

IMPLEMENTATION OF TEST MANAGEMENT ON OBA SYSTEM

AHMED IBRAHIM SAFANA

UNIVERSITY TEKNOLOGI MALAYSIA

IMPLEMENTATION OF TEST MANAGEMENT ON OBA SYSTEM

AHMED IBRAHIM SAFANA

A thesis submitted in fulfillment of the
Requirement for the award of the degree of
Master of computer science (Real Time software Engineering)

Center for Advance Software Engineering (CASE)
Faculty of Computer and Information System
Universiti Teknologi Malaysia

APRIL 2010

ABSTRACT

The growing complexity of today's software applications has catapulted the need for testing to new heights, shrinking development and deployment schedules. Organizations all over the world required a high turnover rates for skilled employees for software testing and related issues. Schedules are always running tight during the software system development, thereafter reducing efforts of performing software testing management. In such a situation, improving software quality becomes an impossible mission [1]. It is our belief that software industry needs a grand new software management tool to promote software testing management. This cater the needs for a reliable software testing management tool, which adopts structure behavior coalescence methodology, and is able to guide testing engineers "what to do" and "how to progress" as well. Knowing "what to do" and "how to progress" meliorates software testing management in a great deal. Perhaps, a number of varieties existed, but seems to be half-baked for their inability to synchronize the hitherto separate worlds of development and testing [2]. Once the software testing management is improved, then the software quality can also be enhanced. This research attempts to quantify the benefits of using SpiraTeam a software Test management tool in providing assurance of proper requirement test coverage, requirement traceability and improving the quality of the software development.

ABSTRAK

Pertumbuhan yang sangat kompleks dari perisian komputer pada masa sekarang ini telah melontarkan keperluan daripada pengujian ke tingkat yang lebih tinggi, mengurangi jadual pembinaan dan pelaksanaan. Pertumbuhan di seluruh dunia memerlukan satu tingkat pertukaran yang tinggi untuk pengujian perisian dan isu-isu yang berkaitan. Jadual-jadual sentiasa berjalan ketat semasa pembinaan sistem perisian, sesudah itu mengurangi usaha untuk melakukan pengurusan pengujian perisian. Dalam keadaan seperti itu, perbaiki kualiti perisian menjadi tugas yang mustahil. Kita percaya industri perisian memerlukan suatu alat pengurusan perisian yang sangat baru untuk mempromosikan pengurusan pengujian perisian. Ini mengendalikan keperluan untuk alat pengurusan pengujian perisian, yang mengutip metodologi penggabungan tingkah laku struktur, dan boleh memandu jurutera penguji melakukan "apa mesti buat" dan "bagaimana untuk maju" dengan baik. Mengetahui "apa mesti buat" dan "bagaimana untuk maju" akan mengecilkan pengurusan pengujian perisian dengan baik. Mungkin, beberapa jenis pilihan alat wujud, tetapi akan nampak separuh jadi sehingga tidak mampu untuk menyamakan dua dunia berbeza daripada pembinaan dan pengujian sehingga sekarang. Sekali pengurusan pengujian perisian itu telah ditingkatkan, kualiti perisian akan boleh ditinggikan juga. Penyelidikan ini mencuba untuk mengukur manfaat daripada menggunakan alat pengurusan pengujian perisian SpiraTeam dalam menyediakan jaminan liputan keperluan pengujian yang betul, jejukan keperluan, dan meningkatkan kualiti daripada pembinaan perisian.

TABLE OF CONTENTS

CHAPTER	TITLE	PAGE
ii	DECLARATION	
iii	DEDICATION	
iv	ABSTRACT	
	ABSTRAK	v
vii	TABLE OF CONTENTS	
xii	LIST OF TABLES	
xiii	LIST OF FIGURES	
1.	INTRODUCTION	1
	1.1 Overview of the study	1
	1.2 Background of the Study	1
	1.3 Statement Problems	2
	1.4 Objectives	3
	1.5 Scope	3
	1.6 List of Deliverables	4
	1.7 Thesis Outline	5
2.	LITERATURE REVIEW	7

2.1	Introduction	7
2.2	What is Test Management?	7
2.3	Test Management Tools	8
2.3.1	Testopi	9
2.3.2	QaTraq	9
2.3.3	Borland Tool	10
2.3.4	SilkCentral Tool	10
2.3.5	Ken Test	11
2.3.6	Quality Center	11
2.3.7	Rational Test Manager	12
2.3.8	Wip CAFÉ TMS	12
2.3.9	Choosing SpiraTeam	13
2.3.9.1	Requirement Management	13
2.3.9.2	Test Case Management	14
2.3.9.3	Release Planning	15
2.3.9.4	Iteration Planning	15
2.3.9.5	Incident Tracking	16
2.3.9.6	Task Management	17
2.3.9.7	Project and Users	17
2.3.9.8	Artifact Relationships	18
2.3.10	Comparative Study on Test Management Tools	19
2.4	Test Management Processes	21
2.4.1	Tmap	21
2.4.2	Trigent Test Management Methodology	25
2.4.2.1	Plan/Initiate	26
2.4.2.2	The Design	27
2.4.2.3	Execute Phase	27
2.4.2.4	Run Test and Write Report	27
2.4.2.5	Adapt	28
2.4.3	Heuristic Testability Method	29
2.4.3.1	Controllability	29
2.4.3.2	Observability	30
2.4.3.3	Availability	30

2.4.3.4	Simplicity	30
2.4.3.5	Stability	31
2.4.3.6	Information	31
2.4.4	Session Based Test Management Method	32
2.4.4.1	Exploratory Testing	32
2.4.4.2	Application	32
2.4.4.3	The Session Metrics	33
2.4.4.4	Test Coverage Totals	34
2.4.4.5	Debriefings	35
2.4.5	The Scan Tool Approach	36
2.4.6	Waterfall	38
2.4.7	Agile Methodology	39
2.4.7.1	Scrum	41
2.4.7.2	Extreme Programming	42
2.4.7.3	Crystal	44
2.4.7.4	Dynamic System Method	45
2.4.7.5	Feature Driven Development	46
2.4.7.6	Lean Software Development	47
2.4.8	Conclusion	48
3.	RESEARCH METHODOLOGY	53
3.1	Introduction	53
3.2	Approaches of SpiraTeam on OBA	53
3.2.1	Approach 1: Requirement Definition	54
3.2.2	Approach 2: High Level Scheduling	54
3.2.3	Approach 3: Iteration Planning	55
3.2.4	Approach 4: Task Resourcing Allocation	55
3.2.5	Approach 5: Task Execution	56
3.3	Artifacts Workflow	56
3.3.1	User	57
3.3.2	Roles	57
3.3.3	Project	58
3.3.4	Management	58
3.4	Entity Workflow	59

3.5	Incident Workflow	59
3.6	OBA as a Case Study	62
3.7	Conclusion	63
4.	DESIGN AND IMPLEMENTATION OF TEST MANAGEMENT	64
4.1	Introduction	64
4.2	Research Procedure and Design	64
4.3	System Administrator	68
4.3.1	Project	69
4.3.2	View/Edit User	69
4.3.3	Project Membership	70
4.3.4	Incident	71
4.3.5	Document	72
4.4	My Page	74
4.4.1	My Projects	74
4.4.2	My Save Searches	75
4.4.3	My Assigned Requirements	75
4.4.4	My Assigned Test Cases	76
4.4.5	My Assign Test Set	78
4.4.6	My Pending Test Runs	78
4.4.7	My Assigned Incidents	79
4.4.8	My Detected Incidents	79
4.5	Requirement Management	79
4.5.1	Requirement Details	81
4.5.2	Requirement Coverage/Test Coverage	82
4.5.3	Tasks	83
4.6	Release Planning	83
4.7	Test Cases	84
4.8	Test Execution	86
4.9	Incident Tracking	87
4.10	Conclusion	88
5	SUMMARY AND CONCLUSION	89

5.1	Summary	89
5.2	Shortcomings	93
5.3	Recommendation and further Enhancement	93
5.4	Conclusion	94

REFERENCES

97

Appendices A

98

LIST OF TABLES

TABLE NO.	TITLE	PAGE
2.1	Projects' Artifacts	18
2.2	Artifacts Description	19
2.3	Brief Analysis of Tools	20
2.4	Completed Session Report	33
2.5	Test Coverage	34
2.6	Analysis Summary of Test Management Tools	51
4.1	Research Planning	65

LIST OF FIGURES

FIGURES NO.	TITLE	PAGE
2.1	Tmap next	21
2.2	Tmap stages	22
2.3	Business design test methodologies	24
2.4	Test management support methodology	26
2.5	Activity Hierarchy	37
2.6	Work breakdown	37
2.7	Waterfall methodology	38
2.8	Scrum methodology	42
2.9	XP methodology	44
2.10	Phase transition	49
2.11	Agile methodology	49
3.1	SpiraTeam architecture	53
3.2	SpiraTeam management phases	54
3.3	The main entities of SpiraTeam	56
3.4	User login screen	57
3.5	Project view/edit	58
3.6	SpiraTeam entity relationship	59
3.7	Incidents workflow	60

3.8	Incident workflow steps	61
3.9	Workflow transition	62
3.10	OBA System composition	63
4.1	Phases in service Engineering	64
4.2	Research Flowchart	66
4.3	Installation wizard	67
4.4	Administrators page	68
4.5	View/edit user	70
4.6	Project membership	71
4.7	Edit incident type	71
4.8	Edit incidents status	72
4.9	Edit incident priority	72
4.10	Document list	73
4.11	Hierarchy of documents folder	74
4.12	My page task pane	76
4.13	My assigned test cases	77
4.14	Pending test run	78
4.15	Excel importer log in screen	80
4.16	Requirement list	80
4.17	OBA requirements	81
4.18	Requirements details	81
4.19	Requirements coverage	82
4.20	Tasks assignment window	83
4.21	Releases planning	84
4.22	Test case list	84
4.23	Test case description	85
4.24	Test steps	85
4.25	Releases	86
4.26	Test run	86
4.27	Incident list	87
5.1	Requirement summary	90
5.2	Test case summary	91
5.3	Test run summary	92
5.4	Incident summary	92

5.5	Synchronization chart	95
-----	-----------------------	----

CHAPTER 1

INTRODUCTION

1.1 Introduction

The management of the test processes has become of paramount important to the success in finding the shortcomings of the software system under development. The increase in the needs of reliable software system makes software testing necessary; hence the management of the test processes cannot be left out in order to maintain the exercise within the schedule frame.

The traditional method used in undertaking the management of the test processes is becoming unaffordable to the customers and a very difficult task to the testing engineers. The use of test management tool in other way round leads to another set of problems due to the absence of some required features. This project concentrate in studying, applying, implementing and evaluating a software test management tool (SpiraTeam) using the On-board automobile software project as a case study in order to eradicate the stated problems.

1.2 Background of the Study

Software test management is one of the direct factors of reducing the dependability of the software due to the inconsistencies and lack of conformity between the requirements and the test cases.

Software quality is very essential to business success. However, software quality means different things to different people, making it difficult to discuss on software quality. Example, for development teams, software quality means performance; scalability, functionality, reliability, accuracy and usability must all be addressed. For end users, it means IT systems that deliver “what we need, when we need it.” This terms is used worldwide and cross-platform functionality further complicate the process to another level, because team members often use different test management tools to handle a particular software project due to the features presence in one tool and absence in another one, which are all required.

This complexity, and the volume of tasks involved in managing manual software testing, automated regression and performance testing, makes formalized software test management a must, if organizations want to gain control of the process. The On-board Automobile system is a software project developed by the students at the Center for Advanced Software Engineering (CASE). The tests processes of this software are managed manually. This leads to a number of errors. It is to the believe of the author that the introduction of the SpiraTeam in such test management will enhance the quality of the development and reduce the time the scheduled for testing as we will see in the coming chapters.

1.3 Statement of the Problems

There are a lot of problems associated with the manual test management of the On-board automobile system (OBA), which makes the whole processes boring and inadequate. This leads to the abandoning of the process uncompleted. Some of the main problems include the followings:

1. Processes are ad-hoc and not repeatable across project
2. There is no visibility between test cases, requirements and defects. How do you know when you are truly ‘done’?
3. Measuring progress and productivity during testing is time consuming and difficult

4. It is hard to share information across the project and get real-time metrics regarding the quality of the system being tested
5. There is no central repository of test results from all sources.
6. Difficulty in shearing the tasks among the testing team and time consuming
7. So many incidents are left undiscovered and unreported

1.4 Objectives

The problem statement mentioned above serves as a premise to establish a set of specific objectives that will constitute major milestones of this research work. The following are the objectives of this research;

1. To study the features and the functionalities of the SpiraTeam.
2. To apply the features to On-board Automobile Software system (OBA)
3. To implement Test Management processes on the requirement of the On-board Automobile (OBA)
4. To demonstrate and evaluate the benefit of using SpiraTeam in managing the test processes of On-boat Automobile software
5. To be able to informed the testing team members their individual tasks online via e-mails

1.5 Scope

The software test management tool in question (SpiraTeam), has so many advantages and abilities, such as integration with software test run tools, central data repository system, and can allow the migration of data from either the test run tool or some data management system such as MS-excels, JIRA, Bugzilla and so on. It is also able to plan the whole project used as project plan as well as test management. The scope of this project will lay much emphasis on the software test management of the selected case study, even though, all the features of the SpiraTeam mentioned involve with each other. The focus is on the following parts;

1. Requirement Management.
2. Test case Management.
3. Requirement Test coverage
4. Release Management
5. Incidence and defect tracking

1.6 List of Deliverables

The following items are the expected deliverables to be presented to the client at the end of the processes, for the onward review and the understanding of the achievement made in managing the testing of the software, and the satisfaction of the software requirement's needs.

1. Requirement Report
 - a. Requirement Summary
 - b. Requirement Detailed
 - c. Requirement Plan
 - d. Requirement Traceability
2. Test Case Report
 - a. Test Case summary
 - b. Test Case Details
 - c. Test Set Summary
 - d. Test Set Details
 - e. Printable Test scripts
 - f. Test Run Details
 - g. Test Run Summary
3. Incidence report
 - a. Incidence Summary
 - b. Incidence Detail
4. Task report
 - a. Task Summary
 - b. Task Detail

5. Release Report
 - a. Release Summary
 - b. Release Detail
 - c. Release plan
6. Requirement Graphs
 - a. Requirement Summary
 - b. Requirement Coverage
7. Test Case Graphs
 - a. Test Case Summary
 - b. Test Case Run Summary
 - c. Test Run Progress Rate
8. Incidence Graphs
 - a. Incident Summary
 - b. Incident Progress
 - c. Incident Cumulative Count
 - d. Incident Aging
 - e. Incident Turnaround Time
9. Task Graphs
 - a. Task Summary
 - b. Task Velocity
 - c. Task Burnup
 - d. Task Burndown
10. Test Plan
11. Test Case description
12. Test Summary Description

1.7 Thesis Outline

Chapter 1: This chapter introduces the topic as a whole, outlines the background of the study, problems observed from the manual system used in managing the tests of the On-board automobile, the scope of the thesis, the objective, list of deliverables and the activities to be perform on the coming chapters.

Chapter 2: This chapter describes the literature review of software test management, test management tools, SpiraTeam and several discussions on the specific issue regarding test management. The features of the SpiraTeam and test management tools will also be discussed. This leads to the improvement of the management of tests by means of the selected tool for greater performance in achieving a better result.

Chapter 3: Research methodology. Even though, the project work is more on applying a software management tool on a specific software project (OBA), to achieve a good testing processes, prior to a research on topic that focus on deep research about some findings. This chapter will outline the methodologies used in evaluating the tool and the suitable software development life cycles it support as well as the emergence of the test management from the project plan and development phases. The workflow of the incidence, artifacts and the tool will be discussed as well.

Chapter 4: The design and implementation of the SpiraTeam on On-board automobile will be discussed on this chapter, how the application took place, the snapshots, the stages in managing the tests, the test run progress, the workflows agreed in solving a particular type of incidence and the method of finding a particular report.

Chapter 5: The conclusion and recommendation on the activities carried out so far will be discussed here, the result of the application and implementation of the test management on the case study will be evaluated, and the final decision on the use of the tool to adopt the test management in CASE will be discussed as well.

REFERENCES

1. H. Ohtera, S. and Yamada, *Optimal allocation & control for software testing resources*, IEEE Transection on reliability, vol. 39, pp. 171–172, June 1999.
2. L. Lui, and D. J. Robson, *A support Enviroment for the managing of software testing*, IEEE conference on software engineering 1992.
3. N. Kicillof. W. Garieskamp and V. Braberman, *Achieving both model and code coverage with aytomated gray-box testing*, AMOST'07 ACM London, UK July 2007, pp. 1–5.
4. F. Li, W. M. Ma, and A. Chao, *Architecture centric approach to enhance software testing management*, IEEE Eight International Conference on Intelligent Systems Design and Application 2008
5. P. M. Kamde, V. D. Nandavadekar and R. G. Pawar, *Value of test cases in software testing*, IEEE International Conference On Management of Innovation and Technology, 2006.
6. J. W. Cangussu, R. A. Decarlo, and A. P. Mathur, *Monitoring the software Test process using statistical process control: A logarithmic approach*, ESEC/ESE 03, September 1-5 2003 Helsinki, Finland
7. Y. Jun-feng, Y. Shi, L. Ju-bo, X. Dan, J. Xiang-yang, *Reflective architecture based software testing management model*, 2006 IEEE International Conference on Management and Technology.
8. H. Pieire, *Testing networking* ,Vol. 39, no. 2, 1990 june, P. 172:
9. P. Farell-Vinay, *Manage software testing*, Auerbach publication tayloy & Fracis group, vol. 2, Dec. 2008, pp. 39-89.
10. I. Burnstain, *Practica software testing*, Springer Verley publication, vol. 2, Dec. 2003, pp. 39-89.
11. White paper Gain control of the chaotic software test management , *Software Test Management*

12. Test management approaches and methodology, *Tmap Sogeti*, retrieved February 2010 from <http://www.tmap.net/info>
13. Scan tool net, *Scan tool management tool*, retrieved in January 2010 from <http://www.scantool.net/scan-tools/>
14. Testing services, *Tragen Tool* , retrieved in February 2010 from <http://www.trigent.com/services/testing/>
15. N. Palani, *Test Management tool review*, Wipro technology & Electronic. Unpublished paper.
16. Y. shen, and J. Liu, *Research on the application of data mining in software testing and defects analysis*, academy publosher vol: pp15, pp 29.
17. SpiraTeam software *Administrators and user guides* <http://www.inflactar.com>
18. M. Grindal, J. Offutt and J. Mellin “On the testing maturity of software producing organizations,” *Proceedings of the Testing: Academic & Industrial Conference-Practice And Research Techniques (TAIC PART ‘06) 2006 IEEE*
19. Y. shen, and J. Liu “*Research on the application of data mining in software testing and defects analysis*,” 2009 Second International Conference on Intelligent Computation Technology and Automation
20. M. A. Hennell, D. Hedley, and I. J. Riddell, “*Assessing a class of software tools*,” International Conference On Management of Innovation and Technology 1984.
21. P. M. Duernberger, “*Software testing application in a computer science curriculum*,”
22. White Paper “*taming software quality complexity with virtual automation*”.
23. Mashable, *English Wikipedia Reaches 3 million Articles*, Retrieved in March 2010 from <http://mashable.com/2010/01/02/wikipedia-3million-articles>
[Retrieved in February 2010.](#)
24. Techrunch, *twitter reaches 445 million people Worldwide in June Comscore*, Retrieved in March 2010.
25. Facebook, *Press Statistics* Retrieved in march 2010 from <http://facebook.com/press/infor.testing?statistics>

APPENDIX A

SpiraTeam Installation Guide