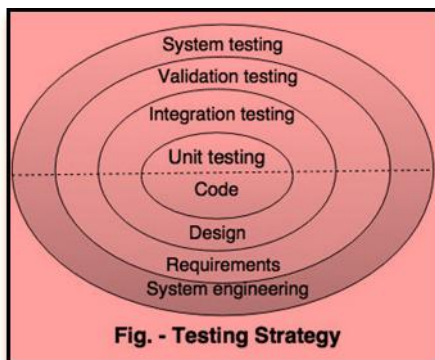
case has a set of inputs and a list of estimated outputs.

- **Test:** Testing is related with errors, faults, failures and incident. A test has two different objectives to hit upon failures or to express correct execution.
- **Incident:** When a failure takes place, it may or may not be readily obvious to the user. An incident is the indication linked with a failure that attentive the user to the happening of a failure.

II. STRATEGIES OF SOFTWARE TESTING

A test approach is summarizing that demonstrate the testing approach of the software development cycle. It is formed to let know project managers, testers, and developers about some key issues of the testing process.

Unit Testing: Unit testing is the smallest testable part is isolated from the software or



application code and tested to resolve whether it works properly.

Integration Testing: Integration testing is a method where the all component is combined and tested as a cluster.

Validation Testing: Validation testing to review the software at the ending of the development process to check whether software or application meets the client requirements.

System testing: System testing to check the performance of a entire and wholly integrated

software product. The major focus of this trying to assess end-user requirements.

Software Testing Techniques:

Black Box Testing (or Functional Testing): Black box testing is a software testing method in which functionality of the software or application is tested without knowledge of internal code structure and implementation details. This type of testing is based completely on the software requirements and specifications.

Black Box Testing Strategies:

- ✓ Boundary Value Analysis
- ✓ Equivalence Class Testing
- ✓ Decision Table Testing
- ✓ Cause Effect Graph

Gray Box Testing: It is a technique to test the application with having restricted information of the internal workings of an application. The area of a system always gives the tester a boundary over someone with limited domain knowledge.

Gray Box Testing Strategies:

- ✓ Regression Testing
- ✓ Pattern Testing
- ✓ Matrix Testing

White Box Testing /Glass testing (or Structural Testing): White-box testing is the complete analysis of internal logic and structure of the code. To execute white-box testing on an application, a tester needs to familiar with the internal workings of the code.

White Box Testing Strategies:

- ✓ Path Testing
- ✓ Control structure Testing
- ✓ Loop Testing

Software Testing Principles

Exhaustive testing is not possible: Exhaustive testing is not possible. In its position, we require the finest amount of testing based on the risk assessment of the application.

Defect Clustering: It states that a little number of components contains most of the flaws identify. If the same tests are repeated over and over again, in the long run the same test cases will no longer find new bugs.

Pesticide Paradox

If the same set of recurring tests is performed, the process will be ineffective for find out new defects.

Testing shows presence of defects

Software Testing shrink the possibility of undiscovered defects left over in the software but if no defects are found, it is not a verification of accuracy. If, you effort extra rigid, taking all safety & build your software product 99% bug-free, and the software does not meet the requirements & requirements of the customers.

Absence of Error

It is feasible that software which is 99% bug-free is still impractical. Software testing is not simple finding defects, as well to make sure that software deal with the business needs.

Finding and setting up defects does not help if the system construct is impractical and does not accomplish the user's requirements.

Early Testing

Testing should begin as early as possible in the Software Development Life Cycle. If any defects in the requirements or design phase are captured in early phase. It is much cheaper to fix a defect in early stages of testing. It is suggested that you begin finding the bug the instant the needs are defined.

Testing is context dependent

Testing is situation which means that the approach you analysis commercial site will be dissimilar from the way you test a business off the shelf application. The complete developed software's are not the same. You might employ a different approach, methodologies, techniques and types of testing depending upon the application type.

Software Testing Tools

There is a numeral tools offered in market for software testing. Several have been used as of a long time and some latest tools have also been developed with a lot of latest functionalities. Now, we are discussing few tools that are used for automated testing.

Selenium: It is a tool which automates browsers. It mostly used for automating Web Based applications. It is Freeware tool or open source tool.

Sahi: It is a testing automation tool to automate web applications testing. It is a freeware tool and its source is written in Java and JavaScript programming languages.

Watir (aka water): It is an open source testing tool made up of Ruby libraries to automate web application testing.

III. CONCLUSION

Excellence is the major center of any software engineering project. Without assessing, we cannot be persuaded of the level of quality in software. Thus the technique of measuring the superiority is software testing techniques. In this paper communicate different types of testing method to facilitate in measuring various quality aspects. Software testing study is the driving factor of development and application. In this period fresh and privileged demand of software testing, it is important to persistently review new achievements, and recommend different thoughts encourage the study on software testing thoughts in order to encourage the study on software testing system engineering, to ease the quick development on software testing field.

REFERENCES

- [1] <http://inpressco.com/wp-content/uploads/2014/07/Paper122368-2372.pdf>
- [2] https://www.google.co.in/search?q=software+testing+strategies&safe=active&source=Inms&tbm=isch&sa=X&ved=0ahUKEwj57nd-8_aAhXHbbwKHa0yDu8Q_AUICygC&bi

w=1137&bih=735#imgrc=Ca-
unt7gYoYaVM

- [3] <https://medium.com/@briananderson2209/best-automation-testing-tools-for-2018-top-10-reviews-8a4a19f664d2><https://medium.com/@briananderson2209/best-automation-testing-tools-for-2018-top-10-reviews-8a4a19f664d2>
- [4] Zhang Hongchun, Research on New Techniques and Development Trend of Software Testing. Myers, Glenford J.(1979), IBM Systems Research Institute, Lecturer in Computer Science, Polytechnic Institute of New York, The Art of Software Testing, by John Wiley & Sons, Inc.
- [5] Fu Bo (2007), Automatic Generation Method of Test Data Based on Ant Colony Algorithm, Computer Engineering and Applications.43 (12).
- [6] Nancy Bordelon, A comparison of automated software testing tools R.S. Pressman & Associates, Inc. (2005). Software Engineering: A Practitioner's Approach, 6/e; Chapter 14: Software Testing Techniques IEEE(1990), IEEE Standard Glossary of Software Engineering Terminology , Los Alamitos, CA: IEEE Computer Society Press.
- [7] Redmill, Felix (2005), Theory and Practice of Risk-based Testing, Vol. 15, No. 1.
- [8] Stacey, D. A.(2004), Software Testing Techniques Guide to the Software Engineering Body of Knowledge, Swebok – A project of the IEEE Computer Society Professional Practices Committee.