



UNIVERSITI PUTRA MALAYSIA

***ADAPTING AGENT TECHNOLOGY IN MANAGING
REQUIREMENTS REUSE PROCESS***

WAN NURSYAMSIAH WAN JUSOH

FSKTM 2017 23



**ADAPTING AGENT TECHNOLOGY IN MANAGING
REQUIREMENTS REUSE PROCESS**

By

WAN NURSYAMSIAH WAN JUSOH

**Thesis Submitted to School of Graduate Studies,
University Putra Malaysia, in Fulfillment of the
Requirement for the Degree of Master of Computer Science**

January, 2017

DEDICATION

Alhamdulillah, my grateful and praises to the Almighty of Allah who has inspired, strengthened, guided and ease the implementation of this project. I dedicate this dissertation to my husband and family for their countless supports through this journey to finish up the thesis as part of requirement fulfillment for Master of Computer Science. I also would like to express appreciation from my deepest heart to all my family members and friends who have assisted and supported me during this postgraduate study.

ABSTRACT

Requirement Engineering (RE) is a process to define, document and maintain software requirements. Requirement process involve a great deal of works from requirement elicitation to requirement documentation, thus reuse requirements become an alternative to speed up the process. Requirements reuse is an approach to use existing requirements documents to reduce the general effort inside Software Development Life Cycle (SDLC). However proper approach for small-sized enterprises is not adequately explored. Hence improvement and systematic management approach is required for requirements reuse in small-sized enterprises. A model develop by a team of research on software requirements catalog only focus on functional requirements. As such this research propose a well-defined and mature reuse methods by integrating non-functional requirements in the model. The first part of this project study the requirements reuse process and existing models. In the next activity, a new model is proposed to include non-functional requirements in the process. Lastly a prototype is developed by implementing agent technology and evaluation is done to prove the quality of requirements reuse. The results show that the proposed model have a better quality compared to previous model. As correct requirements prevent software failure, it will assist in delivering a successful software and shorten the SDLC.

ABSTRAK

Requirement Engineering (RE) adalah satu proses untuk menentukan, mendokumen dan menyalurkan keperluan sistem. Proses ini melibatkan banyak kerja bermula daripada cara memperolehi keperluan sehingga proses mendokumenkan keperluan. Oleh itu, penggunaan semula keperluan menjadi alternatif untuk mempercepatkan proses ini. Cara ini adalah dengan menggunakan keperluan sedia ada untuk mengurangkan kerja dalam rangkaian pembangunan sistem (SDLC). Walaubagaimanapun pendekatan ini masih belum diteroka sepenuhnya untuk syarikat pembangunan perisian berskala kecil. Oleh itu peningkatan dan pendekatan pengurusan yang sistematik amatlah diperlukan di dalam bidang ini. Sebuah model telah dicipta oleh satu pasukan penyelidikan dengan memanfaatkan keperluan perisian katalog namun usaha hanya tertumpu kepada keperluan fungsi (*functional requirement*) sahaja. Oleh itu kajian ini mencadangkan kaedah guna semula yang lebih baik dengan memasukkan keperluan bukan fungsi (*non-functiona requirements*) ke dalam model. Bahagian pertama projek ini mengkaji kaedah penggunaan semula keperluan dan model yang telah dicadangkan sebelum ini. Seterusnya, model baru dicadangkan dengan memasukkan keperluan bukan fungsi (*non-functiona requirements*) di dalam penggunaan semula keperluan. Akhir sekali, sebuah prototaip dibangunkan dengan mengaplikasikan teknologi ejen dan penilaian dibuat untuk membuktikan kualiti penggunaan semula keperluan. Hasil kajian menunjukkan bahawa model yang telah dicadangkan mempunyai kualiti yang lebih baik berbanding model sebelum ini. Oleh kerana keperluan yang betul dapat menghalang kegagalan sistem, diharap kajian ini dapat membantu mengatasi masalah ini dan sekaligus memendekkan rangkaian pembangunan sistem (SDLC).

ACKNOWLEDGEMENT

I would like to express my immense gratitude to Almighty of Allah for the wellness and mercies for allowing me to complete this project within the timeline. I also would like to thank my supervisor Dr. Noraini Che Pa for her guidance, motivation and patience during my research. Your good deeds will always be in my thoughts and prayers.

My appreciation goes to UPM lecturers, facilitators and staffs who have been involved directly or indirectly in this project. Besides that I would like to thank my colleagues and my family for endless support in completing this project. Last but not least, I am indebted to Public Service Department (JPA) and Government of Malaysia for awarding me scholarship to further postgraduate study (master level) in FSKTM, UPM.

DECLARATION

I hereby confirm that:

- This thesis is my original work.
- Quotations, illustrations and citations have been duly referenced.
- This thesis has not been submitted previously or concurrently for any other degree at any other institutions.
- Intellectual property from the thesis and copyright of thesis are fully-owned by University Putra Malaysia, as according to the University Putra Malaysia (Research) Rules 2012; written permission must be obtained from supervisor and the office of Deputy
- Vice-Chancellor (Research and Innovation) before thesis is published (in the form of written, printed or in electronic form) including books, journals, modules, proceedings, popular writings, seminar papers, manuscripts, posters, reports, lecture notes, learning modules or any other materials as stated in the University Putra Malaysia (Research) Rules 2012.
- There is no plagiarism or data falsification/fabrication in the thesis, and scholarly integrity is upheld as according to the University Putra Malaysia
- (Graduate Studies) Rules 2003 (Revision 2012-2013) and the University Putra
- Malaysia (Research) Rules 2012. The thesis has undergone plagiarism detection software.

Signature: -----

Date:

Name and Matric No: WAN NURSYAMSI AH WAN JUSOH, GS42092

Signature: -----

DR NORAINI CHE PA (Supervisor)

TABLE OF CONTENTS

	Page
DEDICATION	ii
ABSTRACT	iii
ABSTRAK	iv
ACKNOWLEDGEMENT	v
DECLARATION	vi
TABLE OF CONTENTS	vii
LIST OF FIGURES	x
LIST OF TABLES	xii
CHAPTER 1	1
INTRODUCTION	1
1.1 Problem Statement	2
1.2 Objectives	3
1.3 Scope of the Research	3
1.4 Contribution	4
1.5 Research Structure	4
CHAPTER 2	5
LITERATURE REVIEW	5
2.0 Introduction	5
2.1 Requirement Engineering	5
2.1.1 Functional Requirements	6
2.1.2 Non-Functional Requirements	7
2.1.3 Elicitation and Analysis Process	10
2.2 Requirements Reuse	11
2.2.1 Requirements Reuse Methods and Models	12
2.2.2 Software Requirements Catalog	16
2.3 Requirements Standard	19
2.4 Software Agent	21
2.5 Conclusion	22
CHAPTER 3	23
RESEARCH METHODOLOGY	23
3.0 Introduction	23
3.1 Theoretical Study	23
3.2 Propose a Model	24

3.3	Build Prototype	25
3.4	Model Evaluation	25
3.5	Summary	26
CHAPTER 4		27
IMPEMENTATION AND EVALUATION		27
4.0	Introduction	27
4.1	Conceptual Model	27
4.2	Prototype Design	30
4.2.1	Flowchart	31
4.2.2	Use Case	32
4.2.3	Architecture	34
4.2.4	Interface Design	36
4.2.4.1	Main page	36
4.2.1.1	Manage Requirements Catalog	38
4.2.1.2	Manage Project	41
4.2.2	Database Design	47
4.2.2.1	Table structure	48
4.2.3	Agent Algorithm	54
4.2.3.1	Suggest NFR	54
4.2.3.2	Enable/Disable NFR	57
4.2.4	Prototype Integration and Testing	57
4.3	Evaluation	58
4.3.1	Case Study	58
4.3.2	Data Preparation	58
4.3.3	Post Case Study	59
4.4	Summary	59
CHAPTER 5		60
RESULT AND DISCUSSION		60
5.0	Introduction	60
5.1	Model and Prototype Evaluation	60
5.1.1	Demographic Information	61
5.1.2	Quality Attributes	63
5.1.2.1	Completeness	64
5.1.2.2	Consistent	65
5.1.2.3	Traceable	66
5.1.3	General questions	68

5.2	Summary	69
CHAPTER 6		70
CONCLUSION AND FUTURE WORKS		70
6.0	Introduction	70
6.1	Conclusion	71
6.2	Future work	72
REFERENCES		73
APPENDICES		78



LIST OF FIGURES

Figure 1: Method to Determine Quality Attributes.....	7
Figure 2: Elicitation Techniques	10
Figure 3: Difference between SRC and SPL	16
Figure 4: SIREN Approach.....	17
Figure 5: Creation Model of RRMSRC	18
Figure 6: Requirement Management Activity in RRMSRC	19
Figure 7: Phases in Research Methodology	23
Figure 8: Agent Based Requirements Reuse Model Catalog (ABRRMC).....	29
Figure 9: Prototype Flow Chart.....	31
Figure 10: Use Case for ABRRMC Prototype.....	33
Figure 11: ABRRMC Architecture	34
Figure 12: User Directories of ABRRMC	37
Figure 13: Main Page of ABRRMC	37
Figure 14: Manage Domain Interface	38
Figure 15: Add Functional Requirements	39
Figure 16: Parameterization and Dependency Setting	40
Figure 17: Update Functional Requirements	40
Figure 18: NFR Maintenance Page	41
Figure 19: Navigation Sequence for a Project	41
Figure 20: Add New Project	42
Figure 21: Example of New Project Creation	42
Figure 22: Interface to Select Functional Requirements.....	43
Figure 23: Requirement Information	43
Figure 24: Set Parameter.....	44

Figure 25: NFR Suggested by the System	44
Figure 26: Validation Step	45
Figure 27: Interface to View SRS	46
Figure 28: Menu for Existing Project	46
Figure 29: ERD for ABRPMC Prototype.....	47
Figure 30: Age Information of the Participants	61
Figure 31: Working Experience Related to Requirement Engineering Field	62
Figure 32: Participant's Education Level.....	62



LIST OF TABLES

Table 1: Software Quality Models	8
Table 2: Requirements Reuse Methods and Models.....	12
Table 3: SRS Hierarchy Based on IEEE Std 29148.....	20
Table 4: Pages implementation for ABRRMC prototype	35
Table 5: Table Domain Master	48
Table 6: Table Functional Requirements Master	49
Table 7: Table Parameter Master	50
Table 8: Table Dependency Master	50
Table 9: Table NFR Master.....	51
Table 10: Table Project	51
Table 11: Table Project Functional Requirements.....	52
Table 12: Table Project NFR	52
Table 13: Table tempNFR.....	53
Table 14: Table parameterProject	54
Table 15: NFR and Related Keywords	55
Table 16: Overall Quality Attributes Result	63
Table 17: Result of Completeness Attribute from Control Group.....	64
Table 18: Result of Completeness Attribute from Experimental Group.....	64
Table 19: Percentage Result of Completeness Attribute.....	65
Table 20: Result of Consistent Attribute from Control Group	65
Table 21: Result of Consistent Attribute from Experimental Group	66
Table 22: Percentage Result of Consistent Attribute	66
Table 23: Result of Traceable Attribute for Control Group.....	67
Table 24: Result of Traceable Attribute for Experimental Group	67

Table 25: Percentage Result of Traceable Attribute 67

Table 26: Result on General Questions..... 68



© COPYRIGHT UPM

CHAPTER 1

INTRODUCTION

Requirements engineering (RE) is “*the subset of systems engineering concerned with discovering, developing, tracing, analyzing, qualifying, communicating and managing requirements that define the system at successive levels of abstraction*” (Hull et. al., 2010). Ghaisas and Ajmeri (2013) mentioned that the aim of RE is to convert from uncertain and ambiguous problem into features and attributes of software system. During this process, it is crucial to understand the services required and the constraints involved as any misunderstanding at this stage will lead to later problem in system design and implementation (Sommerville, 2010). Thus it is a vital step to ensure other stages follow the right direction and the output is correctly delivered to the customer. In order to gather requirements, there are many techniques can be applied including interviews, questionnaires, prototyping, observation and document analysis. Requirements specification is the output of RE that is used as a basis contract between stakeholders and system developers (Bubenko et. al, 1994; Villegas and Laguna, 2001). Besides that, it serves as an instrument to validate system at the later stage.

Requirement stages involve a great deal of works from requirement elicitation to requirement documentation (Benitti and da Silva, 2013), thus requirements reuse become an alternative to speed up the process. Requirements reuse is an approach to

use existing requirements documents to reduce the general effort inside the software life cycle (Villegas and Laguna, 2001). There are several activities to be done during the process and requires requirement repository to keep as well as to reuse the artifacts. The advantages of requirements reuse is to improve quality and productivity (Pacheco et. al, 2015), accelerate time to market and cut development costs (Villegas and Laguna, 2001). These advantages lead to chances of project success and efficiency in requirement elicitation process (Franch et. al., 2013). Previous research work in requirements reuse include real-time systems, software product lines (SPL), ontologies, metamodeling and software requirements catalog (Benitti and da Silva, 2013).

1.1 Problem Statement

Requirements reuse is widely applied in software product lines (SPL) because of features similarity. SPL is considered as a large-sized software enterprise that having complex and difficult requirements. For small size enterprises, SPL approach is not suitable in applying requirements reuse due to different context, properties and complexity. Pacheco et. al. (2016) introduced a model for small scale software development called requirements reuse model for software requirements catalog (RRMSRC). The catalogs are categorized based on similar domain for easy access and utilization. In this model, it covers three phases of requirements reuse during creation, maintenance and management. However this model only cater functional requirements thus it lacks non-functional requirements. In order to have a well-defined and mature reuse processes, non-functional requirements must be included. As affirmed by Franch et. al. (2013), scope of SRS regularly covers functional, non-functional and non-

technical requirements, and non-functional is often reused compared to functional requirements.

1.2 Objectives

Objective 1:

To study on existing models and requirements reuse process.

Objective 2:

To improve RRMSRC requirements reuse model by including non-functional requirements.

Objective 3:

To develop and evaluate the improved model through a prototype.

1.3 Scope of the Research

This study focus on requirements reuse activities during requirement elicitation and analysis phase which includes functional and non-functional requirements that performed by software engineers. The agent technology is used in developing the prototype to facilitate the process.

1.4 Contribution

The new model proposed is expected to accelerate requirements reuse process and improved requirements reuse quality attributes. This research also highlight the importance of non-functional requirements during the reuse.

1.5 Research Structure

This study comprises six (6) chapters, including this chapter which covers the backgrounds of the study, problem statement, objectives, research questions, scope of the research, contribution and research structure. Chapter 2 present a literature review by covering existing study, requirements reuse concept, method and models. Software agent technology and the latest requirements standard (IEEE Std 29148 – 2011) are also discussed in this chapter. Chapter 3 describes the methodology which include theoretical study, study existing model, propose a new model, build prototype and model evaluation. Proper planning to carry out the project is important to reduce unforeseen problem in the future. Meanwhile in Chapter 4, the execution of the model and prototype development will be covered. Followed by result and discussion which will be elaborated in Chapter 5. Finally, the last chapter, Chapter 6 summarizes the thesis finding and work that can be done in future.

REFERENCES

- Bakar, N. H., Kasirun, Z. M., and Salleh, N. (2015, December). Terms Extractions: An Approach for Requirements Reuse. In Information Science and Security (ICISS), 2015 2nd International Conference on (pp. 1-4). IEEE.
- Benitti, F. B. V., and da Silva, R. C. (2013). Evaluation of a Systematic Approach to Requirements Reuse. *J. UCS*, 19(2): 254-280.
- Bubenko, J., Rolland, C., Loucopoulos, P., and DeAntonellis, V. (1994, April). Facilitating “fuzzy to formal” requirements modelling. In Requirements Engineering, 1994., Proceedings of the First International Conference on (pp. 154-157). IEEE.
- Chung, L. (2006, September). Capturing and reusing functional and non-functional requirements knowledge: a goal-object pattern approach. In 2006 IEEE International Conference on Information Reuse & Integration (pp. 539-544). IEEE.
- Cleland-Huang, J., Settimi, R., Zou, X., & Solc, P. (2006, September). The detection and classification of non-functional requirements with application to early aspects. In 14th IEEE International Requirements Engineering Conference (RE'06) (pp. 39-48). IEEE.
- Cybulski, J. L. (1996). Introduction to software reuse. Department of Information Systems, The University of Melbourne, Parkville, Australia.
- Cybulski, J. L., and Reed, K. (2000). Requirements classification and reuse: crossing domain boundaries. In *Software Reuse: Advances in Software Reusability* (pp. 190-210). Springer Berlin Heidelberg.

- Daud, N., Kamalrudin, M., Sidek, S., & Ahmad, S. S. S. (2014). A Review of Requirements Engineering Tools for Requirements Validation. *International Journal of Software Engineering and Technology*, 1(1).
- de Gea, J. M. C., Nicolás, J., Alemán, J. L. F., Toval, A., Vizcaíno, A., & Ebert, C. (2013). Reusing requirements in global software engineering. In *Managing requirements knowledge* (pp. 171-197). Springer Berlin Heidelberg.
- Dehlinger, J., & Lutz, R. R. (2005, May). A product-line requirements approach to safe reuse in multi-agent systems. In *ACM SIGSOFT Software Engineering Notes* (Vol. 30, No. 4, pp. 1-7). ACM.
- Finkelstein, A. (1988). Re-use of formatted requirements specifications. *Software Engineering Journal*, 3(5), 186-197.
- Franch, X., Palomares, C., Quer, C., Renault, S., and De Lazzer, F. (2010). A metamodel for software requirement patterns. In *Requirements Engineering: Foundation for Software Quality* (pp. 85-90). Springer Berlin Heidelberg.
- Franch, X., Quer, C., Renault, S., Guerlain, C., and Palomares, C. (2013). Constructing and using software requirement patterns. In *Managing requirements knowledge* (pp. 95-116). Springer Berlin Heidelberg.
- Ghaisas, S., & Ajmeri, N. (2013). Knowledge-assisted ontology-based requirements evolution. In *Managing requirements knowledge* (pp. 143-167). Springer Berlin Heidelberg.
- Goldin, L., and Berry, D. M. (2015). Reuse of requirements reduced time to market at one industrial shop: a case study. *Requirements Engineering*, 20(1), 23-44.

- Hauksdóttir, D., Mortensen, N. H., and Nielsen, P. E. (2014). Identified adjustability dimensions when generating a product specific requirements specification by requirements reuse. *Computers in Industry*, 65(6), 952-966.
- Hauksdottir, D., Vermehren, A., and Savolainen, J. (2012, September). Requirements reuse at Danfoss. In *Requirements Engineering Conference (RE), 2012 20th IEEE International* (pp. 309-314). IEEE.
- Hull, E., Jackson, K., & Dick, J. (2010). *Requirements engineering*. Springer Science & Business Media.
- Ian Sommerville (2010). *Software Engineering* (9th ed.). Addison-Wesley Publishing Company, USA.
- ISO, IEC. IEEE. 29148: 2011-Systems and software engineering-Requirements engineering.
- Jennings, N. R., and Wooldridge, M. J. (1996). Software agents. *IEE review*, 17-20.
- Kazman, R. (2014). *Software Architecture in Practice*. Addison-Wesley.
- Mairiza, D., Zowghi, D., & Nurmuliani, N. (2010, March). An investigation into the notion of non-functional requirements. In *Proceedings of the 2010 ACM Symposium on Applied Computing* (pp. 311-317). ACM.
- Mussbacher, G. (1998). *Combining Case Based Reasoning and Commonality Analysis for Software Requirements Reuse* (Doctoral dissertation, Simon Fraser University).
- Niu, N., Savolainen, J., Niu, Z., Jin, M., & Cheng, J. R. C. (2014). A systems approach to product line requirements reuse. *IEEE Systems Journal*, 8(3), 827-836.

- Nuseibeh, B. & Easterbrook, S., 2000. Requirements engineering: a roadmap C. Ghezzi, M. Jazayeri, & A. L. Wolf, eds. *Context*, 1(258), p.35-46.
- Nwana, H. S. (1996). Software agents: An overview. *The knowledge engineering review*, 11(03), 205-244.
- Pacheco, C. L., Garcia, I. A., Calvo-Manzano, J. A., and Arcilla, M. (2015). A proposed model for reuse of software requirements in requirements catalog. *Journal of Software: Evolution and Process*, 27(1): 1-21.
- Pacheco, C., Garcia, I., Calvo-Manzano, J. A., and Arcilla, M. (2016). Reusing functional software requirements in small-sized software enterprises: a model oriented to the catalog of requirements. *Requirements Engineering*, 1-13.
- Palomares, C., Franch, X., and Quer, C. (2014). Requirements reuse and patterns: a survey. In *Requirements Engineering: Foundation for Software Quality* (pp. 301-308). Springer International Publishing.
- Perednikas, E. (2008). Requirements Reuse Based on Forecast of User Needs. In *Proceedings of the 20th EURO Mini Conference on Continuous Optimization and Knowledge-Based Technologies*, Neringa, Lithuania (pp. 450-455).
- Radulovic, F., & Garcia-Castro, R. (2011). Extending Software Quality Models-A Sample In The Domain of Semantic Technologies. In *SEKE* (pp. 25-30).
- Rehman, T., Khan, M. N. A., & Riaz, N. (2013). Analysis of requirement engineering processes, tools/techniques and methodologies. *International Journal of Information Technology and Computer Science (IJITCS)*, 5(3), 40.
- Slankas, J., & Williams, L. (2013, May). Automated extraction of non-functional requirements in available documentation. In *Natural Language Analysis in*

Software Engineering (NaturaLiSE), 2013 1st International Workshop on (pp. 9-16). IEEE.

Stellman, A., & Greene, J. (2005). Applied software project management. " O'Reilly Media, Inc."

Toval, A., Morós, B., & Vicente Chicote, C. (2008). Metamodeling variability to enable requirements reuse.

Toval, A., Morós, B., and Vicente Chicote, C. (2008). Metamodeling variability to enable requirements reuse.

Toval, A., Nicolás, J., Moros, B., and García, F. (2002). Requirements reuse for improving information systems security: a practitioner's approach. *Requirements Engineering*, 6(4): 205-219.

Villegas, O. L., and Laguna, M. A. (2001). Requirements reuse for software development. In RE 01 Doctoral Workshop. in 5th-IEEE International Symposium on Requirements Engineering. Toronto, Canada (pp. 27-31).

Wooldridge, M. (2009). An introduction to multiagent systems. John Wiley and Sons.