UNIVERSITI PUTRA MALAYSIA

QUALITY ENHANCEMENT OF SOFTWARE MAINTAINABILITY EVALUATION MODEL VIA CAPABILITY MATURITY MODEL INTEGRATION

AL-AHMAD HANEEN HASSAN

FSKTM 2015 14

QUALITY ENHANCEMENT OF SOFTWARE MAINTAINABILITY EVALUATION MODEL VIA CAPABILITY MATURITY MODEL INTEGRATION

By

AL-AHMAD HANEEN HASSAN

Thesis Submitted to the School of Graduate Studies, Universiti Putra Malaysia, in Fulfillment of the Requirements for the Degree of Doctor of Philosophy

June 2015

Abstract of the thesis presented to the Senate of Universiti Putra Malaysia in fulfillment of the requirement for the degree of Doctor of Philosophy

QUALITY ENHANCEMENT OF SOFTWARE MAINTAINABILITY EVALUATION MODEL VIA CMMI PROCESS

By

AL-AHMAD HANEEN HASSAN

June 2015

Chairman : Rodziah Atan, Ph.D.
Faculty : Computer Science and Information Technology

The enhancement of software maintenance process is one of the most rapidly growing concerns for many reasons such as successful delivery of projects and organization management. Software maintenance companies are reluctant to implement process improvement models and procedures because of their complex construction and challenging implementation techniques. It has been observed that the enhancement efforts are based on process development frameworks which are considered normally for large organizations. The Capability Maturity Model Integration (CMMI) enables companies and organizations to enhance presentation and rate the maturity of their level of process. This thesis focuses on classifying the significant process areas and components for software maintenance improvement and provides best performance observation for the enhancement process of which that can be applied in small organizations. The main objective of this study is to establish a new predictive model by reducing the CMMI level maintenance process integrated with agent tools. It also aimed to improve the existing model of CMMI for multi agent system (MAS) in the collaborative software maintenance environment. This thesis formulated its research objectives through relevant literature and organized reviews of CMMI and Software Performance Indicator (SPI). The study was developed based on the CMMI process reports. The investigation of the study was divided into 4 phases based on objective directions to obtain the results. The new mapping of maturity level process areas and problems is completed by analyzing CMMI process and specific practices. This research has obtained a significant finding: the establishment of a new predictive model which reduces the CMMI level maintenance process, integrated with agent tools for process enhancement. The conclusions of this study measured the performance of the improved CMMI maintenance process and defined that the existing CMMI methods for advance process just provided the controlling principles to succeed the maturity of the software maintenance process. Finally, the new integrated model based on the proposed components with software maintenance indicates a high reliability data of 0.82 and Cronbach alpha of 0.94 as an output of questionnaire design according to proposed modification.

Abstrak tesis yang dikemukakan kepada Senat Universiti Putra Malaysia sebagai memenuhi keperluan untuk ijazah Doktor Falsafah

PENINGKATAN KUALITI MODEL PENILAIAN PENYELENGGARAAN PERISIAN MELALUI PROSES CMMI

Oleh

AL-AHMAD HANEEN HASSAN

Jun 2015

Pengerusi : Rodziah Atan, PhD
Fakulti : Sains Komputer dan Teknologi Maklumat

Peningkatan proses penyelenggaraan perisian adalah salah satu kebimbangan yang meningkat naik disebabkan oleh banyak perkara seperti kejayaan penghantaran projek dan pengurusan organisasi. Syarikat penyelenggaraan perisian enggan melaksanakan model penambahbaikan proses dan prosedur kerana kaedah implementasi yang kompleks dan teknik pelaksanaan model yang mencabar. Berdasarkan pemerhatian usaha-usaha peningkatan adalah berasaskan kepada rangka kerja pembangunan proses yang dianggap biasa bagi organisasi besar. Model Capability Maturity Model Integration (CMMI) membolehkan syarikat dan organisasi meningkatkan penampilan dan tahap kadar kematangan proses. Fokus tesis ini adalah untuk mengklasifikasikan proses penting dan komponen bagi penambahbaikan proses penyelenggaraan perisian dan menyediakan kaedah terbaik pemantauan prestasi untuk peningkatan proses yang mampu diaplikasi dalam organisasi kecil. Objektif utama kajian ini adalah untuk mewujudkan satu model ramalan baharu dengan mengurangkan proses penyelenggaraan tahap CMMI bersepadu bersama dengan peralatan ejen. Ia juga mensasarkan untuk meningkatkan model sediaada CMMI dengan sistem ejen berbilang (MAS) dalam persekitaran penyelenggaraan perisian kolaboratif. Objektif tesis ini dirumuskan melalui kajian literatur yang berkaitan dan ulasan diunjurkan kepada CMMI dan bidang proses prestasi (KPA). Kajian ini telah dibangunkan berdasarkan laporan proses CMMI. Siasatan kajian telah dibahagikan kepada empat fasa berdasarkan arah objektif untuk mendapatkan keputusan. Penyesuaian baru tahap kematangan proses dan masalah diselesaikan dengan menganalisis proses CMMI serta kaedah amalan yang dilaksanakan. Kajian ini telah menghasilkan satu dapatan penting:
iatu pembangunan sebuah model baharu yang dapat mengurangkan tahap proses penyelenggaraan bersepadu CMMI menggunakan peralatan ejen yang dicadangkan bagi peningkatan proses penyelenggaraan. Kesimpulan bagi kajian ini ditakrifkan sebagai mengukur keber kesanan proses penyelenggaraan CMMI untut bertambah baik dan kaedah CMMI yang sedia ada untuk proses awalan bagi menyediakan prinsip-prinsip untuk berjaya mengawal kematangan proses. Akhir sekali, model bersepadu baharu berasaskan komponen penyelenggaraan perisian berkolaborasi yang dicadangkan menunjukkan kebolehpercayaan data yang tinggi sebanyak 0.82 dan Cronbach alpha sebanyak 0.94 dari output reka bentuk soal selidik validasi mengikut cadangan pengubahsuaian.

AKNOWLEDGEMENTS

For me, the acknowledgment section is a particularly enjoyable part of my thesis to write. It allows me to recall the many people who have helped me out over the years. First of all I would like to express my sincere gratitude to my main supervisor Assoc. Prof. Dr. Rodziah binti Atan who guided me to the end of this PhD journey. I deeply appreciate her enthusiasm and her excellent knowledge of the research area. Especially, I would like to thank her for being a sparring partner in valuable and inspiring discussions on the topic of my research, for being always available when I needed her input, and for providing such inputs in a very thorough way. Furthermore, I would like to give a special word of thanks to my committee members Professor Dr. Abdul Azim Abd. Ghani and Assoc. Prof. Dr. Masrah Azrifah binti Azmi Murad for their support.. The friendliness, openness and honor of my supervisors as well as colleagues have enabled me to enjoy my stay at UPM which I consider as the best part of the period of my doctoral research.

I would like to thank everyone who contributed directly or indirectly to my research. I specially thank Universiti Putra Malaysia (UPM) for its valuable support. Incidentally, I am not forgetting to thank the faculty members for sharing interesting and fruitful discussions with me, providing detailed information on relevant research subjects, giving high quality research combinations, handling administrative issues and dealing with obstacles during the past years.

I also have a special word of thanks for my family for providing substantial support for my PhD research. Without their help and commitment I could not have finished this thesis.

I certify that a Thesis Examination Committee has met on 26 June 2015 to conduct the final examination of Haneen Hassan Al-Ahmad on her thesis entitled "Quality Enhancement of Software Maintainability Evaluation Model via Capability Maturity Model Integration" in accordance with the Universities and University Colleges Act 1971 and the Constitution of the Universiti Putra Malaysia [P.U.(A) 106] 15 March 1998. The Committee recommends that the student be awarded the Doctor of Philosophy.

Members of the Thesis Examination Committee were as follows:

Azmi bin Jaafar, PhD
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Chairman)

Rusli bin Hj Abdullah, PhD
Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Internal Examiner)

Abu Bakar bin Md Sultan, PhD
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Internal Examiner)

Shehab A Gamalel-Din, PhD
Professor
King Abd Aziz University
Saudi Arabia
(External Examiner)

ZULKARNAIN ZAINAL, PhD
Professor and Deputy Dean
School of Graduate Studies
Universiti Putra Malaysia

Date: 22 September 2015

This thesis was submitted to the Senate of Universiti Putra Malaysia and has been accepted as fulfillment of the requirement for the degree of Doctor of Philosophy. The members of the Supervisory Committee were as follows:

Rodziah Atan, PhD.
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Chairman)

Masrah Azrifah Azmi Murad, PhD.
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Member)

Abdul Azim Abd. Ghani, PhD.
Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Member)

BUJANG BIN KIM HUAT, PhD.
Professor and Dean
School of Graduate Studies
Universiti Putra Malaysia

Date:

Declaration by graduate student

I hereby confirm that:

this thesis is my original work

quotations, illustrations and citations have been duly referenced

the thesis has not been submitted previously or comcurrently for any other degree at any institutions

intellectual property from the thesis and copyright of thesis are fully-owned by Universiti Putra Malaysia, as according to the Universiti Putra Malaysia (Research) Rules 2012;

written permission must be owned from supervisor and 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 Universiti 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 Universiti Putra Malaysia (Graduate Studies) Rules 2003 (Revision 2012-2013) and the Universiti Putra Malaysia (Research) Rules 2012. The thesis has undergone plagiarism detection software

Signature: _____ Date:_____

Name and Matric No: Al-Ahmad Haneen GS 26809

Declaration by Members of Supervisory Committee

This is to confirm that:

the research conducted and the writing of this thesis was under our supervision;
supervision responsibilities as stated in the Universiti Putra Malaysia (Graduate Studies) Rules 2003 (Revision 2012-2013) were adhered to.

Signature: _____	Signature: _____
Name of Chairman of Supervisory Committee: Rodziah Atan, PhD.	Name of Member of Supervisory Committee: Abdul Azim Abd. Ghani, PhD.

Signature: _____
Name of Member of Supervisory Committee: Masrah Azrifa Azmi, PhD.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CDLC	Code Development Line Carrier
CMMI	Capability Maturity Model Integration
CMU	Carnegie Mellon University
FPA	Function Point Analysis
IEEE	Institute of Electronical and Electronics Engineers
IT Service CMM	IT Service Capability Maturity Model
KPA	Key Process Area
KPAA	Key Process Area Agent
MA	Maintenance Agent
MAS	Multi Agent System
ML	Maturity Level
MTA	Maintenance Type Agent
PA	Personal Agent
PM	Project Mangement
PSB	Perceived System Benefits
RMM	Rasch Measurement Model
PIDM	Person Item Distribution Map
SCM	Software Configuration Mangement
SEI	Software Engineering Institute
SMP	Software Maintenance Process
SQA	Software Quality Assurance
SR	Service Request
SRS	Software Requerment Spesification
SPI	Software Process Improvement
S3M	Software Maintenance Maturity Model
ROI	Return on investment
VCS	Version Control System

LIST OF SYMBOLS

e	base of nature log
Q	3HUVRQ¶V DELOLW\
/M	,WHP¶V GIficulty
3	Probability of success

CHAPTER 1

INTRODUCTION

1.0 Introduction

Software Maintenance (SM) is of great importance to organizations mainly because organizations depend on how quickly and reliably the software can be changed to meet the ever changing business environment (Bennet & Rajlich, 2000). In order to survive the stiff competition of modern business, software systems and their respective maintenance activates should be incorporated with a current enhanced method for better execution. In recent years, there has been a trend for the software industry to improve system development processes following methodologies, best practices and standards recognized in the market. Different methods exist which are used to manage software change such as software maintenance, architectural alteration and software engineering (April & Abran, 2012).

Software maintenance in software engineering is the modification of a software product after delivery to correct faults, to improve performance or other attributes including non-functional software features. The efforts and expenses involved in maintaining software are considerable and software organizations are demanding greater effectiveness in the process of maintenance. This study proposes a viable and effective process improvement model by describing the theoretical framework that is applied in actual and current organization. The study identifies the problem to be solved and proposed solution for maintenance activities to be improved within key process area covered in popular software standard used by many organizations - the Capability Maturity Model Integration (CMMI) developed by the (SEI) at Carnegie Mellon University (CMU) mainly covered three software areas which are development, acquisition and services.

CMMI consists of KPAs that need to be fully considered by companies applying it. CMMI models provide guidance for developing or improving processes that meet the business goals of an organization. A CMMI model may also be used as a framework for appraising the process maturity of the organization. The practices of software maintenance method studies in this thesis come from a good demonstration, experience-based software maintenance activities.

The focus of this research is the investigation and analysis of software maintenance process and models utilized in organizations as portrayed in CMMI KPAs. The study observed and identified the best approaches needed to improve the performance of software maintenance process. The study analyzed critically important key process areas as explained by CMMI, needed in software maintenance activities. The study classifies the challenges in software maintenance and its associated improvement based on CMMI; design an appropriate model for third party development using the categories of maintenance KPAs. Specific goals and practices to be proposed in the model will be the basis for software development activities which focused on maintenance phase from every CMMI level. All the component actions needed for

software process maintenance process improvement will be considered in this study. Apart of proposing an improvement model to software maintenance activities based on CMMI, this research also translates the model into an architectural design in a multi-agent system (MAS) environment.

A multi-agent system (MAS) is a computerized system composed of multiple interacting intelligent agents within an environment. (Fitzgerald et al., 2007). In Durfee et al., (1989), MAS LV GHILQHG DV ³D ORRVHO\ FRXSOHG solver entities that work together to find answers to problems that are beyond the LQGLYLGXDO FDSDELOLWLHV RU NQRZOHGJH RI HDFK H

Agents are mostly viewed as independent units. These could be in the form of software programs or robots. Their communication can be either supportive or detached. This means that, agents are free to share information on a common ground or decide to independently follow their own interests. This factor that makes it important to form a general architecture design based on MAS for the proposed model. The architecture design can then be used multiple times by software maintainers for different maintenance issues.

Agent properties (Wooldridge & Jennings, 1995) are as follows:

> *Autonomy*: agents are independent and operate with obstruction from outside forces and also have control of their actions and internal state.
> *Social ability*: connection between agents is done through some kind of ACL.
> *Reactivity: agents can understand their environment giving them the ability to respond* promptly to changes occurring therein.
> *Pro-activeness*: More-so, agents can take initiative towards goal achievement.

The robust opinion of agent is the lee way of the weaker idea and supports more on humanistic, mental properties such as belief, desire, and intention of operations (Shoham, 1993).

This study is motivated by the actual scenario happened in organizations with weak management of maintenance process and activities. Without proper maintenance model to follow, these organizations would not be able to conduct maintenance activities or answer to request change quickly or within stipulated time.

1.1 Problem Statement

This study is conducted based on problems in the area of software maintenance (SM) as indicated by other researchers. The three problems are listed, by their respective researches**:**

> It is hard to organize the important SM key process area if the classification of information was not carried out as to highlight the continuous improvement process as proposed in CMMI (CMMI Product, 2006).

The measurements to change request performance indicators defined for SM processes and activities does not accurately reflect to the process maturity and the task difficulty (Pesic, 2009).

Measuring the whole CMMI software maintenance KPAs is time consuming whereby it is affecting the effectiveness of software maintenance activities (SEI, 2006).

1.2 Research Questions

The following research questions need to be addressed in conjunction with the above stated problems.

i. What is the effectiveness of CMMI SM process adapted in five levels of CMMI?
ii. How can CMMI software maintenance process be improved?
iii. How can an agent tool support the maintenance process?

1.3 Research Objectives

There are four main objectives formulated for this research. They are listed as follows:

i. To characterize the typical processes of SM and its activities.
ii. To formulate a model of CMMI based on Multi agent system design in collaborative SM environment.
iii. To design an agent-based architecture that support the proposed mode
iv. To analyze the model effectiveness.

1.4 Contribution of the Research

This research contributes to the software management model for CMMI maintenance process, through the improvement of the CMMI maintenance plan. The deliberated objectives addressed in this study have been achieved following suit the stated goals by obtaining the characteristics of the software environment, establishing proposed enhanced model based on CMMI level, reducing the process layers and incorporate agent tool design to improve the planning.

This study undertakes steps that are presented by initial measurement of questionnaires based on actual environment. The measurement carried out shown to be capable of finding reasonable condition to reduce the CMMI software maintenance activities compared to current CMMI practices. Effective set of questionnaire are distributed and CMMI model for maintenance was formulated accordingly. The proposed model demonstrates inventive steps for maintenance process plan to be considered in software management and maintenance.

This study is significant for several motivations, which are in the sense of identify and reporting problems by third party software process developers. The study highlights the main components and factors for the software process improvement and provides CMMI-based solutions. It categorizes the critical process aspects of software maintenance using the proposed model approach in these key areas of process. It also interprets CMMI-based model for third party software developers ¶ usages that can assist organizations to attain improved maintenance outcomes and manage quality software products. These products eventually will run in minimized effort, time and cost of maintenance. It also helps organizations to improvement their model for process maintenance based on concrete implementation approach.

1.5 Thesis Organization

Chapter 1 provides the general overview of the theory and application, problem statement, research objectives, research questions, a list of contributions and thesis organization associated with this thesis. Capability maturity model integration for multi agent software system maintenance was studied and its importance is discussed. Measured data from questionnaire are analyzed in order to gain knowledge about the typical processes of software management conducts. The relations of these characteristics with software parameters and questionnaire data are studied, using statistical properties. The improvement offered by the proposed models is also listed in this chapter.

Chapter 2 describes the different methods used in software maintenance that are currently available, which could help the research in modeling the software maintenance management. Chapter 3 covers descriptions of issues related to the modeling of the CMMI maintenance process, the MAS architecture framework for the proposed model that would be used in software maintenance, its measurement process and some analysis properties are also described in detail. Chapter 4 describes in detail the questions used as part of model definition, formation and application to the improved SM and questionnaires for model validation.

The system design and proposed maintenance architecture design using MAS is described in Chapter 5. Chapter 6 displays the results of the proposed model and its analysis of evaluation. Chapter 7 ends the thesis with conclusions and recommendations for future work.

REFERENCES

Abran, A. Moore, J. Bourque, W. Dupuis, P., & Tripp, L. (2004). Guide for the Software Engineering Body of Knowledge (SWEBOK), Ironman version, *IEEE Computer Society Press*: Los Alamitos CA, pp.6-15.

Anacleto, A., von Wangenheim, C. G., Salviano, C. F., & Savi, R. (2004). Experiences gained from applying ISO/IEC 15504 to small software companies in Brazil. In *4th International SPICE Conference on Process Assessment and Improvement, Lisbon, Portugal*, pp. 33-37.

Anda, B. (2007). Assessing software system maintainability using structural measures and expert assessments. In *Software Maintenance, 2007. ICSM 2007. IEEE International Conference on Software Maintenance*, pp. 204-213.

Anderson, D. J. (2003). Agile Management for Software Engineering, Applying the Theory and Constraints for Business Results. Prentice Hall, pp. 336.

Anderson, D. J. (2005). Stretching agile to fit CMMI level 3: The story of creating MSF for CMMI process improvement at Microsoft Corporation, *Paper presented at the AGILE Conference 2005*, Denver, CO.

Andrich, D. (1978). Application of a psychometric rating model to ordered categories which are scored with successive integers. *Applied psychological Measurement*, *2*(4), 581-594.

Anquetil, N., de Oliveira, K. M., de Sousa, K. D., & Dias, M. G. B. (2007). Software maintenance seen as a knowledge management issue. *Information and Software Technology*, *49*(5), 515-529.

April, A., & Abran, A. (2012). *Software maintenance management: evaluation and continuous improvement* . John Wiley & Sons.

April, A., & Abran, A. (2009). A software maintenance maturity model (S3M): Measurement practices at maturity levels 3 and 4. *Electronic Notes in Theoretical Computer Science*, *233*, 73-87.

April, A., Hayes, J. Huffman, Abran, A., & Dumke, R.(2005). Software Maintenance Maturity Model (SMmm): The software maintenance process model, J*ournal of Software Maintenance and Evolution: Research and Practice*, 2005, pp. 197-223.

April, A., Desharnais, J.M., & Dumke, R. (2006). A Formalism of ontology to Support a software maintenance knowledge-based system, *In 18th International Conference on Software Engineering and Knowledge Engineering (SEKE 2006), San Francisco, California*, July 2006, pp. 331-336

Baghaei, P. (2008). The Rasch model as a construct validation tool. *Rasch Measurement Transactions*, *2.02*(1), 1145-1146.

Baker, S. W. (2005). Formalizing agility: an agile organization's journey toward CMMI accreditation. In *Agile Conference, 2005. Proceedings,* pp. 185-192.

Baker, S. W. (2006). Formalizing agility, part 2: How an agile organization embraced the CMMI. In *Agile Conference*, 2006, pp. 8-34.

Beck, K. (2000). *Extreme Programming Explained: Embrace Change*. Addison-Wesley Longman.

Bennett, K. H., & Rajlich, V. T. (2000). Software maintenance and evolution: a roadmap. In *Proceedings of the Conference on the Future of Software Engineering,* pp. 73-87.

Boehm, B. (2003). Value-Based Software Engineering. *Computer*, pp. 33 ±41.

Boehm, B., & Turner, R. (2003a). *Balancing agility and discipline: A guide for the perplexed*. Addison-Wesley Professional.

Boehm, B., & Turner, R. (2003b). Using risk to balance agile and plan-driven methods. *Computer*, (6), 57-66.

Boehm, B., & Turner, R. (2005). Management challenges to implementing agile processes in traditional development organizations. *Software, IEEE*, *22*(5), 30-39.

Bond, T. (2004). Validity and assessment: a Rasch measurement perspective *Metodologia de las Ciencias del Comportamiento*, pp. 179-194.

Bond, T. G., & Fox, C. M. (2001). *Applying the Rasch model: Fundamental measurement in the human sciences*.Wiley.

Bond, T. G., & Fox, C. M. (2007). *Fundamental measurement in the human sciences.* Lawrence Erlbaum Associates.

Bond, T. G., & Fox, C. M. (2013). *Applying the Rasch model: Fundamental measurement in the human sciences*. Psychology Press.

Bos, E., & Vriens, C. (2004). An agile CMM. In *Extreme Programming and Agile Methods-XP/Agile Universe 2004,* pp. 129-138.

Brun, Y., Serugendo, G. D. M., Gacek, C., Giese, H., Kienle, H., Litoiu, M., & Shaw, M. (2009). Engineering self-adaptive systems through feedback loops. In *Software engineering for self-adaptive systems,* pp. 48-70.

Chrissis, M. B., Konrad, M., & Shrum, S. (2011). *CMMI for development: guidelines for process integration and product improvement*. Pearson Education.

CMMI Product. (2006). CMMI for Development, version 1.2. Capability Maturity Model® Integration for Development, Technical Software Engineering Institute. Report CMU/SEI-2006-TR-008.

CMMI Product. (2010). *CMMI for Acquisition Version 1.3*. Improving processes for developing better products and services.

CMMI Product. (2013). Maturity Profile Reports ±CMMI® for SCAMPI v1. 2/v1. 3± Class A Appraisal Results. *Software Engineering Institute, Carnegie Mellon University Pittsburgh*.

Cockburn, A., & Highsmith, J. (2001). Agile Software Development: *The People Factor. Computer,* pp. 131-133.

Cockburn, A. (2002). *Agile Software Development*. Boston: Addison-Wesley.

Cohen, D., Lindvall, M., & Costa, P. (2004). An introduction to agile methods. *Advances in Computers*, pp. 1-66.

Cooper, R. B., & Zmud, R. W. (1990). Information technology implementation research: a technological diffusion approach. *Management science*, *36*(2), 123-139.

Craig, R. J., & Amernic, J. H. (2006). PowerPoint presentation technology and the dynamics of teaching. *Innovative Higher Education*, *31*(3), 147-160.

Curtis, P., Phillips, D. M. & Weszka, J. (2002). CMMISM ² The evolution continues! *Systems Engineering*, *5*(1), 7-18.

Dekkers, C., & Emmons, B. (2002). How function points support the capability maturity model integration. *CrossTalk. The Journal of Defence Software Engineering*, pp. 21-24.

Deming, W. E. (1990). *Sample design in business research* . John Wiley & Sons.

Durfee, E. H., Lesser, V. R., & Corkill, D. D. (1989). Trends in cooperative distributed problem solving. *Knowledge and Data Engineering*, pp.63-83.

Eman, K. & Madhavji, N., (1999). Elements of Software Process Assessment & Improvement, *Computer Society*, IEEE

Fayad, M. E., & Laitnen, M. (1997). Process assessment considered wasteful. *Communications of the ACM*, *40*(11), 125-128.

Fichman, R. G. (2004). Going beyond the dominant paradigm for information technology innovation research: Emerging concepts and methods. *Journal of the association for information systems*, *5*(8), 11.

Fisher, C. (2007). *Researching and writing a dissertation a guidebook for business students*. Prentice Hall.

Fitzgerald, B., Hartnett, G., & Conboy, K. (2006). Customizing Agile Methods to Software Practices at Intel Shannon. *European Journal of Information Systems*, pp. 200 ±213.

Fowler, P., & Rifkin, S. (1990). *Software engineering process group guide* (No. CMU/SEI-90-TR-24). Carnegie-Mellon University Pittsburgh PA Software Engineering Inst.

Fritzsche, M., & Keil, P. (2007). Agile Methods and CMMI: Compatibility or Conflict? *Software Engineering Journal*, pp. 9 ±26.

Galin, D., & Avrahami, M. (2006). Are CMM Program Investment Beneficial? Analyzing Past Studies. *IEEE Software*, pp. 81 ±87.

Gallivan, M. J. (2001). Organizational adoption and assimilation of complex technological innovations: development and application of a new frame work. *ACM Sigmis Database*, *32*(3), 51-85.

Horvat, R. V., Rozman, I., & Györkös, J. (2006). Managing the Complexity of SPI in Small Companies. *Software Process Improvement and Practice*, pp. 45 ±54.

Huang, S. J., & Han, W. M. (2006). Selection priority of process areas based on CMMI continuous representation. *Information & Management*, *43*(3), 297-307.

Ilieva, G. (2011). Decision Making Methods in Agent Based Modeling. In *Proceedings of Workshop on Applications of Software Agents, Novi Sad, Serbia*, pp. 8-17.

ISO. (2004). *(SPICE) ISO TR 15504. Part 5. Information technology Software process assessment Part 5: An exemplar Process Assessment Model*, JTC 1/SC 7. ISO TR 15504. Geneva.

Jeffery, R. (2006). Achieving software development performance improvement through process change. In *Unifying the Software Process Spectrum*, pp. 43-53.

Kajko-Mattsson, M., Forssander, S., & Olsson, U. (2001). Corrective maintenance maturity model (CM3): maintainer's education and training. In *Software Engineering, 2001. ICSE 2001. Proceedings of the 23rd International Conference on* IEEE, pp. 610-619.

Kaminka, G. A., Veloso, M. M., Schaffer, S., Sollitto, C., Adobbati, R., Marshall, A. N., & Tejada, S. (2002). Gamebots: a flexible test bed for multi-agent team research. *Communications of the ACM*, *45*(1), 43-45.

Kent, B., & Andres, C. (2000). *Extreme programming explained: embrace change. Reading,* Mass.: Adison-Wesley.

Larman, C., & Basili, V. R. (2003). Iterative and incremental development: A brief history. *Computer*, (6), 47-56.

Li, L., Vaidyanathan, K., & Trivedi, K. S. (2002). An approach for estimation of software aging in a web server. In *Empirical Software Engineering, 2002. Proceedings. 2002 International Symposium on Empirical Software Engineering, 2002,* pp. 91-100.

Linacre, S. (2005). *Australian Social Trends 2005*. Australian Bureau of Statistics.

Martin, D., Rooksby, J., Rouncefield, M., & Sommerville, I. (2007). 'Good' Organizational Reasons for 'Bad' Software Testing: An Ethnographic Study of Testing in a Small Software Company. In *Software Engineering, 2007. ICSE 2007. 29th International Conference on Software Engineering (ICSE '07). IEEE Computer Society, Washington, DC, USA*, pp. 602-611.

Mary, B. C., Michael D. K., & Sandra S. (2011). *CMMI for Development: Guidelines for Process Integration and Product Improvement,* 3rd Edition. Carnegie Mellon.

McCaffery, F., Taylor, P., & Coleman, G. (2007). Adept: A Unified Assessment Method for Small Software Companies. *IEEE Software*, pp. 24 ±31.

McGuire, M. (2002). Managing networks: Propositions on what managers do and why they do it. *Public administration review*, *62*(5), 599-609.

Mouratidis, H., Giorgini, P., & Manson, G. (2003). Integrating security and systems engineering: Towards the modelling of secure information systems. In *Advanced Information Systems Engineering*, pp. 63-78.

Niazi, M., Wilson, D., & Zowghi, D. (2003). A maturity model for the implementation of software process improvement: an empirical study. *The Journal of Systems and Software*, pp. 155 ±172.

Niazi, M., Wilson, D., & Zowghi, D. (2006). Critical success factors for software process improvement implementation: an empirical study. *Software Process: Improvement and Practice, 11*(2), 193-211

Nicole M. (2006). Introduction to CMMI course participant. *Computer and information science,* pp. 37-45

Niessink, F., & Clerc, V. (2004). *IT Service CMM: A Pocket Guide. van Haren: Zalbommel.* The Netherlands.

Pesic, A. M. (2009). Business process management maturity model and Six Sigma: An integrated approach for easier networking. In *Proceedings of the international conference on economics and management of networks, EMNet. Springer, Sarajevo*, pp. 3-5.

Pfleeger, S. L., & Bohner, S. (1990). A framework for software maintenance metrics. In *Software Maintenance, 1990, Proceedings. Conference on Software Maintenance - Computer Soc. Press, IEEE, New York, 1990,* pp. 320-327.

Pigoski, T. M. (1996). *Practical software maintenance: best practices for managing your software investment*. John Wiley & Sons, Inc.

Pigoski, T.M. (2002). *Practical Software Maintenance: Best Practice for Managing your Software Investment*, Wiley.

Pressman, R. S. (2005). *Software engineering: a practitioner's approach*. Palgrave Macmillan.

Raffo, D. M., Vandeville, J. V., & Martin, R. H. (1999). Software process simulation to achieve higher CMM levels. *Journal of Systems and Software*, *46*(2), 163-172

Reifer, D. J. (2003). XP and the CMM. *IEEE Software*, (3), 14-15.

Rogers, E. M. (2003). *Diffusion of Innovations*. 5th Edition, the Free Press, New York.

Ronchetti, M., Succi, G., Pedrycz, W., & Russo, B. (2006). Early estimation of oftware size in object-oriented environments a case study in a CMM level 3 software firm. *Information Sciences*, *176*(5), 475-489.

SEI. (2006). *Standard CMMI® Appraisal Method for Process Improvement (SCAMPISM) A, Version 1.2: Method Definition Document.* CMU/SEI-2006-HB-002.

Shoham, Y. (1993). Agent-oriented programming. *Artificial intelligence*, pp. 51-92.

Sidky, A. (2007). *A Structured Approach to Adopting Agile Practices: The Agile Adoption Framework.* Doctoral Thesis. Virginia Polytechnic Institute and State University.

Sokolova, M. V., & Fernández-Caballero, A. (2009). Multi-Agent Systems Technology for Composite Decision Making in Complex Systems. In *Sustainability in Energy and Buildings*, pp. 29-38.

Sommerville, I. (2007). *Software Engineering, International computer Science Series*, 9th Edition, Addison-Wesley

SPIP06, (2007). *Software Process Improvement Partners. Supporting tools for SPI, 2006 and Product Improvement*, 2nd Edition, Addison Wesley.

Stone, N. H., Raitt, W. J., & Wright, K. H. (1999). The TSS-1R electrodynamic tether experiment: Scientific and technological results. *Advances in Space Research, 24*(8), 1037-1045.

Strode, D. (2007). Characterizing the Agile Methods. *New Zealand Journal of Applied Computing & Information Technology*, *11*(1).

Sudeikat, J., Braubach, L., Pokahr, A., & Lamersdorf, W. (2005). Evaluation of agent ±oriented software methodologies ±examination of the gap between modeling and platform. In *Agent-Oriented Software Engineering V*, pp. 126-141.

Swanson. (2006). A review of past and current overall models for maintenance management. *Quality in maintenance engineering journal*. pp. 1-5.

Trudel, S., Lavoie, J. M, Paré, M. C., & Suryn, W. (2006). The small company edicated software process quality evaluation method combining CMMI and ISO/IEC 14598. *Software Quality Journal*, pp. 7-23.

Von Wangenheim, C. G., Hauck, J. C. R., Zoucas, A., Salviano, C. F., McCaffery, F., & Shull, F. (2010). Creating software process capability/maturity models. *Software, IEEE*, *27*(4), 92-94.

Wang, X. Oconchuir, E. & Vidgen, R. (2008). A paradoxical Perspective on Contradictions in Agile Software Development. *European Conference of Information Systems (ECIS)*, Galway, Ireland.

Weyns, D., Omicini, A., & Odell, J. (2007). Environment as a first class abstraction in multiagent systems. *Autonomous agents and multi-agent systems, 14*(1), 5-30.

Winikoff, M., & Padgham, L. (2004). The Prometheus Methodology. In *Methodologies and Software Engineering for Agent Systems*, pp. 217-234.

Wooldridge, M., & Jennings, N. R. (1995). Agent theories, architectures, and languages: a survey. In *Intelligent agents*, pp. 1-39.

Wright, B. D., & Masters, G. N. (1982). *Rating Scale Analysis. Rasch Measurement*. MESA Press, 5835 S. Kimbark Avenue, Chicago, IL 60637.

Yoo, C., Yoon, J., Lee, B., Lee, C., Lee, J., Hyun, S., & Wu, C. (2004). An integrated model of ISO 9001: 2000 and CMMI for ISO registered organizations. In *Software Engineering Conference, 2004. 11th Asia-Pacific*, pp. 150-157.

Yoo, C., Yoon, J., Lee, B., Lee, C., Lee, J., Hyun, S., & Wu, C. (2006). A unified model for the implementation of both ISO 9001: 2000 and CMMI by ISO-certified organizations. *Journal of Systems and Software*, *79*(7), 954-961.

Yucalar, F. (2006). *Evaluation of companies which are being in the software sector has an understanding process focused quality management with the CMMI staged model.* Master Thesis, Maltepe University, Istanbul, Turkey.