

Analyzing Software Performance and Reliability Estimation in Software Testing

Alireza Souri¹, Monire Norouzi²

^{1,2} Department of Computer Engineering, University College of Nabi Akram, Tabriz, Iran
a.souri@ucna.ac.ir, monire_norouzi@yahoo.com

Abstract: In this paper we analyze reliability estimation and software performance in software testing. One of the important verification and validation techniques is statistical test which is one of the dynamic techniques. We use statistical test in order to improve software performance. By this work we put sets of data in one case study and by presenting a model-based life cycle we will enhance reliability estimation in the case study. By using our model-based life cycle we can improve reliability of systems which is created by some subsystems.

[Alireza Souri, Monire Norouzi. **Analyzing Software Performance and Reliability Estimation in Software Testing**. Journal of American Science 2012;8(3):308-310]. (ISSN: 1545-1003). <http://www.americanscience.org>. 41

Keywords: Software testing, statistical test, reliability, performance, evaluation

1. Introduction

Good software is built when every step is taken with full investigation that 'A right product is developed in a right manner'. Software Verification & Validation is one model which helps the software system designers and test or verifier engineers to justify that a right product is build right way throughout the development process and improve the quality of the software product.

Many Methods and techniques are used in the software Verification and Validation which designed completely, the planning of which starts right from the beginning of the development process. The Verification section comes before Validation section, which concretes Software detection, surveys, purchases, scrutinizes etc. in each part of verification every part of Verification is a phase of the Testing Life Cycle.

In this paper, first we describe Verification and Validation and we explain software testing briefly. So, by using the statistical test, we test a case study.

Next, we present a model-based lifecycle that by using this model; we can enhance reliability estimation and performance of software.

2. Verification and Validation

We describe verification and validation definitions briefly:

Verification and validation (V&V): the checking and analyzing processes that ensure the software satisfies its specification and meets the needs of the clients who are paying for it.

Verification: involves checking the software adapts by its specification while the more general process of *validation* ensures the software meets the needs of the clients.

V&V is a whole life-cycle process, encompassing requirements reviews, design reviews,

code inspections, and program testing. V&V techniques are included Software inspection and Software testing. In software inspections many different defects can be discovered in a single review of the source code or software model.

In software testing, one defect may mask another hence several executions are required. Software inspections reuse domain and programming knowledge so reviewers are likely to have seen the types of error that commonly occur. Software inspections and software testing are complementary, not competing techniques.

3. Software Testing

Software Reliability is one of the important factors in systems engineering and analysis today[.Software testing life cycle identifies what test activities to carry out. Even though testing differs between organizations, there is a testing life cycle.

Software Testing Life Cycle has 7 parts:

- Test Planning
- Test Analysis
- Test Design
- Construction and verification
- Testing Cycles
- Final Testing and Implementation and
- Post Implementation.

In this paper, we explain types of software testing techniques briefly as follow:

Defect Test:

For finding inconsistence and defect between a program and it specification has used. By testing models, we can discover program faults and program defects. A successful defect test is test that reveals the presence of defects in the system.

Statistical Test:

Statistical Test has been designed for software performance and reliability estimation. By running tests that reflect actual user inputs and their frequency, an estimate of operational reliability can be made. In statistical test sets of data are put in system with an algorithm.

4. Software performance and Estimation

According to the definitions which we have in the last sections, we present a model-based of life cycle for a software system.

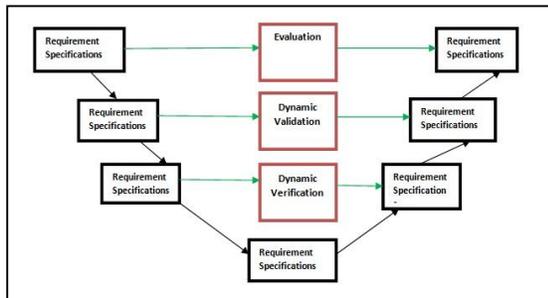


Figure1: model-based of life cycle

We analyze 3 important parts of this design:

- Dynamic Verification
- Dynamic Validation
- Evaluation

For producing a software system, first we should collect all the required specifications of a system and then analyze them.

According to this information we can design a model of the system.

This model can be inverted to codes in order to being analyzed and tested by V & V tools and techniques.

For software testing at first we should test the model of the system. For this act, we test the model by dynamic verification techniques. Then we come to the next section. Now, by using dynamic evaluation we test the system by its entire requirements.

By doing this procedure, our system is tested. Now, we are in evaluating section. We present one example in order to understand this section well.

We want to evaluate credit choosing system by this design. For this evaluation, we use statistical test that we can enhance the ability of system and reliability estimation. For enhancing these sections we have a good evaluation of system and our system will have not error in producing.

For evaluating statistical test we should put sets of data in system. See following table:

Table-1: sets of data for evaluating statistical test

Data	Login	Credit choosing	Online fee payment	acceptance
Data1	Ok	ok	ok	ok
Data2	Ok	ok	ok	ok
Data3	Ok	error	error	error
Data4	Ok	ok	error	error
Result	%100	%75	%50	%50

Our software system are composed some subsystems such as Login, Credit choosing, Online fee Payment and Acceptance of Credit. We explain procedure of subsystems test by using one set of data.

In Login subsystem, we have two objects that first object includes Username and second object includes Password of student. For testing this section, we put set of data in subsystem. For Login in the system both objects should verify input data. Process of the both objects verification is look like to AND Logic. The other words, the Username and Password should consistent each other. If these objects consistence we can go to credit choosing subsystem. By using data collection, we put requirement data in subsystem.

The procedure of credit choosing has been completed. Now, we come to the online fee payment section. In this section, the cost of credit choosing must be in the bank account. Then it's important that the information of bank card, account number and password were the same one and did not expire. After paying the fee, we are in the acceptance section, which presents us the results of the credit choosing.

By this act, all the information of the 1st data collection was tested and was shown the correct results. This procedure is repeated in the row of 2st data collection, too.

We can observe that all amounts of data are consistent with the system and the system doing all of the procedures completely.

But in 3st data collection after the Login process the error will occurred in credit choosing and data do not consistent with system. Also, online fee payment and acceptance sections has confront with error, too.

In 4st data collection test, Login section and credit choosing section have been tested completely, but online fee payment section has error. Maybe, the card information is wrong. We inspect the information and we understand that the bank information is true. Where is the problem?

Come back to the system lifecycle. Before evaluation part, in validation part we see that we should use internet explorer for online fee payment so we conclude validation has important role in evaluation.

In table 1, result section, the percentage of reliability of each subsystem has complete validation and evaluation because their reliability is %100. The reliability of credit choosing subsystem is %75 and the reliability of online fee payment and acceptance subsystems are %50. So, these percentages are problems in system. We should inspect validation and verification sections. For these sections, we can suggest a new solution that we should use dynamic and static validation and verification techniques together that we get better conclude in evaluation.

5. Conclusion

In this paper, we explain one of the V&V techniques. So, we describe type of software testing briefly. By using the statistical test, we present a model that this model influences in reliability of software testing lifecycle. Also, by presentation a case study, we explain our model and we conclude that the factors of reliability and performance of software can be achieve high ability by using dynamic and static V&V techniques.

1/27/2012

Corresponding Author:

Alireza Souri
Department of Computer Engineering
University college of Nabi Akram
Tabriz, Iran
E-mail: a.souri@ucna.ac.ir

References

1. Alireza souri, mohammad ali sharifloo and monire norouzi. research on effective factors on verification results in software systems, *international journal of computer science issues*, Vol. 8, Issue 5, No 2. Pp-464-468, September 2011.
2. Ian Somerville, software engineering, seventh edition, Addison - Wesley, 2004.
3. Roger Pressman, Software Engineering, a Practitioner Approach, 6th ed., McGraw Hill, 2005.
4. <http://hotrecruiter.com>