

A CONCEPTUAL MODEL OF PAIR PROGRAMMING KNOWLEDGE-
BASED SHARING FOR IMPROVING PROGRAMMING SKILLS

SAMARA RAHEEM ISSA

MASTER OF DEGREE
UNIVERSITI UTARA MALAYSIA
2014

A CONCEPTUAL MODEL OF PAIR PROGRAMMING KNOWLEDGE-
BASED SHARING FOR IMPROVING PROGRAMMING SKILLS

A Thesis submitted to the UUM College of Arts and Sciences in
fulfillment of the requirements for the degree of Master of Science in
Information Technology Universiti Utara Malaysia

by
Samara Raheem Issa

© 2014, Samara

Permission to Use

In presenting this thesis in fulfilment of the requirements for a postgraduate degree from Universiti Utara Malaysia, I agree that the Universiti Library may make it freely available for inspection. I further agree that permission for the copying of this thesis in any manner, in whole or in part, for scholarly purpose may be granted by my supervisor(s) or, in their absence, by the Dean of Awang Had Salleh Graduate School of Arts and Sciences. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to Universiti Utara Malaysia for any scholarly use which may be made of any material from my thesis.

Requests for permission to copy or to make other use of materials in this thesis, in whole or in part, should be addressed to:

Dean of Awang Had Salleh Graduate School of Arts and Sciences
UUM College of Arts and Sciences
Universiti Utara Malaysia
06010 UUM Sintok

Abstract

One of eXtream Programming practices is Pair Programming (PP) (the pair consists of a driver and a navigator), which is used for promoting knowledge sharing among students. This practice encourages students to think creatively of programming solutions, and simplify learning, especially for difficult course such as Java. By applying PP, students are enforced to improve their social skills as they communicate with each others. Despite the numerous benefits of PP (discussed by previous studies), statistics show lack of demonstrating the extent at which the knowledge sharing, communication and transfer between the driver and the navigator can improve the code quality. Therefore this study aims propose a conceptual model of a PP knowledge-based sharing for improving programming skills. In order to achieve the stated objective, PP laboratory assignments were conducted and compared to evaluate the impact of PP on code quality with and without adopting the conceptual model. The conceptual model was validated by analyzing the collected data from the participants of PP laboratory assignment using Partial Least Square form of Structural Equation Modeling (PLS-SEM). The findings of the study show that socialization, combination, and internalization are the determinant factors for achieving better code quality in PP environment. The findings of this study would be benefited to academic environment especially the agile programmers in the pair programming domain.

Keywords: Pair programming, Tacit knowledge, Code quality and SECI model.

Acknowledgement

I would like to express my appreciation and gratitude to everyone who has contributed in completing this thesis. I would like to express my thanks to Dr. Mazni Omar and Dr. Mazida Ahmad. It was my pleasure to study under their supervision. It is not enough to thank you very much to them for their guidance to help me to achieve my goal. Without their valuable support, my thesis would not have been possible. Thanks to them for their comments which help to improve my work.

I am greatly indebted to my wonderful beloved mother Batool Saeed Ahmed for her support day by day. She has woken up whole night just to be by my side. I would like to thank my son Amir Sammer for his encouragement during my study. Thanks to my husband Sammer Saeed for his patience and understanding. My goal would not have been achieved without them.

Special thanks to my wonderful friends for their contributions and assistance. Specifically, Ms. Hanaa Khadum, Mdm. Mays Salman, Mr. Ahmed Sayed, Mr. Mohammed Fadil, Mr. Ishola, Ms. Hasana, and Ms. Wani.

I would also like to acknowledge the support and love of all of my relatives in many ways specially, Mdm. Najat Abdullah who supported me directly or indirectly in one way or another to the completion of this thesis.

I am very grateful to Dr. Azida Haji Zainol and Mam Noraziah Che Pa. They were very kind during the viva and during the period of the correction. Additionally their comments have helped to improve this work.

Finally, I am sorry for not being able to detail some of the helpful people here. I seek the magnanimity of Allah to bestow on all of them with His blessing and bountiful.

I had a very enjoyable study at Universiti Utara Malaysia (UUM). Not only, does it has a beautiful natural environment but the university also has helpful staff.

I dedicate this work as present to my father's spirit in his grave.

Table of Contents

Permission to Use	i
Abstract.....	ii
Acknowledgement	iii
List of Tables	1
List of Figures.....	1
List of Appendices	2
List of Abbreviations	2
CHAPTER ONE INTRODUCTION	3
1.1 Overview of the Research	3
1.2 Pair Programming	7
1.3 Code quality	7
1.4 SECI Model.....	8
1.5 Tacit Knowledge	9
1.6 Problem Statement	9
1.7 Research Questions	11
1.8 Research Objectives	11
1.9 Scope of the Research	11
1.10 Significance of the Study	12
1.11 Conclusion of the Chapter.....	12
CHAPTER TWO LITERATURE REVIEW	13
2.1 Introduction	13
2.2 Knowledge	13
2.3 Knowledge Management	15
2.4 Knowledge Management in the Industry	16
2.5 Knowledge Management in Education	17
2.6 Models in knowledge management.....	17
2.6.1 Wiig KM Model (1993).....	18
2.6.2 Schultze and Leidner caution (2002)	18
2.7 Elements of Knowledge Management	19
2.8 Knowledge Sharing.....	19
2.9 System Development Methods	20

2.10 Agile	21
2.11 Pair programming (PP)	22
2.12 The way PP session is conduct.....	23
2.12.1 PP in Academia.....	24
2.12.2 PP in Industry.....	24
2.13 Agile Software Development Adoption to Knowledge Sharing.....	25
2.14 Pair Programming and Knowledge Sharing.....	26
2.15 Correctness of Code Quality	27
2.16 Code Quality and Pair Programming	29
2.17 Conclusion	29
CHAPTER THREE RESEARCH METHODOLOGY	30
3.1 Introduction	30
3.2 Research Design.....	30
3.3 Define Research Context.....	34
3.4 Preliminary Study	34
3.5 Experiment Definition.....	34
3.6 Experimental planning	35
3.6.1 Context Selection	35
3.6.2 Hypothesis Construction	36
3.6.3 Variables Selection	37
3.6.4 Subjects Selection	38
3.6.5 Experiment Design.....	38
3.6.6 Instrumentation	43
3.6.7 Validity Evaluation	44
3.6.7.1 Internal validity	44
3.6.7.2 External validity	45
3.6.7.3 Construct validity	45
3.6.7.4 Conclusion validity.....	45
3.7 Operation.....	46
3.8 Analysis and Interpretation	46
3.9 Summary of the chapter	47
CHAPTER FOUR RESEARCH FINDINGS	48

4.1 Introduction	48
4.2 Descriptive Statistics of Respondents for Experiments 1 and 2 (NSYR & YSYR)	48
4.3 Structural Equation Modeling	51
4.4 Analytical Activities in Structural Equation Modeling.....	52
4.4.1 Reliability of Internal Consistency	53
4.4.2 Indicator Reliability	56
4.4.3 Convergent validity.....	56
4.4.4 Discriminant validity	58
4.5 Validation of Structural Model	61
4.5.1 Coefficient of Determination (R^2).....	62
4.5.2 Path Coefficient	63
4.6 Research Model.....	68
4.7 Summary	69
CHAPTER FIVE DISCUSSION AND CONCLUSION	70
5.1 Introduction	70
5.2 Overview of the Research	70
5.3 Discussion of the results based on the objectives:	72
5.3.1 Discussion based on the first objective:	72
a. Effect of Socialization on Code Quality	72
b. Effect of Externalization on Code Quality	73
c. Effect of Combination on Code Quality	74
d. Effect of Internalization on Code Quality	74
5.3.2 Discussion based on the second objective:	76
5.3.3 Discussion based on the third objective:	76
5.4 Study Contribution	77
5.5 Problems and Limitation	78
5.6 Future Work	78
5.7 Recommendation.....	79
5.8 Summary	79
REFERENCES.....	80

List of Tables

Table 2.1 PP and Knowledge Sharing	27
Table 3.1 Relationships of Goals, Research Questions, and Research Objectives	31
Table 3.2 Relationships of Objectives, Stages and Deliverable.....	32
Table 3.3 Hypotheses Of The Study	36
Table 3.4 Repeated Measure Design.....	39
Table 3.5 SECI Guidelines	42
Table 4.1 Demographic Statistics of Experiment 1 (NSYR)	49
Table 4.2 Demographic Statistics of Experiment 2 (YSYR)	51
Table 4.3 Descriptive and Reliability Statistics for NSYR.....	54
Table 4.4 Descriptive and Reliability Statistics for YSYR.....	55
Table 4.5 AVE Values of NSYS Model	57
Table 4.6 AVE Values of YSYS Model	58
Table 4.7 Discriminant Validity for YSYR	59
Table 4.8 Discriminant Validity for NSYR	59
Table 4.9 Discriminant Validity (Cross Loading Criterion) For NSYR Model.....	60
Table 4.10 Discriminant validity (cross loading criterion) for YSYR model.....	61
Table 4.11 Standard Values for Assessing Measurement Model	64
Table 4.12 Supported Hypotheses of The Study.....	65
Table 4.13 Standard Values for Validating Structural Model.....	66

List of Figures

Figure 1.1. Conceptual idea of the study	5
Figure 1.2. Relationship of keywords	6
Figure 2.1. Relationship of elements in related to knowledge sharing	20
Figure 3.1. Research Process	33
Figure 3.2. Independent and dependant variables	37
Figure 3.3. Lab experiments	40
Figure 4.1. Result of NSYR structural model.....	62
Figure 4.2. Result of YSYR structural model.....	63
Figure 4.3. Results of the hypothesis for experiment 1 (NSYR)	66
Figure 4.4. Results of the hypothesis for experiment 2 (YSYR)	67

Figure 4.5. Research model	68
Figure 5.1. A revised model for PP knowledge-based sharing for improving programming skills	75

List of Appendices

Appendix A QUESTIONNAIRE.....	87
-------------------------------	----

List of Abbreviations

CQ	Code Quality
CS	Computer Science
EK	Explicit Knowledge
KM	Knowledge Management
PLS	Partial Least Square
PP	Pair Programming
SE	Software Engineering
SECI	Socialization Externalization Combination Internalization
SEM	Sequential Equation Modeling
TK	tacit knowledge

CHAPTER ONE

INTRODUCTION

1.1 Overview of the Research

This study views with concern on the possibility of improving students' programming skills. Many students think that computer science (CS) and software engineering (SE) courses take the lead in dropout rates than other courses (Md Rejab, Omar, & Ahmad, 2013). This motivates the practitioners to urgently employ a technique or practice that can facilitate teaching and learning practices in CS/SE courses.

The success of Pair Programming (PP) in IT industry has been seen in terms of enhancing knowledge transfer (di Bella et al., 2013b), facilitating integration of novice members (Sillitti, Succi, & Vlasenko, 2012; SOUZA, 2012), reducing costs for training (Sillitti et al., 2012), and improving coding structure (Xu & Rajlich, 2006). This encourages practitioners in pedagogical context to rely on PP to overcome students' failure in programming course. Additionally, it not only encourages students to accept programming curricula, but also encourages to innovate in producing better end-programs (di Bella et al., 2013b). SE community accepts PP as one of many innovative approaches that had been considered to overcome programming issues in CS/SE courses (Omar, Syed-Abdullah, & Yasin, 2010; Syed-Abdullah, Omar, Hamid, bt Ismail, & Jusoff, 2009). Eventually, in the late 1990s, PP has been embedded in the teaching of CS (Keefe, Sheard, & Dick, 2006; Rimington, 2010).

The contents of
the thesis is for
internal user
only

REFERENCES

- Abidi, S. S. R., Cheah, Y.-N., & Curran, J. (2005). A knowledge creation infrastructure to acquire and crystallize the tacit knowledge of health-care experts. *Information Technology in Biomedicine, IEEE Transactions on*, 9(2), 193-204.
- Abran, A., Moore, J. W., Bourque, P., Dupuis, R., & Tripp, L. (2004). Guide to the software engineering body of knowledge, 2004 version. *IEEE Computer Society, 1*.
- Aggestam, L. (2006). Learning organization or knowledge management—which came first, the chicken or the egg. *Information technology and control*, 35(3A), 295-302.
- Ahmad, M., Md Rejab, M., Syazwan Abdulah, M., Omar, M., Bariah Ahmad, K., & Abbas, M. (2012). Measuring tacit knowledge acquired during problem based learning teaching method in learning management system environment. *AWERProcedia Information Technology and Computer Science, 1*.
- Alegre, J., Sengupta, K., & Lapiedra, R. (2013). Knowledge management and innovation performance in a high-tech SMEs industry. *International Small Business Journal*, 31(4), 454-470.
- Alipour, F., Idris, K., & Karimi, R. (2011). Knowledge Creation and Transfer: Role of Learning Organization. *International Journal of Business Administration*, 2(3).
- Ally, M. (2004). Designing effective learning objects. *Online education using learning objects*, 87-97.
- Anantatmula, V., & Kanungo, S. (2005). Establishing and structuring criteria for measuring knowledge management efforts. In *Proceedings of The 38th Annual Hawaii International Conference on System Sciences, 2005*
- Argote, L. (2013). *Organizational learning: Creating, retaining and transferring knowledge*: Springer.
- Bass, L., Clements, P., & Kazman, R. (2003). *Software architecture in practice*: Addison-Wesley Professional.
- Beck, K. (2001). *Planning extreme programming*: Addison-Wesley Professional.
- Berenson, S., Williams, L., & Slaten, K. (2005). Using Pair Programming and Agile Development Methods in a University Software Engineering Course to Develop a Model of Social Interactions. In *Proceedings of the International Conference on Crossing Cultures, Changing Lives*.
- Billett, S., & Choy, S. (2013). Learning through work: emerging perspectives and new challenges. *Journal of Workplace Learning*, 25(4), 264-276.
- Blech, J. O., & Glesner, S. (2004). A Formal Correctness Proof for Code Generation from SSA Form in Isabelle/HOL. *GI Jahrestagung (2)*, 51, 449-458.
- Brockmann, E. N., & Anthony, W. P. (2002). Tacit knowledge and strategic decision making. *Group & Organization Management*, 27(4), 436-455.
- Cayaba, C., & Pablo, Z. (2013). A Qualitative Investigation of the SECI Model's Knowledge Conversions in the Applications Development Context.

- Chau, T., & Maurer, F. (2004). Knowledge sharing in agile software teams *Logic versus approximation* (pp. 173-183): Springer.
- Chigona, W., & Pollock, M. (2008). Pair programming for information systems students new to programming: Students' experiences and teachers' challenges. In *Proceedings of the International Conference on Management of Engineering & Technology, 2008*. PICMET 2008. Portland.
- Chin, W. W. (1998). The partial least squares approach to structural equation modeling. *Modern methods for business research*, 295(2), 295-336.
- Ciolkowski, M., & Schlemmer, M. (2002). Experiences with a case study on pair programming. In *Proceedings of the Workshop on Empirical Studies in Software Engineering*.
- Cockburn, A. (2004). *Crystal clear: a human-powered methodology for small teams*: Pearson Education
- Cockburn, A., & Williams, L. (2000). The costs and benefits of pair programming. *Extreme programming examined*, 223-247.
- Cohen, L., Manion, L., & Morrison, K. (2000). *Research Methods in Education* [5 th edn] London: Routledge Falmer. *Teaching in Higher Education*, 41.
- Crawford, B., de la Barra, C. L., Soto, R., Dorochesi, M., & Monfroy, E. (2013). The Role of Knowledge Management in Agile Software Development *HCI International 2013-Posters' Extended Abstracts* (pp. 17-21): Springer.
- Dalkir, K. (2013). *Knowledge management in theory and practice*: Routledge.
- de Azevedo Santos, M., de Souza Bermejo, P. H., de Oliveira, M. S., & Tonelli, A. O. (2011). Agile practices: An assessment of perception of value of professionals on the quality criteria in performance of projects. *Journal of Software Engineering and Applications*, 4(12), 700-709.
- di Bella, E., Fronza, I., Phaphoom, N., Sillitti, A., Succi, G., & Vlasenko, J. (2013a). Pair Programming and Software Defects--A Large, Industrial Case Study. *Software Engineering, IEEE Transactions on*, 39(7), 930-953.
- di Bella, E., Fronza, I., Phaphoom, N., Sillitti, A., Succi, G., & Vlasenko, J. (2013b). Pair Programming and Software Defects-A Large, Industrial Case Study.
- Dijkstra, E. W. (1993). *A discipline of programming* (Vol. 1): prentice-hall Englewood Cliffs.
- Dingsøyr, T., Nerur, S., Balijepally, V., & Moe, N. B. (2012). A decade of agile methodologies: Towards explaining agile software development. *Journal of Systems and Software*, 85(6), 1213-1221.
- Dorairaj, S., Noble, J., & Malik, P. (2012). Knowledge Management in Distributed Agile Software Development. In *Proceedings of the Agile Conference (AGILE)*, 2012.
- Dudley, P. (2013). Teacher learning in Lesson Study: What interaction-level discourse analysis revealed about how teachers utilised imagination, tacit knowledge of teaching and fresh evidence of pupils learning, to develop practice knowledge and so enhance their pupils' learning. *Teaching and Teacher Education*, 34, 107-121.
- Dybå, T., & Dingsøyr, T. (2008). Empirical studies of agile software development: A systematic review. *Information and software technology*, 50(9), 833-859.
- Erdogmus, H. (2005). On the effectiveness of test-first approach to programming.

- Fengjie, A., Fei, Q., & Xin, C. (2004). Knowledge sharing and web-based knowledge-sharing platform. In *Proceedings of IEEE International Conference on E-Commerce Technology for Dynamic E-Business*, 2004..
- Fornell, C., & Bookstein, F. L. (1982). Two structural equation models: LISREL and PLS applied to consumer exit-voice theory. *Journal of Marketing Research (JMR)*, 19(4).
- Fornell, C., & Larcker, D. F. (1981). Evaluating structural equation models with unobservable variables and measurement error. *Journal of Marketing Research (JMR)*, 18(1).
- Foss, N. J., Lyngsie, J., & Zahra, S. A. (2013). The role of external knowledge sources and organizational design in the process of opportunity exploitation. *Strategic Management Journal*.
- Gerard, J. G. (2003). Measuring knowledge source tacitness and explicitness: A comparison of paired items. In *Proceedings of the 5th Annual Organizational Learning and Knowledge Conference*.
- Gerholm, T. (1990). On tacit knowledge in academia. *European Journal of Education*, 263-271.
- Guthrie, S. (1996). The Role of Tacit Knowledge in Judgement and Decision Making.
- Hannay, J. E., Arisholm, E., Engvik, H., & Sjoberg, D. I. (2010). Effects of personality on pair programming. *Software Engineering, IEEE Transactions on*, 36(1), 61-80.
- Hannay, J. E., Arisholm, E., Engvik, H., & Sjoberg, D. I. K. (2010). Effects of Personality on Pair Programming. *IEEE Transactions on Software Engineering*, 36(1), 61-80. doi: <http://dx.doi.org/10.1109/TSE.2009.41>
- Hashim, K. F. (2012). *Understanding the determinants of continuous knowledge sharing intention within business online communities*. AUT University.
- He, W., Qi, C., Xu, X., & Guiyang, P. (2013). Linking Knowledge Sharing and Employee Creativity: Decomposing Knowledge Mode and Improving the Measure of Tacit Knowledge Sharing.
- Henseler, J., Ringle, C. M., & Sinkovics, R. R. (2009). The use of partial least squares path modeling in international marketing. *Advances in international marketing*, 20(1), 277-319.
- Ho, C.-w. (2003). Tacit Knowledge Management and Pair Programming.
- Ikujiro, N., & Takeuchi, H. (1995). The knowledge-creating company. *Harvard Business Review on Knowledge Management*.
- Iranzo, P. J. (2002). On the correctness of the factoring transformation *Functional and Logic Programming* (pp. 119-133): Springer.
- Jabar, M. A., Sidi, F., & Selamat, M. H. (2010). Tacit knowledge codification. *Journal of Computer Science*, 6(10), 1170.
- Jensen, R. (2003). A pair programming experience. *The journal of defensive software engineering*, 16(3), 22-24.
- Jiangping, W., Ming, Z., & Yahui, Z. (2013). Case Study on Tacit Knowledge Management Systems within X Company. *Technology and Investment*, 4, 92.
- Kashefipour, S., Falconer, R., & Lin, B. (2002). Modeling longitudinal dispersion in natural channel flows using ANNs. *River Flow 2002*, 111-116.

- Kashif, M., & Kelly, K. (2013). *Knowledge Management and Sharing Within Project Teams: A qualitative Study of Ericsson*. Mälardalen University.
- Kavitha, R., & Ahmed, I. (2011). A Knowledge Management Framework for Agile Software Development Teams. In *Proceedings of the International Conference on Process Automation, Control and Computing (PACC)*, 2011.
- Keefe, K., Sheard, J., & Dick, M. (2006). Adopting XP practices for teaching object oriented programming. In *Proceedings of of the 8th Australasian Conference on Computing Education-Volume 52*.
- Kent, A. E. S., & Selic, B. (2000). «UML» 2000–The Unified Modeling Language.
- Kimble, C. (2013). What Cost Knowledge Management? The Example of Infosys. *Global Business and Organizational Excellence*, 32(3), 6-14.
- King, W. R. (2009). *Knowledge management and organizational learning*: Springer.
- Kline, R. B. (2005). Principles and Practice of Structural Equation Modeling. 2005. *New York, NY: Guilford*.
- Koulikov, M. (2011). Emerging problems in knowledge sharing and the three new ethics of knowledge transfer. *Knowledge Management & E-Learning: An International Journal*, 3(2), 237--250.
- Lavrakas, P. J. (2008). *Encyclopedia of survey research methods*: Sage.
- Layman, L. (2006). Changing students' perceptions: An analysis of the supplementary benefits of collaborative software development. In *Proceedings of the 19th Conference on Software Engineering Education and Training*, 2006. Proceedings..
- Leonard, N., & Insch, G. S. (2005). Tacit knowledge in academia: a proposed model and measurement scale. *The Journal of Psychology: Interdisciplinary and Applied*, 139(6), 495-512.
- Lewis, B. R., Templeton, G. F., & Byrd, T. A. (2005). A methodology for construct development in MIS research. *European Journal of Information Systems*, 14(4), 388-400.
- LI, Z.-x., WANG, Q., & Cao, L. (2013). An Analysis of the Structure and Evaluation Methods of Individual Tacit Knowledge.
- Liao, S.-h., & Wu, C.-c. (2009). The relationship among knowledge management, organizational learning, and organizational performance. *International Journal of Business and Management*, 4(4), P64.
- Liaw, S.-S., Huang, H.-M., & Chen, G.-D. (2007). Surveying instructor and learner attitudes toward e-learning. *Computers & Education*, 49(4), 1066-1080.
- Lui, K. M., Barnes, K. A., & Chan, K. C. (2010). Pair Programming: Issues and Challenges *Agile Software Development* (pp. 143-163): Springer.
- Mahdavi-Hezavehi, S., Galster, M., & Avgeriou, P. (2013). Variability in quality attributes of service-based software systems: A systematic literature review. *Information and software technology*, 55(2), 320-343.
- Markoski, B., Hotomski, P., Malbaški, D., & Obradović, D. (2013). Resolution methods in proving the program correctness. *The Yugoslav Journal of Operations Research ISSN: 0354-0243 EISSN: 2334-6043*, 17(2).
- Mazida, A. (2010). An investigation of knowledge creation processes in LMS-supported expository and PBL teaching methods. *Unpublished doctoral dissertation*). *Universiti Sains Malaysia*.

- Md Rejab, M., Omar, M., & Ahmad, M. (2012). Knowledge internalization in pair programming practices. *Journal of Information and Communication Technology (JICT)*, 11, 163-177.
- Murphy, G., & Salomone, S. (2013). Using social media to facilitate knowledge transfer in complex engineering environments: a primer for educators. *European Journal of Engineering Education*, 38(1), 70-84.
- Nissen, M. E. (2002). An extended model of knowledge-flow dynamics. *Communications of the Association for Information Systems*, 8(18), 251-266.
- Nonaka, I., & Konno, N. (1998). The Concept of "Ba": Building A Foundation For Knowledge Creation. *California management review*, 40(3).
- Nonaka, I., & Takeuchi, H. (1995). The knowledge-creating company. 1995.
- Nonaka, I., & Takeuchi, H. (1996). The knowledge-creating company: How Japanese companies create the dynamics of innovation. *Long Range Planning*, 29(4), 592.
- Nunnally, J. C. (1994). Bernstein, IH (1994). Psychometric theory: New York: McGraw-Hill.
- Okumus, F. (2013). Facilitating knowledge management through information technology in hospitality organizations. *Journal of Hospitality and Tourism Technology*, 4(1), 4-4.
- Omar, M., Romli, R., & Hussain, A. (2007). Automated Tool to Assess Pair Programming Code quality.
- Omar, M., Syed-Abdullah, S.-L., & Yasin, A. (2010) Adopting Agile Approach: A Case in Malaysia.
- Porter, L., Guzdial, M., McDowell, C., & Simon, B. (2013). Success in introductory programming: what works? *Communications of the ACM*, 56(8), 34-36.
- Rejab, M. M., Omar, M., & Mazida Ahmad, K. B. A. (2011). Pair Programming in Inducing Knowledge Sharing.
- rey Voas, J. (1996). Testing software for characteristics other than correctness: Safety, failure tolerance, and security.
- Rimington, K. B. (2010). Expanding the Horizons of Educational Pair Programming: A Methodological Review of Pair Programming in Computer Science Education Research.
- Ryan, S., & O'Connor, R. V. (2013). Acquiring and Sharing Tacit Knowledge in Software Development Teams: An Empirical Study. *Information and software technology*.
- Sajeev, A., & Datta, S. (2013). Introducing Programmers to Pair Programming: A Controlled Experiment *Agile Processes in Software Engineering and Extreme Programming* (pp. 31-45): Springer.
- Salleh, N., Mendes, E., & Grundy, J. (2011). Empirical studies of pair programming for CS/SE teaching in higher education: A systematic literature review. *Software Engineering, IEEE Transactions on*, 37(4), 509-525.
- Santos, V., Goldman, A., & Roriz Filho, H. (2013). The influence of practices adopted by agile coaching and training to foster interaction and knowledge sharing in organizational practices. In *Proceedings of the 46th Hawaii International Conference on System Sciences (HICSS)*, 2013.

- Schultze, U., & Leidner, D. E. (2002). Studying knowledge management in information systems research: discourses and theoretical assumptions. *MIS quarterly*, 213-242.
- Sharma, D. (2014). Knowledge Management & Organizational Structure: A study of indian companies. *International Journal of Organizational Behaviour & Management Perspectives*, 2(4), 585-593.
- Sillitti, A., Succi, G., & Vlasenko, J. (2012). Understanding the impact of Pair Programming on Developers Attention. In *Proceedings of the ICSE2012*, Zurich, Switzerland.
- Singh, A., Singh, K., & Sharma, N. (2013). Knowledge Management: the agile way. In *Proceedings of the Information and Knowledge Management*.
- Sommerville, J., & Craig, N. (2012). *Implementing IT in construction*: Routledge.
- SOUZA, C. R. D. (2012). Fostering Inter-Team Knowledge Sharing Effectiveness In Agile Software Development.
- Stankosky, M. (2005). *Creating the discipline of knowledge management*: Routledge.
- Straub, D., Boudreau, M.-C., & Gefen, D. (2004). Validation Guidelines For Is Positivist Research. *Communications of the Association for Information Systems*, 13.
- Straus, S. E., Tetroe, J., & Graham, I. D. (2013). *Knowledge translation in health care: moving from evidence to practice*: John Wiley & Sons.
- Sung, S. Y., & Choi, J. N. (2013). Do organizations spend wisely on employees? Effects of training and development investments on learning and innovation in organizations. *Journal of Organizational Behavior*.
- Swamidurai, R. (2009). *Collaborative-adversarial pair (CAP) programming*. 3394643 Ph.D., Auburn University, Ann Arbor. Retrieved from <http://eserv.uum.edu.my/docview/304850186?accountid=42599> ProQuest Dissertations & Theses Full Text database.
- Syed-Abdullah, S. L., Omar, M., Hamid, M. N. A., bt Ismail, C. L., & Jusoff, K. (2009). Positive affects inducer on software quality. *Computer and Information Science*, 2(3), P64.
- Tomayko, J. E. (2002). A comparison of pair programming to inspections for software defect reduction. *Computer Science Education*, 12(3), 213-222.
- Urbach, N., & Ahlemann, F. (2010). Structural equation modeling in information systems research using partial least squares. *Journal of Information Technology Theory and Application*, 11(2), 5-40.
- Wan, J., Wan, D., Luo, W., & Wan, X. (2011). Research on Explicit and Tacit Knowledge Interaction in Software Process Improvement Project. *JSEA*, 4(6), 335-344.
- Wang, L. (2009). The multi-dimension spiral model of the knowledge-based corporation-theoretical thinking of beijing future advertising corporation. *Journal of Software*, 4(5), 469-477.
- Werner, L., Denner, J., Campe, S., Ortiz, E., DeLay, D., Hartl, A. C., & Laursen, B. (2013). Pair programming for middle school students: does friendship influence academic outcomes? In *Proceedings of the 44th ACM technical symposium on Computer science education*.

- Wiig, K. M. (2003). A knowledge model for situation-handling. *Journal of Knowledge Management*, 7(5), 6-24.
- Williams, L. (2001). Integrating pair programming into a software development process. In *Proceedings of the 14th Conference on Software Engineering Education and Training*, 2001.
- Williams, L. (2007). Lessons learned from seven years of pair programming at North Carolina State University. *ACM SIGCSE Bulletin*, 39(4), 79-83.
- Williams, L. (2010). Pair Programming. *Encyclopedia of Software Engineering*, 2.
- Williams, L., McCrickard, D. S., Layman, L., & Hussein, K. (2008). Eleven guidelines for implementing pair programming in the classroom. In *Proceedings of the Agile 2008 Conference*. AGILE'08..
- Williams, L., Shukla, A., & Anton, A. I. (2004). An initial exploration of the relationship between pair programming and Brooks' law. In *Proceedings of the Agile Development Conference*, 2004.
- Wohlin, C., Runeson, P., Hst, M., Ohlsson, M. C., Regnell, B., & Wessln, A. (2012). *Experimentation in software engineering*: Springer Publishing Company, Incorporated.
- Wood, W. A., & Kleb, W. L. (2003). Exploring XP for scientific research. *Software, IEEE*, 20(3), 30-36.
- Wray, S. (2010). How pair programming really works. *Software, IEEE*, 27(1), 50-55.
- Xu, S., & Rajlich, V. (2006). Empirical validation of test-driven pair programming in game development. In *Proceedings of the 5th IEEE/ACIS International Conference on Computer and Information Science, 2006 and 2006 1st IEEE/ACIS International Workshop on Component-Based Software Engineering, Software Architecture and Reuse*. ICIS-COMSAR 2006..
- Yi, J. (2006). Externalization of tacit knowledge in online environments. *International Journal on E-learning*, 5(4), 663-674.