

SECURE APPRECIATIVE INQUIRY FUZZY QUANTIFICATION
TECHNIQUE FOR QUANTIFYING SOFTWARE SECURITY
REQUIREMENTS

Omar Isam Homaidi Al Mrayat
(Matric No. 4110194)

Thesis submitted in fulfillment for the degree of
DOCTOR OF PHILOSOPHY
IN
SCIENCE AND TECHNOLOGY
(COMPUTER SCIENCE)

Faculty of Science and Technology
UNIVERSITI SAINS ISLAM MALAYSIA
NICALI

January 2015

AUTHOR DECLARATION

I hereby declare that the work in this thesis is my own except for quotations and summaries which have been duly acknowledge.

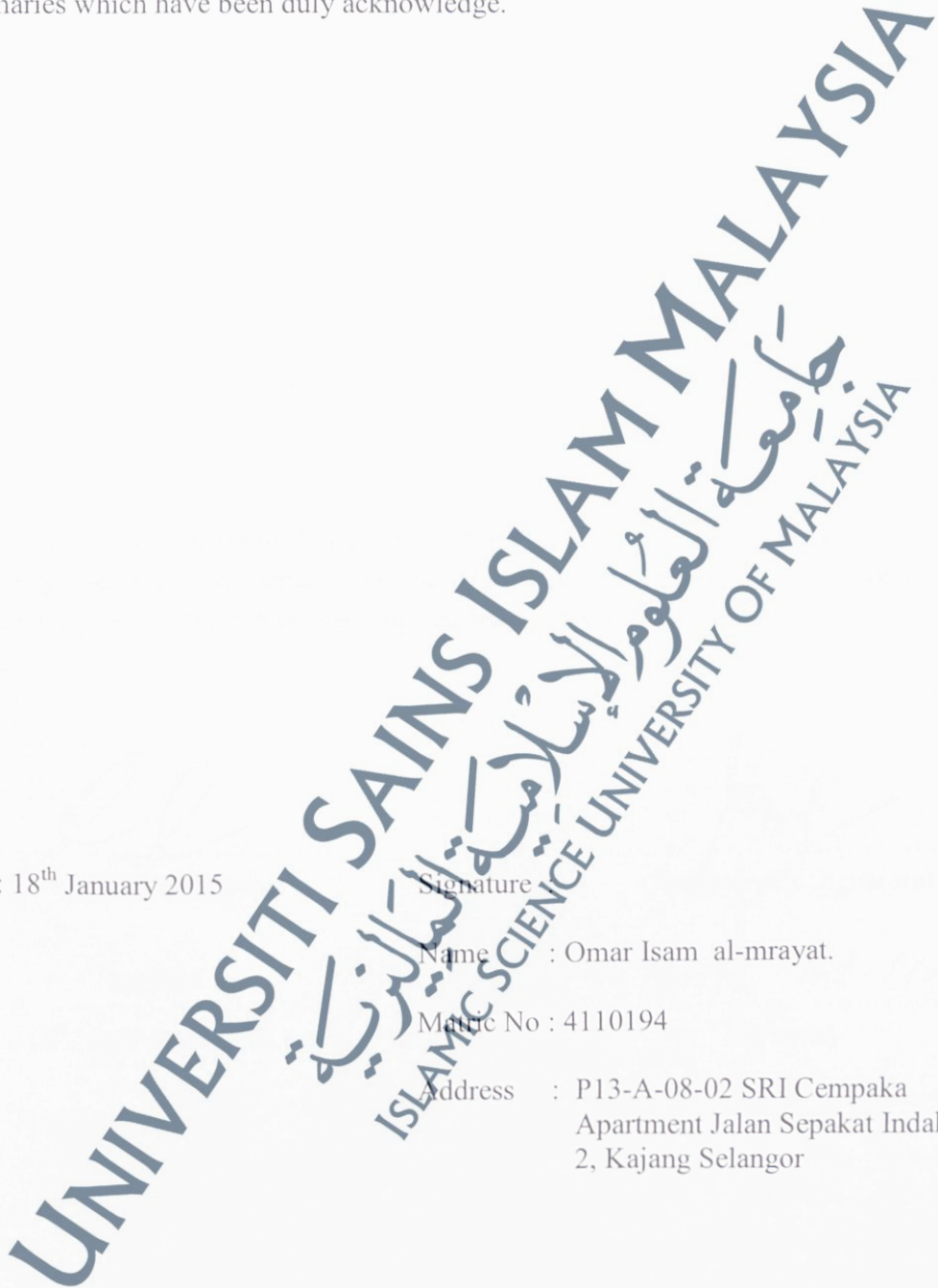
Date: 18th January 2015

Signature

Name : Omar Isam al-mrayat.

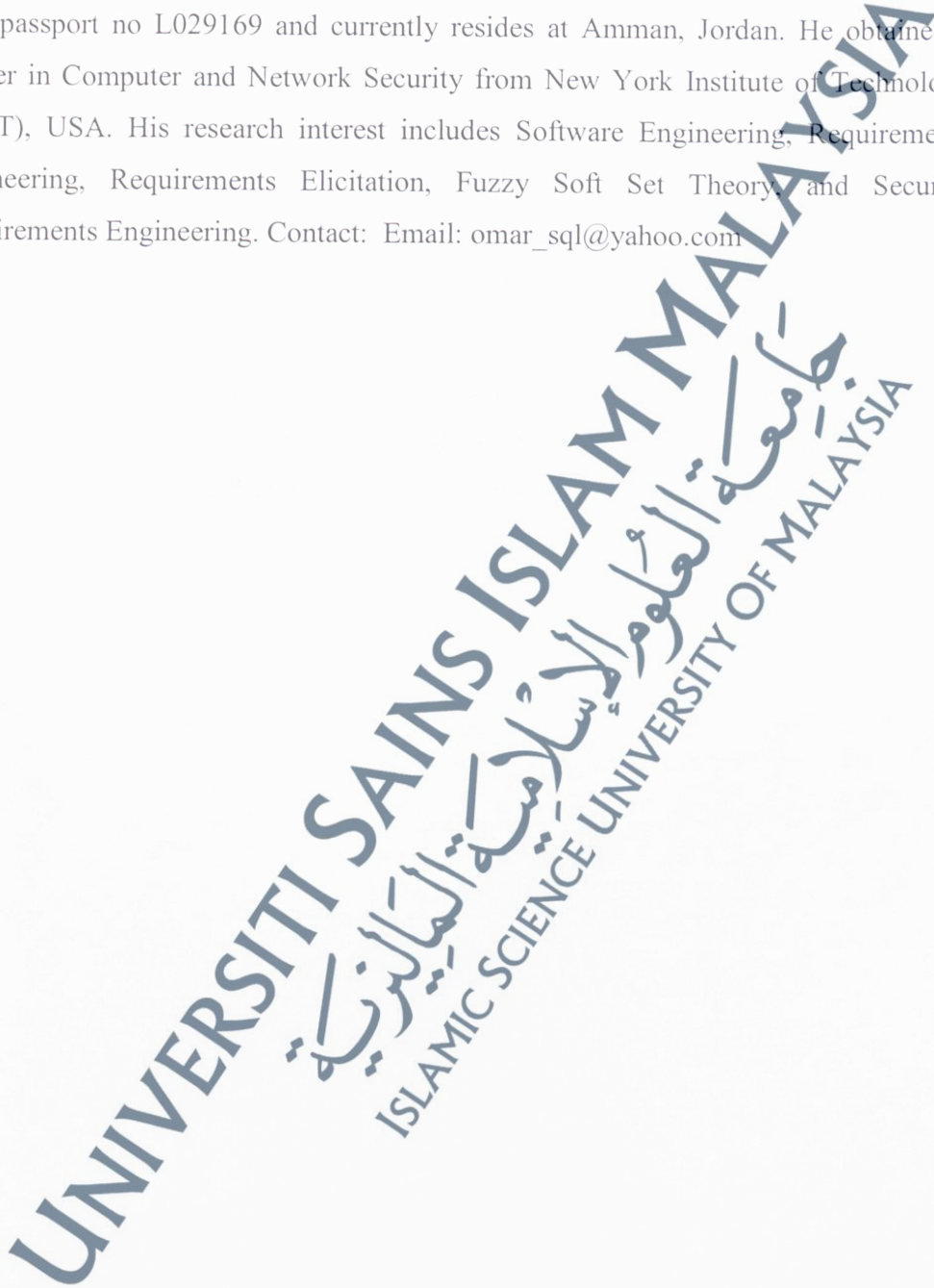
Matric No : 4110194

Address : P13-A-08-02 SRI Cempaka
Apartment Jalan Sepakat Indah
2, Kajang Selangor



BIODATA OF AUTHOR

Omar Isam Homaidi AL- Mrayat is pursuing his PhD from faculty of Science and Technology, Universiti Sains Islam Malaysia with matric no 4110194. He is Jordanian with passport no L029169 and currently resides at Amman, Jordan. He obtained a Master in Computer and Network Security from New York Institute of Technology (NYIT), USA. His research interest includes Software Engineering, Requirements Engineering, Requirements Elicitation, Fuzzy Soft Set Theory, and Security Requirements Engineering. Contact: Email: omar_sql@yahoo.com

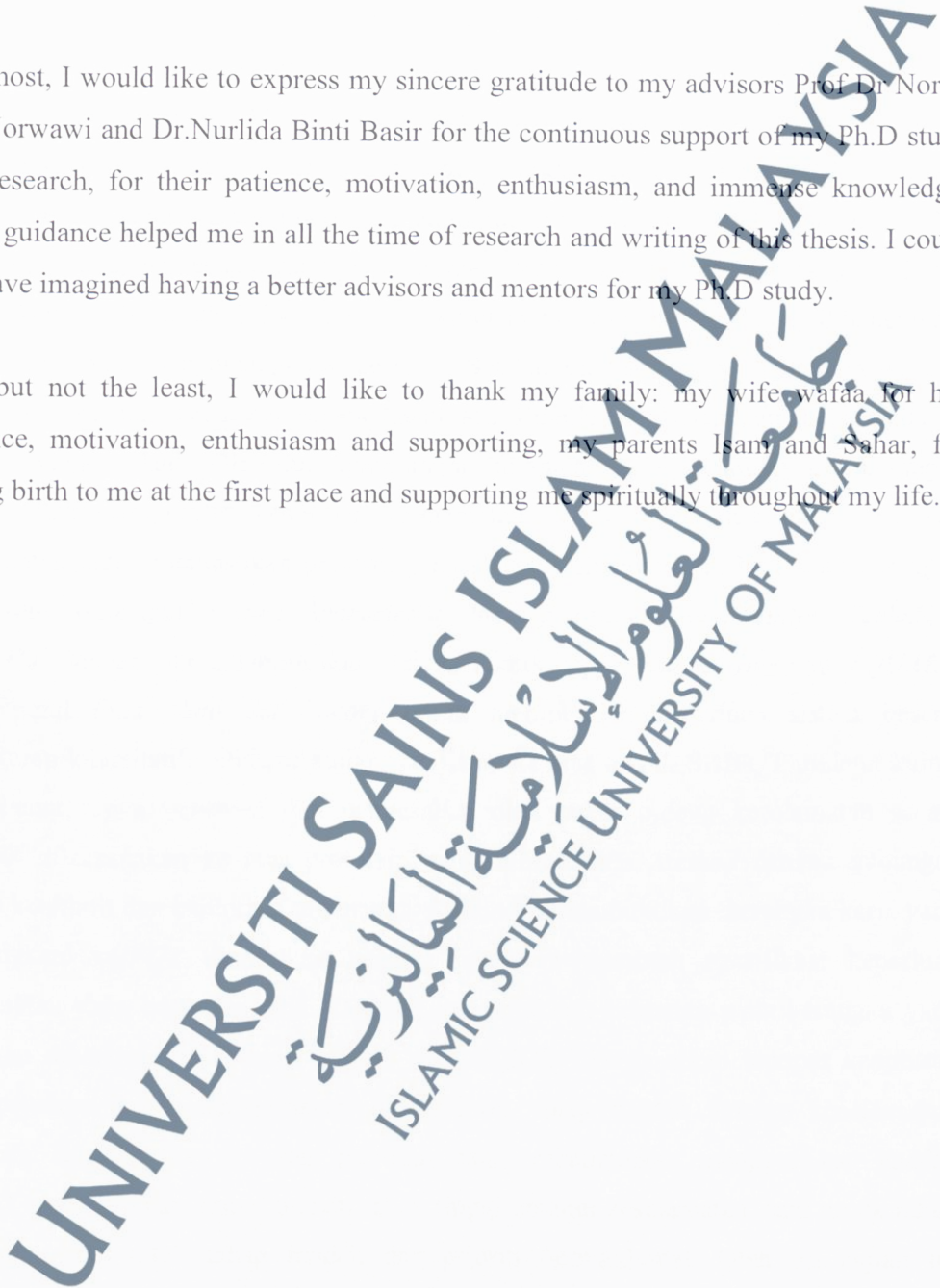


ACKNOWLEDGEMENT

This thesis was not by any means a solo effort. It's my pleasure to convey my gratitude to them all in my humble acknowledgment.

Foremost, I would like to express my sincere gratitude to my advisors Prof Dr Norita Md Norwawi and Dr.Nurlida Binti Basir for the continuous support of my Ph.D study and research, for their patience, motivation, enthusiasm, and immense knowledge. Their guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisors and mentors for my Ph.D study.

Last but not the least, I would like to thank my family: my wife wafaa for her patience, motivation, enthusiasm and supporting, my parents Isam and Sahar, for giving birth to me at the first place and supporting me spiritually throughout my life.



ABSTRAK

Pembangun perisian umumnya memfokuskan kepada fungsi utama dan ciri-ciri asas sistem di mana aspek keselamatan ditumpukan di fasa penghujung dan lewat. Keperluan keselamatan yang kurang diberi perhatian di fasa awal pembangunan boleh menyebabkan perisian yang lemah ciri keselamatan serta melibatkan kos yang berlipat ganda tinggi untuk dibaiki disetiap fasa. Menilai tahap keselamatan perisian di fasa awal pembangunan sistem membantu menghasilkan rekabentuk, aplikasi yang selamat serta berupaya melindungi dari serangan yang bertujuan untuk merosak. Oleh itu pembangun sistem dan perisian perlukan pendekatan yang praktikal dan sistematik untuk mendapatkan maklumat yang cukup dan berkredibiliti ke atas tahap keselamatan sistem yang sedang dibangunkan di fasa awal kitaran pembangunan perisian. Setakat ini teknik yang bolehpercayaan dan kukuh atau kaedah mengukur secara kuantitatif keperluan keselamatan dalam industri perisian sangat terhad. Oleh itu, tujuan kajian ini adalah untuk membina satu kerangka kerja untuk memperoleh keperluan keselamatan dan mengkuantifikasikan keperluan keselamatan untuk memastikan perisian yang selamat dibangunkan. Usul satu kerangka kerja yang dipanggil *Secure Appreciative Inquiry Fuzzy Quantification Technique (SAIFQT)* dengan mengintegrasikan teknik-teknik *Appreciative Inquiry*, *SQUARE*, *CLASP* and *Fuzzy Soft Set Theory* untuk memperoleh keperluan sistem beserta pengukuran kuantitatif. Diakhir kajian, penilaian ke atas teknik SAIFQT melalui kajian kes sebenar, ujian penetrasi dan pengesahan oleh pakar bidang keselamatan secara heuristik dilaksanakan ke atas prototaip yang dibina. *Mix Method* dengan gabungan kaedah kualitatif dan eksploratif digunakan. Kajian ini membuktikan kerangka kerja yang dicadangkan SAIFQT dibuktikan berjaya untuk memperoleh spesifikasi keperluan keselamatan yang baru dan unik. Dapatan menunjukkan kekuatan pada kerangka yang diusulkan dibandingkan dengan kaedah normal SDLC berdasarkan laporan keputusan ujian penetrasi yang menggariskan tiga prioriti tahap rendah amaran keselamatan. Manakala laporan ujian ke atas prototaip yang dibangunkan menggunakan kaedah SAIFQT menunjukkan empat prioriti tahap tinggi amaran keselamatan, satu untuk tahap sederhana, satu untuk tahap rendah dan prioriti bermaklumat. Oleh itu kajian ini melaporkan sumbangan kerangka ini meliputi kelonggaran atau *vulnerability* keselamatan perisian yang akan dibangunkan di masa akan datang.

ABSTRACT

Software developers generally focused on the core's functions and features, but the security was only addressed as an afterthought even though it was too late. The lack of proper consideration of security requirements during the early stages may lead to the development of an application with a poor security and the cost of correcting it might increase several times with every additional developmental phase. Assessing security at an early stage helps to design a secure application that can withstand malicious attacks. Therefore, software and system developers need practical and systematic approaches to obtain sufficient and credible evidence of the security level in the system, which is under development in the early phases of software development life-cycle (SDLC). Currently, there is limited number of reliable technique or method to quantify security requirements in software industry. Thus, the objective of the study is to construct a framework to elicit and quantify security requirements in order to ensure secure software been developed. Here, the work introduce a framework called Secure Appreciative Inquiry Fuzzy Quantification Technique (SAIFQT) which integrate Appreciative Inquiry, SQUARE, CLASP and Fuzzy Soft Set Theory techniques in eliciting and quantifying security requirements. The proposed framework, SAIFQT was evaluated with by case studies, penetration testing and validated by security experts. A mixed methodology was used in this study, a qualitative and explorative method. The results show that the proposed technique, SAIFQT was proved successfully in eliciting a new and unique software and security requirements specification. The results show the strong points in the proposed technique comparing to the normal SDLC, according to penetration testing reports, which shows three low priority alerts for the proposed technique. Meanwhile the report related to the prototype built using normal SDLC shows four high priority alerts and one for each of medium, low and informational priority security alerts. Thus, this study registered their contribution to cover security vulnerability in software intended to be build in future.

ملخص

مطورين البرمجيات يركزون بشكل عام على المتطلبات الأساسية للنظام، ولكن ما يخص المسائل الأمنية تضاف في وقت متأخر للنظام. أن عدم الاهتمام بأضافة المتطلبات الأمنية في مراحل مبكرة في تصنيع النظام يؤدي إلى تسليم نظام ضعيف ومليء بالثغرات يكلف تصليحها مستقبلاً أضعاف مضاعفة من تكاليف البرنامج الأصلية. توظيف العناصر الأمنية في مراحل مبكرة يزود المستفيدين بنظام آمن ومتصدي للهجمات. لذلك فإن مطوري البرمجيات بحاجة ماسة إلى أدوات وطرق تعزز وتمنح حالة ومستوى الأمن عند صناعة البرمجيات وخاصة في مراحل مبكرة من دورة حياة البرنامج (SDLC). في الوقت الحالي هناك عدد محدود من التقنيات والطرق التي يتم الاعتماد عليها في تزويد كم هو النظام آمن قبل أن يكتمل بنائه. لذلك تهدف هذه الدراسة إلى بناء تقنية تقوم بتزويد المتطلبات الأمنية المتعلقة بالنظام لظمان تزويد المستخدمين بأنظمة آمنة. ومن هنا فإن الدراسة تزود تقنية تدعى "تقنية التقدير الكمي الضبابي للأمن" (SAIFQT). لقد تم تقييم التقنية المقترحة باستخدام حالات دراسة وتقنية الإحتراق وتم التأكد من صحة التقنية المقترحة بواسطة خبراء الأمن. منهجية الجودة والطريقة الاستعراضية استخدمت في هذه الدراسة. أظهرت النتائج ان التقنية المقترحة (SAIFQT) قد أثبتت نجاحتها في استنباط متطلبات أمنية جديدة وفريد من نوعها إلى جانب ذلك استخلاص المتطلبات الأمنية لأي نظام منوي إنشائه كميأ وقبل بنائه وتسليمه وليس بعد وذلك باستخدام القليل من دراسات الحالة. أظهرت النتائج النقاط القوية في التقنية المقترحة بالمقارنة مع (SDLC) العادية، والتي غطت الكثير من الثغرات الأمنية التي لم تستطع (SDLC) العادية تغطيتها بناء على تقارير الإحتراق التي أظهرت ثلاثة تنبيهات منخفضة الأولوية للتقنية المقترحة في حين أن التقارير أظهرت أربعة تنبيهات عالية الأولوية وواحد لكل من متوسط ومنخفض ومعلوماتي التنبيه فيما يتعلق بالنموذج المبني الذي صمم باستخدام (SDLC) العادية. لذلك سجلت هذه الدراسة مساهمتها العلمية لتغطية أي ثغرات أمنية في أي نظام مرجو بنائه مستقبلاً.

TABLE OF CONTENTS

AUTHOR DECLARATION	I
BIODATA OF AUTHOR.....	II
ACKNOWLEDGEMENT	III
ABSTRAK.....	IV
ABSTRACT.....	V
ملخص.....	VI
TABLE OF CONTENTS	VII
LIST OF TABLES	X
LIST OF FIGURES	XI
LIST OF ACRONYMS	XIII
CHAPTER I INTRODUCTION	1
1.1 Overview.....	1
1.2 Background.....	2
1.3 Problem Statement.....	3
1.4 Research Questions.....	5
1.5 Research Objective	6
1.6 Significance of Study.....	6
1.7 Scope and Limitation of the Study.....	7
1.8 Organization of the Thesis.....	7
1.9 Summary.....	9
CHAPTER II LITERATURE REVIEW.....	10
2.1 Introduction.....	10
2.2 Functional and Non-Functional Requirements	11
2.3 Requirements Significance	12
2.4 Requirements Engineering.....	13
2.5 Requirements Elicitation.....	14
2.6 Various Requirements Elicitation Techniques	16
2.7 Comparison between Categories of Requirements Elicitation Techniques... 23	
2.8 Why Appreciative Inquiry (AI)?.....	27
2.9 Quantifying Software Requirements	30
2.10 Related Issues in Security Requirements Engineering	33
2.11 Methods and Best Practices in Security Requirements Engineering.....	36
2.12 Quantifying Security Requirements in Software.....	54
2.13 Requirements Elicitation Technique for Security Requirement	58

2.14	Exhibiting the Problems of Current Frameworks and Methods Related to Security Requirements.....	62
2.15	Fuzzy Soft Set Theory	64
2.16	Research Gap	69
2.17	Summary	70
CHAPTER III RESEARCH METHODOLOGY		71
3.1	Introduction.....	71
3.2	Validation of Proposed Integrated Technique (SAIFQT).....	79
3.3	Evaluation of Proposed Integrated Technique (SAIFQT)	81
3.4	Summary.....	83
CHAPTER IV ELICITING SECURITY REQUIREMENTS USING THE PROPOSED SECURE APPRECIATIVE INQUIRY TECHNIQUE		84
4.1	Introduction.....	84
4.2	Benefits of Integrating Security Requirements into Software Requirements	84
4.3	Secure Appreciative Inquiry Technique: Embedding Square, Class With AI	85
4.4	Summary	109
CHAPTER V QUANTIFYING SECURITY REQUIREMENTS USING APPRECIATIVE INQUIRY AND FUZZY THEORY		110
5.1	Introduction.....	110
5.2	Benefits of Integrating Algorithm of Fuzzy Soft Set Theory into SAIT	111
5.3	Embedding Fuzzy Soft Set Algorithm into SAIT: Design Phase	111
5.4	Proposed Quantifying Security Requirements Using Fuzzy Soft Set Theory	112
5.5	Summary	122
CHAPTER VI CASE STUDY: DEVELOPMENT OF ONLINE E-BUSINESS WEBSITE (PRICE) USING SAIFQT		123
6.1	Introduction.....	123
6.2	Discovery Phase.....	123
6.3	Dream Phase	128
6.4	Design Phase.....	130
6.5	Destiny Phase.....	159
6.6	Website Screens and Main Functions	162
6.7	Summary.....	165
CHAPTER VII DISCUSSION AND FINDINGS		166
7.1	Introduction.....	166
7.2	Evaluation and Validation Measures	166

7.3	Summary.....	173
CHAPTER VIII CONCLUSION AND FUTURE WORK.....		174
8.1	Introduction.....	174
8.2	Review of Research Background.....	174
8.3	Review of Research Methodology.....	176
8.4	Research Contribution	177
8.5	Research Limitations	179
8.6	Recommendations for Future Research.....	180
REFERENCES.....		181
APPENDIX A PILOT STUDY: APPRECIATIVE INQUIRY TECHNIQUE.....		198
APPENDIX B PILOT STUDY: ISLAMTAG SOCIAL NETWORK.....		204
APPENDIX C SEQUENCE DIAGRAM AND COLLABORATIVE DIAGRAM FOR REAL CASE STUDY: ONLINE E-BUSINESS WEBSITE (PRICE).		218
APPENDIX D SECURITY EXPERTS REPORTS.....		226
APPENDIX E PENETRATION TEST FOR THE TWO PROTOTYPES.....		237

LIST OF TABLES

TABLE 1: Conversational methods for requirements elicitation	18
TABLE 2: Observational methods for requirements elicitation	19
TABLE 3: Analytic methods for requirements elicitation	20
TABLE 4: Synthetic methods for requirements elicitation	22
TABLE 5: Comparison between various types of elicitation techniques	24
TABLE 6: SQUARE Process	38
TABLE 7: CLASP activities, related project roles, and best practices	48
TABLE 8: Comparison between methods and best practices in security requirements engineering	54
TABLE 9: Some studies about quantifying security in software	56
TABLE 10: Proposed discovery phase for SAIT	86
TABLE 11: Proposed dream phase for SAIT	88
TABLE 12: Proposed design phase for SAIT	90
TABLE 13: Proposed destiny phase for SAIT	95
TABLE 14: Complete phases for proposed secure appreciative inquiry technique (SAIT)	97
TABLE 15: Summarizing phases of SAIT based on concern of the security issue ..	107
TABLE 16: Vulnerabilities, Errors and Security index	113
TABLE 17: Summarizing phases of proposed Secure Appreciative Inquiry Fuzzy Quantification Technique (SAIFQT)	119
TABLE 18: The functional requirements of online e-business website	134
TABLE 19: Software requirements and related security requirements	144
TABLE 20: Software requirements and related security requirements value	161
TABLE 21: Summary of experts' feedback	170
TABLE 22: Summary of study	175
TABLE 23: Ignored steps from SQUARE and CLASP in integrating process with AI	179
TABLE 24: Techniques comparison	202

LIST OF FIGURES

FIGURE 1: Software Timeline.	7
FIGURE 2: Requirements Engineering Phases Surveyed.	14
FIGURE 3: Core process (iterative 4-D cycle).	29
FIGURE 4: CLASP views and their interactions.	44
FIGURE 5: Use case diagram containing misusers and misuse cases.	51
FIGURE 6: Research Framework.	73
FIGURE 7: Theoretical Study and Defining the Security Requirement Elicitation Problem.	74
FIGURE 8: Integrating AI with SQUARE and CLASP.	76
FIGURE 9: Integrate SAIT with Fuzzy Algorithm.	77
FIGURE 10: Conduct a Real Test Case and Build Prototypes.	78
FIGURE 11: Validation and Evaluation.	79
FIGURE 12: Research Process.	82
FIGURE 13: Discovery Phase of SAIT.	100
FIGURE 14: Dream Phase of SAIT.	102
FIGURE 15: Design Phase of SAIT.	103
FIGURE 16: Destiny Phase of SAIT.	106
FIGURE 17: Proposed Secure Appreciative Inquiry Technique (SAIT) Process.	108
FIGURE 18: Proposed Secure Appreciative Inquiry Fuzzy Quantification Technique (SAIFQT) Process.	121
FIGURE 19: Use case for Online E-Business Website.	135
FIGURE 20: Misuse Case for Online E-Business Website.	135
FIGURE 21: Class diagram for customer and attacker.	136
FIGURE 22: Main Page.	162
FIGURE 23: Registration Page.	162
FIGURE 24: Login Page.	163
FIGURE 25: Customer Page.	163
FIGURE 26: Post Services Page.	164
FIGURE 27: Payment Page.	164
FIGURE 28: Vulnerabilities Alerts for Normal SDLC Prototype.	172
FIGURE 29: Vulnerabilities Alerts for SAIFQT Prototype.	172
FIGURE 30: Sequence diagram of login use case.	219
FIGURE 31: Collaborative diagram of login use case.	219
FIGURE 32: Sequence diagram of make registration use case.	220
FIGURE 33: Collaborative diagram of make registration use case.	220
FIGURE 34: Sequence diagram of purchase services use case.	221
FIGURE 35: Collaborative diagram of purchase services use case.	221
FIGURE 36: Sequence diagram of make payment use case.	222
FIGURE 37: Collaborative diagram of make payment use case.	222
FIGURE 38: Sequence diagram of brute force login use case.	223
FIGURE 39: Collaborative diagram of brute force use case.	223

FIGURE 40: Sequence diagram of malicious code injection use case.....	224
FIGURE 41: Collaborative diagram of malicious code injection use case.....	224
FIGURE 42: Sequence diagram of disclose user information use case.....	225
FIGURE 43: Collaborative diagram of disclose user information use case.....	225



LIST OF ACRONYMS

AI	Appreciative Inquiry
CIA	Confidentiality, Integrity and Availability
CLASP	Comprehensive, Lightweight Application Security Process
CREE	Confidentiality Requirements Elicitation and Engineering
EI	Errors Index
MSRA	Multilateral Security Requirements Analysis
RE	Requirements Engineering
SAIFQT	Secure Appreciative Inquiry Fuzzy Quantification Technique
SAIT	Secure Appreciative Inquiry Technique
SDLC	Software Development Life Cycle
SE	Software Engineering
SI	Security Index
SQUARE	Security Quality Requirements Engineering
SR	Security Requirements
SRE	Security Requirements Engineering
SRI	Security Requirements Index
STEP	Software Test and Evaluation Panel
UML	Unified Modeling Language
VI	Vulnerabilities Index
XP	Extreme Programming