A HYBRID TESTING APPROACH FOR TEST CASE GENERATION OF WEB BASED FUNCTIONAL REQUIREMENTS

SINA BINTI ALI

Universiti Teknologi Malaysia

A HYBRID TESTING APPROACH FOR TEST CASE GENERATION OF WEB BASED FUNCTIONAL REQUIREMENTS

SINA BINTI ALI

A thesis submitted in fulfilment of the requirements for the award of the degree of
Master of Science (Computer Science)

Faculty of Computing
Universiti Teknologi Malaysia

March 2013

Dedicated to my parent and all family members
and my nieces and nephews that made my life worth living
and a best friend who has always believed in me

ACKNOWLEDGEMENT

First of all, I would like to warmly express my greatest gratitude to Almighty God for His blessing and given strength that lead to the completion of this research. I would like to sincerely express my gratitude to my supervisor Assoc. Prof. Dr. Norazah Yusof for her continuous motivations, advices, encouragements and supports from the beginning till the end of my studies, till I was able to develop a deep understanding of the research topic. Thank you for supervising me and providing important feedbacks and encouragements for my studies. I also sincerely express my appreciation to Puan Shahliza Abdul Halim for her supports, ideas and encouragement in my thesis writing. And my sincere and deepest appreciation goes to my best friend Yavuz Selim Şengöz, whom has always and to every extent supported, encourages and always believe in me. I am also thankful to my dearly friends Ferhan Kılıçer, Aziah Abdollah, Nazeema Abdul Rahim, Afsaneh Sheikhi, Zabha Ibrahim, Nurul Badriah Abu Bakar for their advices and supports, and to all lab mates at the Software Engineering Laboratory 1 for stimulating discussions and creating friendly atmosphere. The sincerest appreciation goes to my parents Encik Ali Omar and Puan Siti Halijah Poongodi for their continues support and encouragement, both psychological and financial and I owe my sincerest gratitude to my brother Azni and my sister in-law Siti Fauziah, my sister Sila and my aunties Mrs. J. Stella and Mrs. Kumari and my lovely twin brothers Amzal and Amzannoh for never letting me down and always giving me motivation and courage to look on the bright side every time I felt unmotivated.

ABSTRACT

Web-based applications have started to play an important role in industries and academia due to their use of online applications that are accessible via the internet browser. This factor indirectly shows that there is a demand for a proper testing process to ensure the quality of a web-based application. Basically, the test phase is described as one of the most significant phases for quality assurance in software projects. Unfortunately, the test phase is usually not strictly executed and is totally dependent on a tester's skills. Although literature has shown that various testing approaches have been proposed to test web-based applications, but these approaches provided insufficient information for functional requirements as well as introduced massive steps during testing. Therefore, this research proposed a hybrid testing approach called Hybrid Functional-System Testing Approach (HyF-sTa) to test the functional requirements of a web-based application using systematic steps. This is done by analysing potential testing approaches for web-based application, developing and implementing a hybrid testing approach model based on a case study, and validating the results by comparing them with previous testing results. The results of HyF-sTa were compared with Ad-Hoc based on three chosen metrics: Test Coverage, Test Effectiveness and Test Effort. The validation results proved that the HyF-sTa approach using the systematic steps successfully obtained higher values for all three chosen metrics as compared to the Ad-Hoc testing manner.

ABSTRAK

Aplikasi berasaskan web telah mulai memainkan peranan penting dalam industri dan akademik oleh kerana kegunaan aplikasinya yang boleh dicapai melalui pelayar internet. Faktor ini secara tidak langsung menunjukkan bahawa terdapat permintaan bagi proses pengujian yang sewajarnya untuk memastikan kualiti aplikasi berasaskan web. Pada asasnya, fasa pengujian digambarkan sebagai salah satu fasa yang paling penting untuk jaminan kualiti dalam projek perisian. Namun, fasa pengujian biasanya tidak dilaksanakan dengan ketat dan bergantung kepada kemahiran penguji sepenuhnya. Walaupun literatur telah menunjukkan bahawa pelbagai pendekatan pengujian telah dicadangkan untuk menguji aplikasi berasaskan web, tetapi pendekatan-pendekatan ini menyediakan maklumat yang tidak mencukupi untuk pengujian keperluan kefungsian serta memperkenalkan langkah-langkah yang sukar semasa pengujian. Oleh itu, penyelidikan ini mencadangkan pendekatan pengujian hibrid yang dipanggil Hibrid Fungsian Sistem Ujian Pendekatan (HyF-sTa) untuk menguji keperluan fungsian aplikasi berasaskan web menggunakan langkah-langkah yang sistematik. Ini dilakukan dengan menganalisis pendekatan pengujian yang berpotensi untuk aplikasi berasaskan web, membangunkan dan melaksanakan pendekatan pengujian model hibrid berdasarkan kajian kes, dan mengesahkan keputusan dengan membandingkannya dengan keputusan ujian sebelumnya. Keputusan HyF-sTa dibandingkan dengan Ad-Hoc berdasarkan tiga metrik yang dipilih: Ujian Liputan, Ujian Keberkesanan dan Ujian Usaha. Keputusan pengesahan membuktikan bahawa pendekatan HyF-sTa menggunakan langkah-langkah yang sistematik berjaya mendapatkan nilai yang lebih tinggi untuk ketiga-tiga metrik yang dipilih berbanding dengan cara pengujian Ad-Hoc.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

WBIS	*Web Based Information System*
MBT	*Model Based Testing*
UML	*Unified Modelling Language*
HTML	*Hyper-Text Markup Language*
ASP	*Active Server Pages*
DBMS	*Data-Base Management System*
LAN	*Local Area Network*
CSA	*Client-Server Application*
SUT	*System Under Test*
SDL	*Specification and Description. Language*
OMG	*Object Management Group*
ATM	*Auto Teller Machine*
ADT	*Activity Dependency Table*
ADG	*Activity Dependency Graph*
SDG	*Sequence Diagram Graph*
PACS	*Picture Archiving and Communication System*
UIT	*Use Interaction Test*
RRS	*Railway Reservation System*
ITS	*Industrial Training System*
HyF-sTa	*Hybrid Functional-System Testing Approach*
UTM	*Universiti Teknologi Malaysia*
SRS	*System Requirements Specification*
IEEE	*The Institute of Electrical and Electronics Engineers*
NDT	*Node Description Table*

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Overview

Web-based application rapidly turn out to be the most demanded application in all over the world, the ability of the application to be accessed online via the internet browser is really seizing industry and academic focus toward it. This factor indirectly demands a proper testing process to ensure quality of the developed web-based application (Xu et al., 2005). This chapter presents an introduction of this study. Problems background of this study were discussed in Section 1.2 and in Section 1.3 the problems statement was defined, followed by Section 1.4 introduced the aim of this study and in Section 1.5 the objectives of this study were listed and Section 1.6 is stating the scopes and 1.7 justify the significance of the study. Finally, Section 1.8 which summarizes the outline of this study's thesis.

1.2 Research Background

One type of commonly demanded web-based application is Web Based Information System (WBIS), this kind of application stores information in the database and developed for particular organization in order to manage and computerized the work process. However, WBIS often encounter problem when the testing are not adequately done. This is due to the characteristics of WBIS development process and delivering is faster than traditional application development due to the market pressure and short time to market and this factor indirectly affecting the quality of the product (Hieatt and Mee, 2002; Heiser, 1997).

Testing on quality perspective, highlights that testing is a quality control function that conducted with the intention of finding errors (Horch, 2003). Testing can be defined as the process of finding technique of verification and validation as well. Verification refers to "Are we building the product right?" means the product that developed is as what the end-user request. Whereas validation refers to "Are we building the right product?" which means the product that developed is matched to end-user requirement (Boehm, 1981).

Generally, quality of application product can be measured from four aspects or goals, that are requirement quality, design quality, code quality and quality control effectiveness and each of this goals consists of its own attributes and metrics (Hyatt and Rosenburg, 1996). In WBIS, requirement is very important attribute that must be tested. Moreover, testing often carried out in general ad-hoc approach and the test cases are often developed in unstructured and unsystematic approach (Ryser and Glinz, 1999). This problem also occurs due to overlooked of the testing importance where most developers presume testing as a common casual (Hieatt and Mee, 2002). However, apart from the mind set of developers the quality of existing web-based application is often poor due to insufficient comprehensive testing approach defined (Ricca and Tonella, 2005).

Strecker and Memon (2007) stressed that when selecting a testing approach or technique, the practitioners or tester should be focused to detect the most faults on the program that targeted to test. On the other hand, there are lack of methods to conduct the key process of testing even there are several of methodological and technological proposal that proposed from industry and academic sectors. This lack might be resulted by immature of the field (Di-Lucca et al., 2002). Effective testing of web-based application should in fact base on clear testing strategies that defined the heuristic or algorithms to create test cases from the test model.

Testing is extremely significant in verifying and validating the system. However, test preparation often prepared very late or at the end of the software development lifecycle. Another point is existing testing approach is hard to apply, this is due to the lack of understanding on existing testing approaches (Kasnia and Mehta, 2011). The testing approach would be easier to be applied if implementation manner is introduced detail and do not require an inappropriate overhead or intolerable cost (Ryser and Glinz, 1999).

1.3 Research Problems

Although testing phase is introduced in the literature as one of the most significant for phase to evaluate the quality of software projects, this phase is often unusually well followed and occasionally reliant on individual expertise. In reality, web-based applications were often developed without following a formalized process and the requirements are not elicited properly and the architecture and detailed design of the system are not considered (Ricca, 2004). Even though, there are numerous of approaches and guidelines that proposed for testing but in most cases it do not cover sufficient portions of the software product (Tran and Chechik, 2001).

Moreover, according to Mustafa et al. (2007), existing techniques seem insufficient for web-based application because the problem associated with the web-based application characteristic. Another reason is organisational tend to move fast to the implementation phase and deliver the system to the production without able to adequately testing due to short time to market (Hieatt and Mee, 2002). Moreover, the existing techniques need some enhancements because many existing testing approaches were hard to be practiced due to many factors such as unrealistic expectation and organizational issues (Mustafa et al., 2007). And existing approaches also inadequately highlight the process in detail where the processes are very massive and there is still gap between "state-of-art" and "state-of-practice" (Kasnia and Mehta, 2011). In web-based application development, massive testing approach cannot be utilized due to its short time to market characteristic. Therefore, an approach is needed to enable the testing particularly test case generation to be planned at an early stage.

1.4 Research Question

The aim of this study is to propose hybrid testing approach for testing functional requirements of web-based information system using model based testing technique in term of test effectiveness, test coverage and test effort.

The research questions to be answered are:

i. What are the challenges in testing web-based application?
ii. Which Unified Modelling Language (UML) behavioural diagrams used to assist tester to generate test cases from functional requirement?
iii. How test cases generated from chosen UML behavioural diagram?

1.5 Objectives

In order to complete this study, some objectives has been stated as follows:

i. To analyse potential testing approaches for web-based application.
ii. To develop and implement hybrid testing approach as a model based on the type of web-based application case study.
iii. To validate the proposed hybrid testing approach result with previous testing results.

1.6 Scopes

In order to keep the study focused only in concerned area, the study area will be only within defined scopes that are:

i. It focused on generating test cases only from functional requirements.
ii. It focused to derive the test cases manually.
iii. It will not focus on automation process.
iv. It focused the testing only on one subsystem from the chosen we-based application case study.

1.7 Significance of the Research

Even though the test phase is described in the literature as one of the most significant for quality assurance in software projects, this test phase is not usually well followed and totally dependent on individual skills. In fact, most web-based applications have been so far developed without following a formalized process.

And organisational tend to move fast to the implementation phase and deliver the system to the production without able to adequately testing due to short time to market (Hieatt and Mee, 2002). In web-based application development, massive testing approach cannot be utilized due to its short time to market characteristic. Therefore, an approach is need to enable the test case generation to be planned at an early stage of the software development life cycle, allowing software developer to carry out coding and testing in parallel. For these reason, model-based test case generation methodology becomes an obvious choice in software industries (France et al., 2006) therefore, it is focused in this study. Taking into account these issues, this study proposed a hybrid testing approach to generate test cases from web-based information system functional requirements using Model-Based Testing (MBT) technique.

1.8 Organisation of Thesis

The thesis is divided into following chapters. Chapter 1 introduces the problem background, motivation, aim, objectives and scope of this study. Chapter 2 describes the MBT process in general and gives a short introduction to the UML, UML behavioural diagrams and related works. Chapter 3 provides an insight on research operational framework, case study and previous ad-hoc testing results. Chapter 4 introduces the construction of proposed hybrid testing approach in detail and Chapter 5 is on implementation and validation of proposed approach. And finally, Chapter 6 discussed on the conclusion and contribution of this study.

REFERENCES

Agarwal, B. B., Tayal, S. P., and Gupta, M. (2010). Software Engineering and Testing. *Infinity Science Press, Jones and Bartlett, Hingham, Toronto.*

Alalfi, M. H., Cordy, J. R. and Dean, T. R. (2009). Modelling Methods for Web Application Verification and Testing: State of the Art. *Software Testing Verification and Reliability* 19(4), 265-296.

Baresi, L., Garzotto, F. and Paolini, P. (2001). Extending UML For Modeling Web Applications. *System Sciences'01. Proceedings of the 34th Annual Hawaii International Conference*.

Basanieri, F., Bertolino, A., and Marchetti, E. (2002). The cow suit approach to planning and deriving test suites in UML projects. *Proceedings of Fifth International Conference on the UML* 60, 38-397.

Berenbach, B., Paulish, D., and Kazmeier, J. A. (2009). Software and Systems Requirements Engineering in Practice. *The McGraw-Hill Companies Inc*.

Bertolino, A., Marchetti, E. and Muccini, H. (2005). Introducing a Reasonably Complete and Coherent Approach for Model-based Testing. *Electronic Notes Theory of Computing Sci.* 116, 85-97

Biswal, B., Nanda, P., Mohapatra, D. (2008). A Novel Approach for Scenario-Based Test Case Generation. *IEEE Society*.

Blackburn, M., Busser, R. and Nauman, A. (2004). Why model-based test automation is different and what you should know to get started. *International Conference on Practical Software Quality and Testing (PSQT/PSTT'2004)*

Boehm, B. W. (1981). *Software Engineering Economics*. Englewood Cliffs.

Boghdady, P. N., Badr, N. L., Hashem, M. and Tolba, M. F. (2011). A Proposed Test Case Generation Technique Based on Activity Diagrams. *International Journal of Engineering & Technology IJET-IJENS* 11(03).

Brumbulli, M., Topçiu, B. and Dalaçi, A. (2008). SMIS: A Web-Based School Management Information System. *International Scientific Conference Computer Science*.

Correa, A. L. and Werner, C. M. L. (2004). Precise specification and validation of transactional business software. *Proceedings. 12th IEEE International Requirements Engineering Conference,* 16-25.

Cockburn, A. (1995). Structuring Use Cases with Goals. *Technical report of Human and Technology*. Accessed via http://members.aol.com/ acocburn/ papers/ usecases.htm

Cockburn, A. (2001). *Writing Effective Use Cases*. Addison-Wesley.

Craig, R. D. and Jaskiel, S. P. (2002). *Systematic Software Testing*. Artech House.

Di-Lucca, G. A., Fasolino, A. R., Faralli, F. and De Carlini, U. (2002). Testing Web applications. *Software Maintenance, 2002. Proceedings. International Conference on*. 310-319.

Di-Lucca, G. A. and Fasolino, A. R. (2006). Testing Web-based applications: The state of the art and future trends. *Information and Software Technology* 48(12), 1172-1186.

Dias-Neto, A. C. and Travassos, G. H. (2009). Model-based testing approaches selection for software projects. *Information and Software Technology* 51(11), 1487-1504.

El-Far, I.K. and Whittaker,J.A. (2001). *Model-Based Software Testing*, Encyclopedia of Software Eng., J.J. Marciniak, John Wiley & Sons. 825–837.

Escalona, M. J., Gutierrez, J. J., Mejias, M., Aragon, G., Ramos, I., Torres, J. and Dominguez, F. J. (2011). An overview on test generation from functional requirements. *Journal of Systems and Software* 84(8), 1379-1393.

Fraikin, F. and Leonhardt, T. (2002). SeDiTeC-testing based on sequence diagrams. *Proceedings of 17th IEEE International Conference on Automated Software Engineering (ASE 2002)*, 261-266.

France, R. B., Ghosh, S., Dinh-Trong, T. and Solberg, A. (2006). Model-Driven Development Using UML 2.0: Promises And Pitfalls. *Computer* 39(2), 59-66.

Graham, D., Veenendaal, E., Evans, I., and Black, R. (2010). *Foundations of Software Testing ISTQB Certification*. International Software Testing Qualifications Board.

Gupta, P. and Surve, P. (2011). Model based approach to assist test case creation, execution, and maintenance for test automation. *Proceedings of the First International Workshop on End-to-End Test Script Engineering, ser. (ETSE'11)*, 1–7.

Hasling, B., Goetz, H., and Beetz, K. (2008). Model Based Testing of SystemRequirements using UML Use Case Models. *Proceedings of the International Conference on Software Testing, Verification, and Validation.*

Hassan, H. A. and Yousif, Z. E. (2010). Generating Test Cases for Platform Independent Model by Using Use Case Model. *International Journal of Engineering Science*.

Heinecke, A., Bruckmann, T., Griebe, T. and Gruhn, V. (2010). Generating Test Plans for Acceptance Tests from UML Activity Diagrams. *17th IEEE International Conference and Workshops on Engineering of Computer Based Systems (ECBS 2010)*. 22-26 March 2010. 57-66.

Heiser, J. E. (1997). An Overview of Software Testing. *IEEE Autotestcon Proceedings*, 204-211.

Heumann, J. (2001). Generating test cases from use cases. *Technical report, Rational Software*. Accessed via http: //www.ibm.com/developerworks/ rational/library/content/RationalEdge/jun0l/GeneratingTestCasesFromUseCa ses June01.pdf

Hieatt, E. and Mee, R. (2002). Going faster: Testing the web application. *IEEE Software*, 60-65.

Hierons, R. M. (2001). Testing A Distributed System: Generating Minimal Synchronised Test Sequences That Detect Output-Shifting Faults. *Information and Software Technology* 43(9), 551-560.

Holt, J. (2004). *UML for Systems Engineering - Watching the Wheels*. IET

Horch, J. W. (2003). *Practical Guide to Software Quality Management*. Artech House.

Hutcheson, M.L. (2003). *Software Testing Fundamentals: Methods and Metrics*. John Wiley & Sons Inc.

Hyatt, L. and Rosenberg, L. (1996). A Software Quality Model and Metrics for Identifying Project Risks and Assessing Software Quality. *Proceedings of 8th Annual Software Technology Conference*.

IEEE (2008). *IEEE Standard for Software and System Test Documentation*. Computer Society.

Jablonski, S., Petrov, I., Meiler, C. and Mayer, U. (2004). *Guide to web application and platform architectures*. Springer-Verlag New York Inc.

Jorgensen, P. (2002). *Software Testing: A Craftsman's Approach*. CRC Press.

Kaner, C. (2003). The Power of' What If... and Nine Ways to Fuel Your Imagination: Scenario Testing. *Software Testing and Quality Engineering* 5, 16-22.

Kasnia, S., and Mehta, D. (2011). Validating and Testing Software System using Use Case Based Approach. *International Journal of Computational Engineering & Management (IJCEM'11)*, 44–53.

Kobryn, C. (1999). UML 2001: A Standardization Odyssey. *Communications of the ACM* 42(10), 29-37.

Kundu, D., Sarma, M. and Samanta, D. (2008). A Novel Approach to System Testing and Reliability Assessment Using Use Case Model. *Proceedings of the 1st India software engineering conference*: 147-148.

Kundu, D. and Samanta, D. (2009). A novel approach to generate test cases from UML activity diagrams. *Journal of Object Technology* 8(3), 65-83.

Lang, P., Card, M., Saalwaechter, S. and Godkin, T. (1995). Application of test effectiveness in spacecraft testing. *Reliability and Maintainability Symposium, 1995. Proceedings., Annual*. 486-495.

Li, J. and Xing, D. (2011). User Session Data based Web Applications Test with Cluster Analysis. *Advanced Research on Computer Science and Information Engineering*, 415-421.

Liu, H. and Kuan Tan, H. B. (2009). Covering Code Behavior on Input Validation in Functional Testing. *Information and Software Technology* 51(2), 546-553.

Luo, L. (2009). Software Testing Techniques, Technology Maturation and Research Strategies. *Class Report, Institute for Software Research International, Carnegie Mellon University, Pittsburgh, USA*.

Mingsong, C., Xiaokang, Q. and Xuandong. L. (2006). Automatic Test Case Generation for UML Activity Diagrams. *Proceedings of the International Workshop on Automation of Software Test.*

Mitra, P., Chatterjee, S. and Ali, N. (2011). Graphical analysis of MC/DC using automated software testing. *3rd International Conference on Electronics Computer Technology (ICECT'2011),* 145-149.

Mustafa, G., Shah, A. A., Asif, K. H. and Ali, A. (2007). A Strategy for Testing of Web Based Software. *Information Technology Journal* 6(1), 74-81.

Nebut, C., Fleurey, F., Le Traon, Y. and Jézéquel, J. M. (2006). Automatic test generation: A use case driven approach. *Software Engineering, IEEE Transactions on* 32(3), 140-155.

Neogy, R. and Dharan, H. (1986). Measures of Test Effectiveness in a Communications Satellite Program. *IEEE Journal on Selected Areas in Communications.*

OMG. (2009). Unified Modelling Language Specification. Accessed via http://www.omg.org/.

Perry, W.E. (2006). *Effective Methods for Software Testing*. John Wiley & Sons Inc.

Pressman, R. S. (2010). *Software engineering: a practitioner's approach*. McGraw-Hill Higher Education.

Reza, H., Ogaard, K. and Malge, A. (2008). A Model Based Testing Technique to Test Web Applications Using Statecharts. *Proceedings of the Fifth International Conference on Information Technology: New Generations*, 183-188.

Ricca, F. (2004). Analysis, testing and re-structuring of Web applications. *Software Maintenance, 2004. Proceedings. 20th IEEE International Conference on*. 11-14 Sept. 2004. 474-478.

Ricca, F., and Tonella, P. (2005). Web Testing: A Roadmap for the Empirical Research. *Web Site Evolution, 2005. (WSE 2005). Seventh IEEE International Symposium on*. 63-70.

Rob P, Coronel C. (2004). *Database Systems: Design Implementation and Management*. Course Technology.

Rosenberg, L.H., Hammer, T.F., and Huffman, L.L. (1998). Requirements, Testing and Metrics. *15th Annual Pacific Northwest Software Quality Conference*.

Ryser, J. and Glinz, M. (1999). A practical approach to validating and testing software systems using scenarios. *Quality Week Europe (QWE'99)*.

Saglietti, F., Oster, N. and Pinte, F. (2008). White and grey-box verification and validation approaches for safety-and security-critical software systems. *Technical Report* of *Information Security* 13(1), 10-16.

Sarma, M., Kundu, D. and Mall, R. (2007). Automatic test case generation from UML sequence diagram. *International Conference on Advanced Computing and Communications (ADCOM 2007)*, 60-67.

Sarma, M., Murthy, P.V.R., Jell, S., and Ulrich, A. (2010). Model-based testing in industry: a case study with two MBT tools. *Proceedings of the 5th Workshop on Automation of Software Test, AST'10*, 87–90

Schieferdecker, I., Dai, Z., Grabowski, J. and Rennoch, A. (2003). The UML 2.0 testing profile and its relation to TTCN-3. *Testing of Communicating Systems*, 609.

Shaw, M. (2002). What makes good research in software engineering? *International Journal on Software Tools for Technology Transfer (STTT)* 4(1), 1-7.

Shao, D., Khurshid, S. and Perry, D. E. (2007). A Case for White-box Testing Using Declarative Specifications Poster Abstract. *Testing: Academic and Industrial Conference Practice and Research Techniques (TAICPART-MUTATION 2007),* 137.

Strecker, J. and Memon, A. M. (2007). Faults' context matters. *Proceedings of 4th International Workshop on Software Quality Assurance (SOQUA '07)*

Sudhakar, G. P. (2010). *Elements of Software Project Management*. PHI Learning Private Limited.

Swain, S. K., Mohapatra, D. P. and Mall, R. (2010). Test case generation based on use case and sequence diagram. *International Journal of Software Engineering, IJSE* 3(2), 21-52.

Tran, H. and Chechik, M. (2001). Test generation using model checking. *European Conference on Software Maintenance and Re-engineering (CSMR2001)*.

Utting, M. and Legeard, B. (2007). *Practical Model-Based Testing: A Tools Approach*. Morgan Kaufmann.

Wang, X., Zhou, B., and Li, W. (2010). Model Based Load Testing of Web Applications. *International Symposium on Parallel and Distributed Processing with Applications*, (C), 483–490

William, G. J. H. and Alessandro, O. (2007). Improving test case generation for web applications using automated interface discovery. *Proceedings of the 6th Joint Meeting of the European Software Engineering Conference And The ACM SIGSOFT Symposium On The Foundations Of Software Engineering*.

Windle, D. R. and Abreo, L. R. (2003). *Software Requirements Using the Unified Process: A Practical Approach*. Prentice Hall.

Xu, L., Xu, B., and Jiang, J. (2005). Testing Web Applications Focusing on Their Specialties. *ACM SIGSOFT Software Engineering Notes*, *30*(1), 1–6.