International Journal of Informatics and Communication Technology (IJ-ICT)
Vol.1, No.2, December 2012, pp. 72~81
ISSN: 2252-8776 ❒ 72

Measuring Security in Requirements Engineering

Souhaib Besrour*, **Dr. Imran Ghani****
* Department of software engineering, UTM University, Johor, Malaysia
** Department of software engineering, UTM University, Johor, Malaysia

Article Info

Article history:

Received Jun 25th, 2012
Revised Aug 20th, 2012
Accepted Sept 23th, 2012

Keyword:

Requirement Engineering
Requirement Quality
Measuring Security
Software equipment

ABSTRACT

The aim of this paper is to measure the security and related verification method in requirements engineering (RE). There are a few existing approaches to measure RE performance like IEEE Software Requirements Specification (SRS) and Security Quality Requirements Engineering (SQUARE). However, these existing approaches have some limitations such as lack of flexibility and require long implementation period. In order to address these issues, this paper intends to propose a new set of tools. First is the Effective Security Check List (ESCL), which is a check list with security questions that should be considered for measuring security. Secondly, the Traceability Matrix(TM), which is a two dimensional matrix to measure security during RE. Thirdly, Requirement Engineering Assessment Document (READ), which is a tool containing all statistical information about security performance during RE. The combination of presented approaches had been implemented in a case study. The outcome results are encouraging and illustrated integrated outcomes within existing RE model. The security level had also been properly measured.

Corresponding Author:
Souhaib Besrour
Department of software engineering
UTM university Johor, Malaysia
Email: utm142@yahoo.com

1. INTRODUCTION

Requirements engineering is a field of software engineering that uses clear techniques and procedures to produce a high quality requirement [7]. Nowadays, RE [21], [22], [23] is widely used in software companies to reduce RE failure probability. However there are still issues that have not obtained much attention, for example security requirement. Security in RE is considered one of the main important attributes to build a strong system. Increased problems in RE have caused many errors during design, implementation and testing stages that further lead to vulnerable software. Thus, this paper focuses on the measurement of security in RE and the related quality verification methods. RE has many different processes during its life cycle. Based on previous research [1], RE typically has four main processes, i.e. elicitation, analysis, verification, and management as shown in (Figure 1).

The rest of paper is organized as follow: section 2 presents a brief background about requirements engineering. Section 3 presents few issues in previous approaches, advantages and limitations. Section 4 outlines the presents proposed model with all details about model structure. Section 5 presents a simple case study has been conducted in UTM University to verify the effectivity of presents model. Section 6 presents few discussions about present’s techniques. Section 8 outline limitations as any research based work there is no perfect research in this section a brief descriptions a bout same points could be considered as limitations. Section 8 presets future work and what to be done to make enhancement in RE techniques .Section 9 conclude the paper.

2. BACKGROUND OF REQUIREMENT ENGINEERING (RE)

A more precise definition of RE is “a systems and software engineering process which covers all of the activities involved in discovering, documenting and maintaining a set of requirements for a computer-based system”[13].There are many other definitions, but a common agreement is that RE is a sub discipline of systems and software engineering and is concerned with establishing the goals, functions and constraints of hardware and software systems[14][15].The term RE first appeared in 1979 in a TRW technical report [16], but it did not come into general use until the 1990s with the publication of an IEEE Computer Society tutorial [17] and the establishment of a conference series on RE. In the traditional waterfall model of the systems or software engineering, [9]10[11] [12] [18] RE is presented as the first stage of the development process with the outcome being a requirements document or software requirements specification. In fact, RE is a process that continues throughout the lifetime of a system as the requirements are subject to change and new requirements must be elicited and documented and existing requirements have to be managed over the entire lifetime of the system. For example, the project [19] maintains an extensive bibliography of RE. The sub-processes that are parts of a general RE process vary widely depending on the type of system being developed and the specific practice of the organization developing the requirements[20]. Activities within the RE process may include: requirements elicitation, requirements analysis, requirements verification, and requirements management.

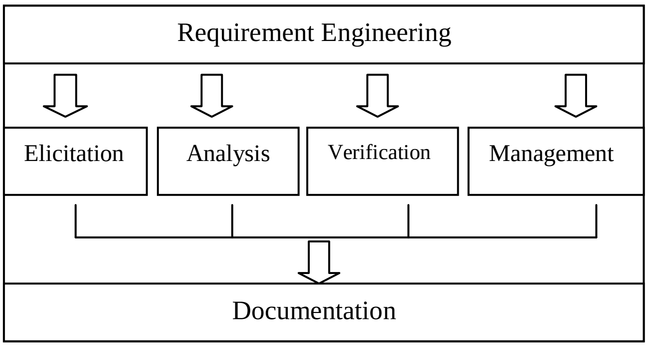

Figure1. Existing RE processes

3. ISSUES IN PREVIOUS APPROACHES

There had been few existing approaches to measure RE performance like IEEE Software Requirements Specification (SRS) and Security Quality Requirements Engineering (SQUARE). However, these existing approaches have their limitations.

Table1. Comparison between different RE techniques

Tool Name	Main Features	Limitations
SRS	Preserve integrity strong requirement building	take long time to implement
SQUARE	processing speed security strength makes requirement analyses easier	lack of flexibility

As shown is (Table 1) the main strength of SRS over other models is its ability to preserve integrity, build strong requirement and provide detail software requirement specification. SRS reduces failure opportunities and cover most software requirements and constraints. Activities are organized in a clear way [3]. Nevertheless, SRS is time consuming; it takes longer time to be implemented than other techniques. The SQUARE model, on the other hand, is superior in processing speed and security strength [4]. Another advantage of SQUARE model is that it makes requirement analyses and documentation much easier. Even so, it lacks flexibility.

4. PROPOSED MODEL

In order to measure security quality, a set of approaches had been proposed in this paper. These approaches will be explained individually to clarify the function of each tool. In this work, a new phase called "security" was added into the RE process as shown in (Figure 2). Its basic role is to enhance and increase security performance and is consisted of three main tools.

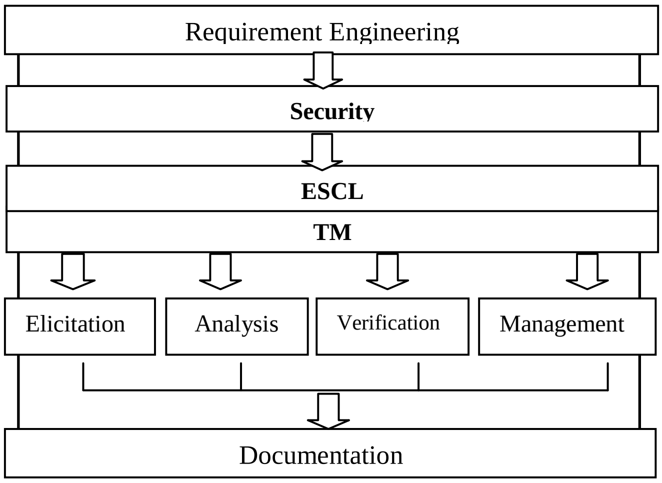

Figure2. Proposed RE model

4.1. Effective security check list (ESCL)

This section presents the design overview of ESCL. It contains a set of security question as shown in (Table2).The security questions have to be chosen and categorized in a systematic way. ESCL provides a better security measurement where the list is based on previous RE studies and techniques [1], [2], [5] with more enhancements on security strength. The ESCL is used during the four processes of RE as mentioned earlier (elicitation, analyses, validation, management).

Table2. ESCL during requirements elicitation

	Level of security				
Security Questions	**UN**	**VL**	**L**	**A**	**H**
1-Have you using any feasibility study before the beginning of new project.					
2-Have you use secure prototype for non-understood requirement.					
3-Have you use secure scenario before the beginning of elicit requirement.					
Official use	overall level of security				
	number of security questions				
	average level of security				

UN= unknown=0 , VL=very low=1, L=Low=2,
A=average=3, H=high=4

As shown in(Table2), all listed security questions need to answer properly according to its level of security into unknown (UN), level 0; very low (VL), level 1; low (L), level 2; average (A), level 3; and high (H), level 4. To start, all questions must be read and understood before choosing the appropriate required security level. In the column of" Official use", all item numbers have to be counted and inserted into the appropriate column. This technique is repeated during all RE processes.

4.2. Traceability Matrix(TM)

The basic role of Traceability Matrix (TM) is to confirm confidentiality, integrity, availability, and complexity of risk assessment. TM has a two-dimensional table with many rows and columns. Security questions are listed in rows and the corresponding security items are listed in columns, as shown in (Table 3). The TM can also be used during all RE processes.TM works with high-level requirement [6] and can be used to measure security in RE.

Table3. TM during requirements elicitation

Re. Tractability Matrix.		1. Have you using any feasibility study before the beginning of new project.	2. Have you use secure prototype for non-understood requirement.	3. Have you use secure scenario before the beginning of elicit req.
Times Security Questions been tested	→			
Times Security Items been tested	↓			
1.0Confidentiality				
2.0 Integrity				
3.0 Availability				
4.0 Complexity				
5.0risk assessment				
For official use	total times security questions been tested			
	non tested security questions			

As described earlier, (Table 3) shows that TM is consisted of many rows and column. To illustrate the usage of TM, in the first column that stated "time security questions have been tested", values should be inserted horizontally. In the subsequent security items (confidentiality, integrity, availability, complexity, of risk assessment), the values have to be written vertically. This column shows how many times the security items have been tested. On the last column "Official use", all numbers have to be counted.

4.3. Requirements engineering assessment document (READ)

Requirements Engineering Assessment Document (READ) provides all statistical results of measuring security practices. As shown in (figure3) .By using READ, the security level can be assessed based on its statistical results. READ is a visualized tool used to assess overall project efficiency and security level.

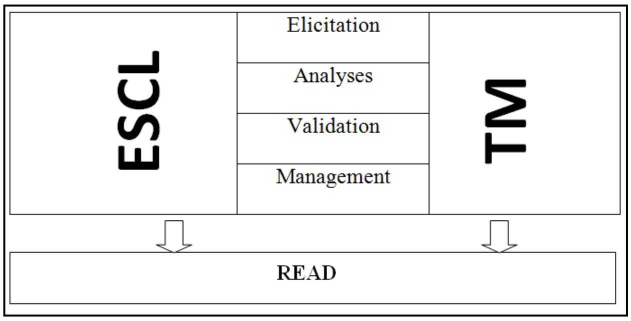

Figure3. READ big picture.

Data used in READ is collected from ESCL and TM. It presents a detailed assessment of all RE assessment, thus contains all data analyses and collected data in all phases. Visualized results are important to convert the tables, numbers and statistics into easily readable format. The following (Table 4) describes the READ items and its contents. The values are just samples to explain and clarify READ techniques and how they work.

Table4. Percentage of highest/ lowest security level questions in ESCL

RE Process	Highest Security Level Questions	Lowest Security Level Questions
Requirements elicitation	40% (Q2,Q3)	40% (Q1,Q4)
Requirements analysis	40% (Q1,Q3)	60% (Q4,Q5,Q6)
Requirements validation	60% (Q1,Q4,Q5)	20% (Q2)
Requirements Management	40% (Q7,Q9)	60% (Q1,Q3,Q4)

As shown is (Table 4) the percentage of highest/lowest security level is presented. The security question is presented by the letter “Q” means the question number. The first column “RE process” represents the elicitation, analysis, verification, and management processes. The second column (highest security level questions) presents the percentage of highest security level questions. To calculate the percentage of highest security level questions in ESCL, all highest security level questions in ESCL must first be counted and then multiplied with the total number of question. The last column shows the percentage of lowest security level questions in ESCL. This means that these security questions have a low security level and thus need to be enhanced. As shown in (Figure 4). All figure values are brought from (Table 4).

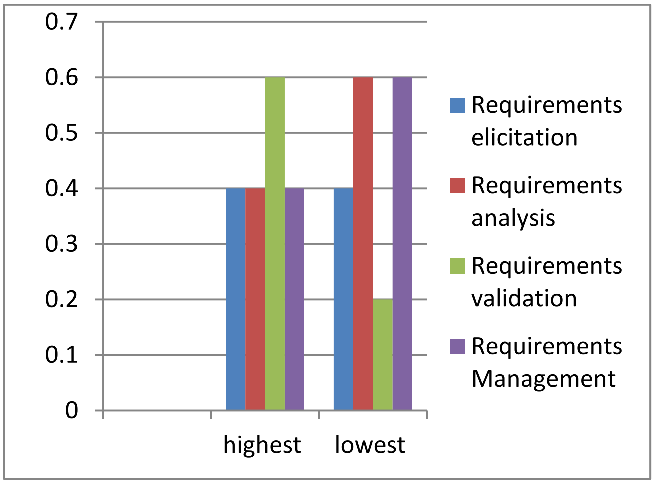

Figure4. Percentage of highest/ lowest security level questions in ESCL

5. CASE STUDY

This study had been conducted using ESCL, TM and READ at the Centre of Information and Communication Technology (CICT), University Technology Malaysia (UTM).

5.1. Implementation of ESCL

The ESCL in this study was designed as shown in (Table 5).The questions were chosen and categorized in a systematic way.

Table5. ESCL during requirements elicitation

	Level of Security				
Security Questions	**UN**	**VL**	**L**	**A**	**H**
1-Have you using any feasibility study before the beginning of new project.		✓			
2-Have you use secure prototype for non understood requirement.					✓
3-Have you use secure scenario before the beginning of elicit requirement.				✓	
Official use	Overall level of security				8
	Number of security questions				3
	Average level of security				2.6

UN= unknown=0 , VL=very low=1, L=Low=2, A=average=3, H=high=4

5.2. Implementation of TM

The implementation of TM is as shown in (Table 6).

Table6. TMduring requirements elicitation

Re. Tractability Matrix.		1. Have you using any feasibility study before the beginning of new project.	2. Have you use secure prototype for non understood requirement.	3. Have you use secure scenario before the beginning of elicit req.
times security questions been tested	→	2	4	2
timessecurity items been tested	↓			
1.0confidentiality	3	✓	✓	✓
2.0 integrity	0			
3.0 availability	2		✓	✓
4.0 complexity	1		✓	
5.0risk assessment	2	✓	✓	
for official use	total times security questions been tested			8
	non tested security questions			0

UN= unknown=0, VL=very low=1, L=Low=2, A=average=3, H=high=4

To illustrate the usage of TM, Question 3 will be taken as an example. This question has a value of 2 in the column stated “Times Security Questions had been tested”. This corresponded to the number of question item being tested, i.e. confidentiality and availability. In the “confidentiality” column, the number 3 had been written, signifying that the security item had been checked three times using different security questions. Similarly, for “availability”, the security item had been verified twice using different security questions as well. In the last column stated “Official Use”, all numbers had been summarized.

5.3. Implementation of READ

The implementation of READ is already done as shown in (Table 7). Explanation on the usage of READ has been provided in Section 4.3.

Table7. Percentage of highest/ lowest security level questions in ESCL

RE Process	Highest Security Level Questions	Lowest Security Level Questions
Requirements elicitation	20% (Q2)	60% (Q1,Q4,Q5)
Requirements analysis	40% (Q1,Q3)	40% (Q4,Q5)
Requirements validation	60% (Q1,Q4,Q5)	20% (Q2)
Requirements Management	60% (Q7,Q9,Q10)	20% (Q1)

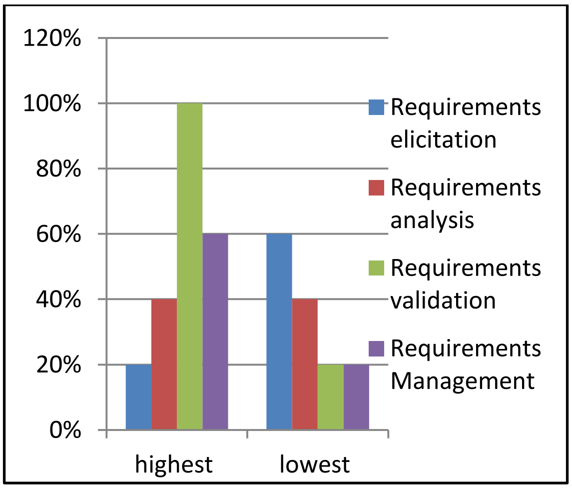

Figure5. Percentage of highest/ lowest security level questions in ESCL

The (Figure 5) presents percentage of highest/ lowest security level questions in ESCL. All figure values were brought from (Table 8).

As shown in (Table 8) the percentage of tested/untested security questions in TM. Similar to that in ESCL, the letter “Q” represents the question number, “RE process” signifies the four RE processes, and “Tested Sec. Questions” represents the percentage of tested requirements in TM. The high values of tested items are considered a technical advantage and have to be tested and organized properly. The “Untested Sec. Questions” column lists down the percentage of all untested requirements in TM and also need to be organized and tested properly. These are the limitations of the company as they can cause technical weaknesses and add more ambiguity to the project. The information in (Table 8) is graphically presented in (Figure 6).

Table8. Percentage of tested /untested security questions in TM.

RE Process	Tested Sec. Questions	None Tested Sec. Questions
Requirements elicitation	20%	80% (Q1,Q4,Q5)
Requirements analysis	30%	70% (Q4,Q5)
Requirements validation	30%	70% (Q2)
Requirements Management	20%	80% (Q1)

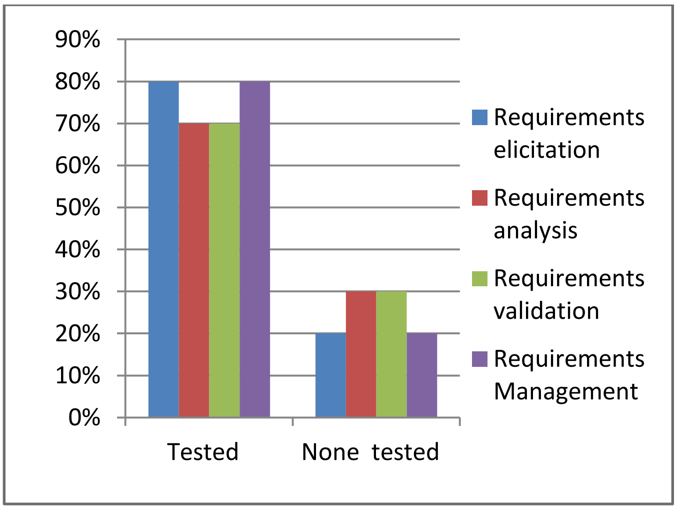

Figure6. Percentage of tested /untested security questions in TM

Table9. Classification of Security Level using ESCL

Requirements elicitation; B
Requirements analysis; B
Requirements validation; A
Requirements management; C

Security Level are classified as shown in (Table 9) using ESCL there is three level of security A,B and C. The data in (Table 9) is graphically presented in (Figure 7). This security classification is an efficient technique to give a clear classification of security level.

The letter' A' ranges from level 5 to 4
The letter 'B' from level 4 to 2
The letter 'C' from level 2 to zero

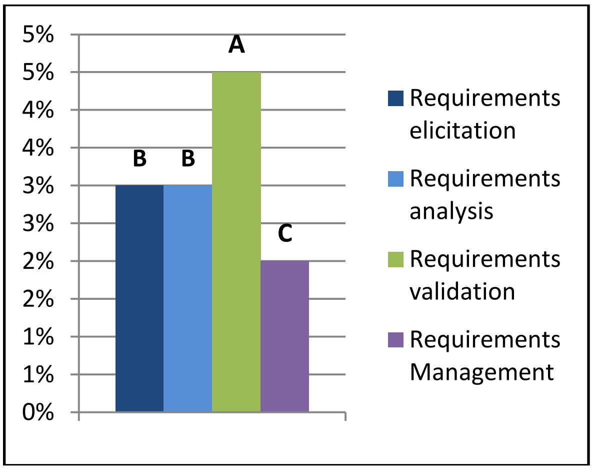

Figure7. Classification of Security Level using ESCL

6. DISCUSSION

The main target of this paper was to measure the security in RE and how the quality of the security can be verified using ESCL, TM and READ.ESCL had been successfully tested within the four RE processes. In regard to TM, the results confirmed its basic role to verify the confidentiality, integrity, availability, and complexity of risk assessment. TM is very sensitive with numbers where all values have equally important weight in the analysis process. The TM used in this study had high level requirement. Results also indicated that, for READ, clear RE assessment could be made to assess the efficiency and security level of the project. The main difference among ESCL, TM and READ is that READ is more focused on the summary of project security level where detailed assessment can be provided. The major strong point of all tools, on the other hand, is that they give satisfying accuracy in security assessment, are easy to use and are able to provide good security enhancements. A combination of all tools will improve the overall security level of the software project under scrutiny.

7. LIMITATIONS OF THIS WORK

The limitation of this work is that the work was done at CICT, UTM in a small scale. For larger IT facilities, the results obtained from this study may not be that accurate and reliable. Although a combination of ESCL, TM and READ is good for such small scale study, it may slow down the assessment if used in bigger IT area.

8. FUTURE WORK

In future works, measuring security in RE has to be more specific and efficient using new mathematical algorithm and better methods enhanced from previous researches[24][25][26] [27]. Exponential advancement in the IT industry has made security threat an increasingly challenging problem. Thus, RE tools have to be more efficient to preserve the confidentiality, integrity, availability and complexity of risk assessment for a software project. This is important throughout the entire life cycle. Other than that, since READ reports have been found to give more details and statistics, more statistical components and different mathematical techniques are expected to be evaluated.

9. CONCLUSIONS

This paper had contributed to the measurement of security in RE using three basic techniques ESCL, TM and READ in the elicitation, analysis, validation, and management processes. The combination of these three presented tools can give a good template to measure security practices.

REFERENCES

[1] Ian Sommerville .2010. Software engineering 9th Edition.
[2] Charles B. Haley, Robin Laney, Jonathan D. Moffett, and Bashar.2008.Security Requirements Engineering: A Framework for Representation and Analysis.
[3] IEEE Recommended Practice for Software Requirements Specifications. 2009. IEEE Computer Society Sponsored by the Software Engineering Standards Committee.
[4] Nancy R. Mead and Ted Stehney. 2005 .Security Quality Requirements Engineering (SQUARE) Methodology by Software Engineering Institute.

[5] Stuart Anderson and Massimo Felici Laboratory. 2001. Requirements Engineering questionair, University of Edinburgh James, 3JZ Scotland, UK.
[6] Carlos, Tom. 2008 .Requirements Traceability Matrix RTM. PM Hut.
[7] Chichester Kotonya, G. and Ian Sommerville. 1998. Requirements Engineering: Processes and Techniques. UK.
[8] SWEBOK, Alain Abran, James. 2004. Guide to the Software Engineering Body of Knowledge. IEEE Computer Society. pp. 1–1. ISBN 0-7695-2330-7
[9] ACM (2006). "Computing Degrees & Careers". ACM.
[10] Laplante, Phillip .2007. What Every Engineer Should Know about Software Engineering. Boca Raton: CRC. ISBN 9780849372285.
[11] Peter, Naur; Brian Randell.2008."Software Engineering: Report of a conference sponsored by the NATO Science Committee" Garmisch, Germany: Scientific Affairs Division, NATO.
[12] Randell, Brian. 2008."The 1968/69 NATO Software Engineering Reports". Newcastle University.
[13] Kotonya G. and Sommerville. 1998. Requirements Engineering: Processes and Techniques. Chichester, UK.
[14] Phillip A. Laplante. 2007. What Every Engineer Should Know about Software Engineering.
[15] Zave, P. 1997. 'Classification of Research Efforts in Requirements Engineering'. ACM Computing Surveys.
[16] Alfor, M. W. and Lawson. 1979. Software Requirements Engineering Methodology (Development) Defense and Space Systems Group.
[17] Thayer, R.H., and M. Dorfman. 1990. System and Software Requirements Engineering, IEEE Computer Society Press, Los Alamitos.
[18] Royce. 1970.'Managing the Development of Large Software Systems: Concepts and Techniques', IEEE Westcon, international conference on Software Engineering.
[19] Requirements bibliography Reviewed November 10th 2011.
[20] Sommerville. 2006. Software Engineering, 7th ed. Harlow, UK: Addison Wesley.
[21] Onkar Nath Pandey. 2012. A Technical Discussion on Requirement Engineering with an Example, Journal of Global Research in Computer Science.
[22] Saima Amber, Narmeen Shawoo, Saira Begum. 2012. Determination of Risk during Requirement Engineering Process.
[23] Ashish Sharma, Dharmender Singh Kushwaha. 2011. Natural Language based Component Extraction from Requirement Engineering Document and its Complexity Analysis.
[24] Daniel Rodriguez, Israel Herraiz and Rachel Harrison. 2012. on Software Engineering Repositories and Their Open Problems.
[25] Leif Singer and Kurt Schneider. 2012. Influencing the Adoption of Software Engineering Methods Using Social Software, Hannover, Germany.
[26] Polala Niranjan Reddy, Kukatlapalli Pradeep Kumar.2010.An Efficient Software Engineering Ontology Tool for Knowledge Sharing.
[27] Rashmi Yadav, Ravindra Patel and Abhay Kothari.2011.Software Engineering for Practiced Software Enhancement.

BIOGRAPHY OF AUTHORS

First Author Souhaib Besrour is a master student at Faculty of Computer Science and Information Systems, University Technology Malaysia (UTM), Johor Campus. He received his Master of Computer Science from UTM (Malaysia). His research focus includes studying security in software engineering and measuring security in requirement engineering.

Second Author Dr. Imran Ghani is a Senior Lecturer at Faculty of Computer Science and Information Systems, University Technology Malaysia (UTM), Johor Campus. He received his Master of Information Technology Degree from UAAR (Pakistan), M.Sc Computer Science from UTM (Malaysia) and Ph.D. from Kookmin University (South Korea). He has been working internationally in IT industry (Pakistan, Dubai and Malaysia) and universities (South Korea and Malaysia). His research focus includes studying semantics techniques, content-based, collaborative filtering techniques, semantic web services, semantics-based software testing, and security in agile software development practices, enterprise architecture and software architecture. Currently, he is accepting Ph.D. and Masters Students who wish to study at UTM University.