

A Brief Overview of Software Testing Techniques and Metrics

Anitha.A

Programmer, School of Computer Studies (PG), RVS college of Arts & science, Coimbatore, India.

Abstract: Software Testing is the process of executing a program or system with the intent of finding errors. Software testing is any activity aimed at evaluating an attribute or capability of a program or system and determining that it meets its required results. Testing is an integral part in software development. It is broadly deployed in every phase in the software development cycle. Typically, more than 50% percent of the development time is spent in testing. Metrics are gaining importance and acceptance in corporate sectors as organizations grow, mature and strive to improve enterprise qualities .In this paper, discuss software testing techniques and testing metrics.

Keywords: software testing, STLC, techniques, metrics

I. INTRODUCTION

A primary purpose of testing is to detect software failures so that defects may be discovered and corrected. Testing can be used as a generic metric as well. Software testing is also used to test the software for other software quality factors like reliability, usability, integrity, security, capability, efficiency, portability, maintainability, compatibility etc. Bugs can cause huge losses.[8][7]

A. Goals for Software Testing

Goals are the output of the software process. Software testing has following goals [6]

- Verification and validation
- Priority Coverage Balanced
- Traceable
- Deterministic

B. Software Testing Life cycle

Software Testing Life Cycle (STLC) is the testing process which is executed in systematic and planned manner. In STLC process, different activities are carried out to improve the quality of the product [4].The STLC acts as a guide to its followers and management so that the progress is measurable in the form of achieving specific milestones.

II. SOFTWARE TESTING TECHNIQUES

In this Section the focus is mainly on the different software testing Techniques. Software Testing Techniques can be divided into two types:- [6]

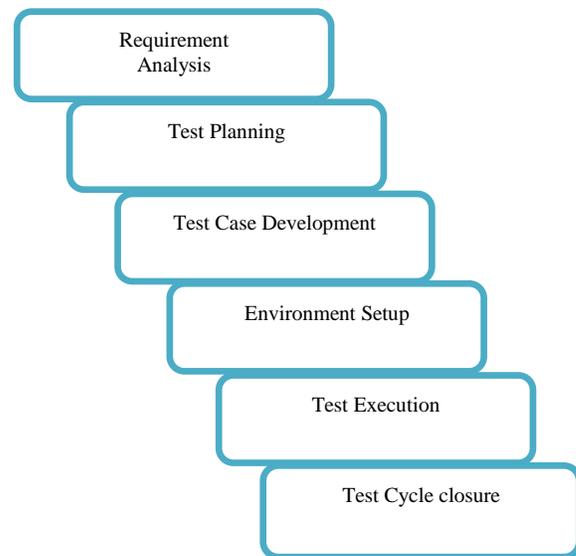


Fig 1 : Software Testing Life Cycle

A. Manual testing (static testing)

Manual testing is done in early phase. This type includes the testing of the Software manually i.e. without using any automated tool or any script. Manual testing also includes exploratory testing as testers explore the software to identify errors in it. Sometimes it's called static testing. Manual testing as follows:- [6]

- Walk through
- Informal Review
- Technical Review



- Inspection

B. Automated Testing (Dynamic testing)

When the tester writes scripts and uses software to test the software. This testing is used to re-run the test scenarios that were performed manually, quickly and repeatedly. It's sometimes called dynamic testing.

Different automatic testing as follows:-

- Correctness testing
- Performance testing
- Reliability testing
- Security testing

1)Correctness Testing:

Correctness testing tells the right behaviour of system from the wrong one for which it will need some type of Oracle. Correctness testing has divided into three forms as follows:-

- White box testing
- Black box testing
- Grey box testing

- *White box Testing*

White box testing is the detailed investigation of internal logic and structure of the code. White box testing is sometimes called *glass testing or open box testing*. In order to perform white box testing on an application, the testing team needs to possess knowledge of the internal working of the code. The tester needs to have a look inside the source code and find out which unit/chunk of the code is behaving inappropriately.

- *Black box Testing*

The technique of testing without having any knowledge of the interior workings of the application is Black Box testing. The testing team is oblivious to the system architecture and does not have access to the source code. Typically, when performing a black box test, a tester will interact with the system's user interface by providing inputs and examining outputs without knowing how and where the inputs are worked upon .This is not suited for algorithm.

- *Gray box Testing*

Grey Box testing is a technique to test the application with limited knowledge of the internal workings of an application. In software testing, the term the more you know the better carries a lot of weight when testing an application [9]

TABLE I
 BLACK BOX vs. WHITE BOX vs. GRAY BOX

S.N.	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 the internal workings are known	Tester has full knowledge of the Internal workings of the application
2	Also known as closed 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 behaviour 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

2)Performance testing:

It means how best something performs under a given benchmark .It is mostly used to identify any bottlenecks or performance issues rather than finding the bugs in software. There are different causes which contribute in lowering the performance of software: This testing divided into two types:-[6][9]

- Load testing
- stress testing

- *Load Testing*

A process of testing the behaviour of the software is by applying maximum load in terms of software



accessing and manipulating the large input data. It is also performance testing but under various loads. It can be done at both normal and peak load conditions. This type of testing identifies the maximum capacity of Software and its behavior at peak time.

• *Stress Testing*

It is performance under stress conditions like Shutdown or restart of Network ports randomly.

3) *Reliability Testing:*

The purpose of reliability testing is to discover potential problems with the design as early as possible and, ultimately, provide confidence that the system meets its reliability requirements.

4) *Security Testing:*

Security testing involves the testing of Software in order to identify any flaws and gaps from security and vulnerability point of view. Following are the main aspects which Security testing should ensure: [8]

- Confidentiality.
- Integrity.
- Authentication.
- Availability.
- Authorization.
- Non-repudiation
- Input checking and validation.
- SQL insertion attacks.

III. SOFTWARE TESTING METRICS

Software metrics are applicable to the whole development life cycle from initiation, when cost must be estimated to monitoring the reliability of the end product in the field, and that product changes over time with enhancement.

The essential step in establishing test metrics is to identify the key software testing processes that can be objectively measured.

A. *Needs of Test Metrics*

A major percentage of software projects suffer from quality problems, which in turn requires new testing metrics to measure test processes effectively. Test metrics are key “facts” that project managers can use: [8]

- To understand their current position
- To prioritize their activities to reduce the risk of schedule over-runs on software releases.

B. *Types of Testing Metrics*

Testing metrics divide into three types as follows:[2]

- Manual Testing Metrics
- Performance Testing Metrics

• **Automation Testing Metrics**

Following table shows different software testing metrics.

TABLE II
 SOFTWARE TESTING METRICS [2]

Manual	<ul style="list-style-type: none"> • Test Case Productivity • Test Execution Summary • Defect Acceptance • Defect Rejection • Bad Fix Defect • Test Execution Productivity • Test Efficiency • Defect Severity Index
Performance	<ul style="list-style-type: none"> • Performance Scripting Productivity • Performance Execution Summary • Performance Execution Data - Client Side • Performance Execution Data - Server Side • Performance Test Efficiency • Performance Severity Index
Automation	<ul style="list-style-type: none"> • Automation Scripting Productivity • Automation Test Execution Productivity • Automation Coverage • Cost Compression
Common Metrics	<ul style="list-style-type: none"> • Effort variance • Schedule Variance • Scope change

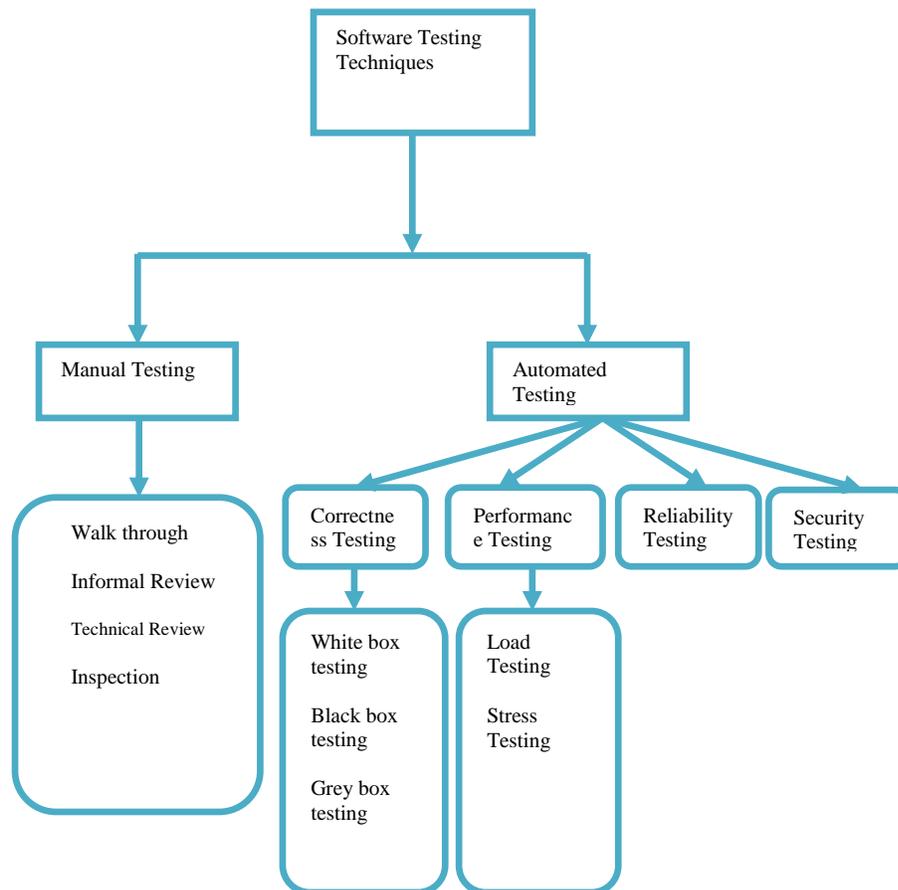


Fig 2: Software testing techniques

IV. CONCLUSION

Software testing can be very costly. Automation is a good way to cut down time and cost. Software testing tools and techniques usually suffer from a lack of generic applicability and scalability. The reason is straight-forward. Software testing is an important technique for the improvement and measurement of a software system quality.

REFERENCES

- [1] Chayanika Sharma, Sangeeta Sabharwal, Ritu Sibal, "A Survey on Software Testing Techniques using Genetic Algorithm", IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 1, No 1, January 2013, ISSN (Print): 1694-0784 | ISSN (Online): 1694-0814.
- [2] Mr. Premal B. Nirpal, Dr. K. V. Kale, "A Brief Overview Of Software Testing Metrics", International Journal on Computer Science and Engineering (IJCSE), ISSN: 0975-3397 Vol. 3 No. 1 Jan 2011.
- [3] Nancy Bordelon, "A Comparison of Automated Software Testing Tools", Degree of Master Science Thesis, 2012.
- [4] Wasif Afzal, "Metrics in Software Test Planning and Test Design Processes", Master Thesis Software Engineering Thesis no: MSE-2007:02, January 2007.
- [5] Sukhjinder Singh, Er.Sukhpreet Singh, Dr.Madhuchanda Rakshit, "A Review of Various Software Testing Techniques", IJREAT International Journal of Research in Engineering & Advanced Technology, Volume 1, Issue 4, Aug-Sept, 2013, ISSN: 2320 – 8791.
- [6] Abhijit A. Sawant, Pranit H. Bari and P. M. Chawan, "Software Testing Techniques and Strategies", International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 ,Vol. 2, Issue 3, May-Jun 2012, pp.980-986.
- [7] Lu Luo, "Software Testing Techniques", School of Computer Science, Carnegie Mellon University.
- [8] Dr. Arvinder Kaur¹, Mrs. Bharti Suri², Ms. Abhilasha Sharma, "Software Testing Product Metrics - A Survey", Proceedings of National Conference on Challenges & Opportunities in Information Technology (COIT-2007), RIMT-IET, Mandi Gobindgarh. March 23, 2007.
- [9] <http://www.tutorialspoint.com>



BIOGRAPHY



A. Anitha received her B.Sc., in 2009, MSc, in 2011, M.Phil. in 2012 from Bharathiyar University Coimbatore. At present she is working as programmer in RVS College of Arts and Science College, Coimbatore. Her area of research includes Software Engineering.