CHAPTER VIII

CONCLUSION AND FUTURE WORK

8.1 Introduction

This chapter synthesizes the findings of the research. It reviews the research background and objectives, it also summarizes the research methodology, research contribution, and research limitations. The chapter concludes with recommendations for future researches, which may be pursued based on the work reported here.

8.2 Review of Research Background

In this study, the main aim is to develop a technique for enhancing software development's life cycle. In particular, it seeks to investigate the security issues arising in the requirement phase. Moreover, the process of elicitation and quantification for security requirements needs to be achieved in the requirement phase of SDLC. To facilitate understanding, the main issues those are summarized in Table 22.

TABLE 22: Summary of study

Problem 1	*Problem 2*	*Problem 3*
Software developers generally focused on core functionality and features, but the security was typically only addressed as an afterthought and too late	There is no reliable concrete technique or method to quantify security requirements in software industry (SDLC artifact). Very little work has been done to help the developers in this direction	It is essential for the developer to know, in early phases of developing any software, how many vulnerabilities are present, what is the potential damage vulnerabilities can cause to various assets of the system the software is going to be a part of and what security requirements have to be incorporated to remove these vulnerabilities
Objective 1	*Objective 2*	*Objective 3*
To propose a Secure Appreciative Inquiry Technique (SAIT) for eliciting security requirements. SAIT is SQUARE, CLASP and AI	To propose Secure Appreciative Inquiry Fuzzy Quantification Technique (SAIFQT) for measuring security requirements using fuzzy soft set theory	To evaluate (SAIFQT) using real case study with penetration testing and validation by security experts on a built prototype
Research question	*Research question*	*Research question*
Q1 & Q2	Q3	Q4
Methodology	*Methodology*	*Methodology*
Undertook a literature review, proposed integrated technique to elicit SR and create a conceptual framework	Use fuzzy soft set theory algorithm to quantifying security requirements, proposed integrated technique to elicit and quantify SR and create a conceptual framework	Conduct a real case study, build a prototype according to the proposed integrated technique, making a penetration testing to the prototype and interview software and security experts
Result	*Result*	*Result*
- Elicit software and security requirements	- Elicit software and security requirements - Quantify security requirements - Quantifying Security Requirements Before Building the System	- Build a secure prototype - The proposed technique help developers to capture security requirements - Succeed integration between software and security approaches

8.3 Review of Research Methodology

This study follows the mix method, which depicts the research process that is undertaken to achieve the research objectives. The process of conducting research should be designed systematically to satisfy the requirements for solving a particular problem. Therefore, the research process is adopted to guide the conducted research.

At stage 1, a literature review of relevant research includes the review and analyses of the requirements elicitation techniques, security requirements elicitation techniques, appreciative inquiry approach (AI) and the fuzzy soft set theory. The review is on the use of appreciative inquiry in eliciting software requirements. At this stage, we study the security requirements, quantify the security requirements in conventional software developments, study the fuzzy soft set theory and identify the problem's statement.

At stage 2, a pilot study is carried out for the appreciative inquiry approach to prove its ability and efficiency in eliciting the normal requirements then, the appreciative inquiry approach is mapped with security requirements elicitation (SQUARE & CLASP). The integrated technique (SAIT) is proposed and finally a conceptual framework is created.

At stage 3, a pilot study for fuzzy soft set algorithm (CCEA) is conducted to investigate its ability and efficiency in quantifying security requirements then, the algorithm is adapted and embedded with the proposed integrated technique SAIT, which is developed in stage 2, to calculate and quantify the security requirements, as well as to create a conceptual framework for the proposed technique (SAIFQT).

In stage 4, a real case is conducted to use the proposed integrated security requirements elicitation technique SAIFQT to elicit software and security requirements and, quantify security requirements in this real case.

The final stage (stage 5) focuses on evaluating the functional reliability and validity for the proposed integrated technique (elicit and quantify security requirements) by the pilot study which uses the IslamTag Website. The proposed integrated technique SAIFQT is reviewed by software experts, security experts and, practitioners. The penetration test is done for two prototypes, the first one is built under the proposed integrated SAIFQT, and the second one built with the same business requirements and programming language, but using the normal SDLC, then the same penetration testing tool is used. The penetration test results for the two prototypes which are mentioned before are compared in order to validate the efficiency of the proposed technique and to register the contributions of this study.

8.4 Research Contribution

This study makes important contributions in the context of software engineering and secure software engineering. This research has made the following important contributions:

a) The first contribution of this research is related to understand security requirements, which cover the functional/business requirements. Moreover, this study highlights the importance of the security in the software industry, especially in the requirement phase. This in turn plays an important role in increasing the level of security of the software artifacts.

b) The second contribution lies in the fact that developing a technique helps each of the researchers, security experts and practitioners to know more about system vulnerabilities, which serve to be the main contribution of this research. This technique is considered to be more appropriate for developing secure software, as it has been developed after thorough investigation of the situation which is made in the current related techniques. This technique reflects the following aspects:

i) The study reveals that the integrated technique (SAIFQT) proves its ability to elicit unique and new security requirements and quantify them.

ii) Elicit functional/business requirements and security requirements at the same time.

c) The third contribution of this research is that important feedback is offered by security experts to validate the proposed technique. Based on the proposed prototype, the result was very satisfying, as revealed by the penetration testing. This research provides important recommendations with regards to the security requirements elicitation and quantification techniques that will help researchers, security experts and practitioners to:

i) Understand security requirements.

ii) Understand the influence of security factor on the SDLC, especially in the requirement phase.

iii) Provide high-quality and secure software by using the proposed technique (SAIFQT), which will help them to acquire new and unique secure software, and which will ultimately enable them to increase security and decrease software failure.

The integration is a modern phase at the methods of eliciting and quantifying security requirements. According to the literature review, using the fuzzy soft set theory to quantify security requirements has been conducted in this study for the first time. The study adds insights to developers and security experts to elicit business and security requirements and to quantify security requirements using one technique in one cycle.

The benefit of using the fuzzy soft set theory by applying the positive aspects of algorithm to quantifying security requirements is that it adds value to the requirements engineering by facilitating the process of quantification security requirements among the security experts and groups of developers. Contributors are motivated, because they feel that it is tangible to know more about software security, especially security requirements; which consequently and positively influence the software industry. It is noteworthy that, the success of any software development process depends on many

issues; one of the most important issues is the process of covering the security requirements (system vulnerabilities).

8.5 Research Limitations

Due to some reasons which are illustrated in the table 23 below, there are some processes or steps in the SQUARE method and CLASP best practices, which are not considered as part of the new techniques. Table 23 illustrates these ignored steps with some justification provided.

TABLE 23: Ignored steps from SQUARE and CLASP in integrating process with AI

Method	Ignored Steps	Reasons
SQUARE	Select elicitation technique	The main objective in the proposed method is elicit requirements (Elicitation Security Requirements Technique)
CLASP	Institute security awareness programs	Basically, instead of training developers to extract security requirements, experts will be used to extract security requirements Another reason, security awareness programs are held for users to avoid expected attacks after deploying the system, not during requirements phase
	Research and assess security posture of technology solutions	This step should be in the implementation phase, which is not included in the proposed technique (Elicitation Security Requirements Technique)
	Implement and elaborate resource policies and security technologies	This step should be in the implementation phase, which is not included in the proposed technique (Elicitation Security Requirements Technique)
	Implement interface contracts	This step should be done after finishing the requirement phase by showing the alpha interface to the stakeholders (type of prototype method) to agree and sign the contracts
	Perform code signing	This step should be in the implementation phase, which is not included in the proposed technique (Elicitation Security Requirements Technique)
	Specify database security configuration	This step should be in the implementation phase, which is not included in the proposed technique (Elicitation Security Requirements Technique)
	Build operational security guide	This step should be delivered to stakeholders after building the whole system (Deployment Phase)

As shown in table 23, there are some ignored steps of the SQUARE method and CLASP best practices, but it is clear that, the SQUARE method misses the proposed techniques by one step, which is "Select elicitation technique", that is compensated by

the proposed integrated techniques, which is "elicit security requirements" serving as one of the important priorities.

On the other hand, CLASP best practices have seven steps, which are different from the proposed techniques, because some steps should be in advanced stages of the SDLC, and others should be done after completing the software; CLASP has many steps of the SDLC not just only the requirement phase, but it also focuses more on the requirement phase.

8.6 Recommendations for Future Research

This study has successfully investigated the security requirements elicitation techniques and developed two techniques to elicit and quantify security requirements in the requirement phase. Moreover, this study has highlighted the influence of the vulnerabilities on the software in general. However, there is still a need to investigate all the security issues that influence the SDLC phases like the design and implementation phases. Therefore, to conclude, this study offers some suggestions for future research as follows:

1) Other techniques or methods could be employed in future studies, such as the Misuse Cases, Secure-UML, and UMLsec to elicit and quantify the security requirements in different phases of the SDLC.

2) Future studies could examine other issues concerning software security such as security design, security requirements specification, policy, in addition to the issues covered in the study.

3) The developed techniques in the study include the (SQUARE, CLASP and FUZZY THEORY) methods, so it is suggested that future studies should adopt other methods such as the MSRA.

4) This proposed technique (SAIFQT) tool is converted to be able to be used automatically to elicit and quantify the security requirements.