LEVERAGING THE POWER OF SQA TO ENHANCE SOFTWARE SECURITY

Kuruppu Arachchilage Hashantha Udara Jayasekara

(179114N)

Degree of Master of Business Administration in Information Technology

Department of Computer Science and Engineering

University of Moratuwa

Sri Lanka

May 2020

LEVERAGING THE POWER OF SQA TO ENHANCE SOFTWARE SECURITY

Kuruppu Arachchilage Hashantha Udara Jayasekara

(179114N)

The dissertation was submitted to the Department of Computer Science and Engineering of the University of Moratuwa in partial fulfillment of the requirement for the Degree of Master of Business Administration in Information Technology.

Department of Computer Science and Engineering

University of Moratuwa

Sri Lanka

May 2020

DECLARATION

I declare that this is my own work and this thesis does not incorporate without acknowledgment any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgment is made in the text.

Also, I hereby grant to the University of Moratuwa the non-exclusive right to reproduce and distribute my thesis/dissertation, in whole or in part in print, electronic, or other media. I retain the right to use this content in whole or part in future works (such as articles or books).

……. ……

K.A.H.U. Jayasekara Date:
Signature of the Candidate

The above candidate has carried out research for the Master's thesis under my supervision.

…….. ……

Dr. Shantha Fernando Date:
Signature of the Supervisor

COPYRIGHT STATEMENT

ABSTRACT

Software security is a growing concern for all ICT organizations since security breaches continue to make headline news. Since the Software Quality Assurance (SQA) professionals are responsible for validating the adherence to software product standards, processes, and procedures, getting them involved can help to solve most of the problem that harms most software development organizations today. Most of the experts involved in the software security industry spend much time discussing how to create secure software. Still, only a few explain how to achieve the goal of successful software security testing. As a result, SQA professionals face many problems in today's dynamic software environments. Organizations pressure them to certify software systems for security, but give little or no detailed advice on how to achieve that objective. It is essential to identify those problems and take the necessary actions to overcome those problems to thrive in the competitive business market so that this research intention is to find out a strategy that can use to develop the security testing mindset of SQA professionals by identifying the significant problems they are facing in software security testing and providing suitable suggestions/recommendations to overcome those problems.

For the research, we used qualitative content analysis research methodology. The survey questionnaires and interviews were conducted to collect data. The preliminary survey was conducted to determine the list of problems that SQA professionals face in software security testing. With the results of the initial study, an online survey was distributed to filter out significant problems. The online survey was shared among different leading IT companies. Lack of specialized SQA people in security testing, Budget, Lack of knowledge about security testing fundamentals, Lack of detailed information and advice, and No security testing training were some of the significant problems identified during the survey. With the results of the survey, a set of follow up interviews been carried with several senior SQA experts to sees their perspective on identified problems. Form a dedicated QA security taskforce to develop and retain the security testing mindset among SQA professionals, Maintain a security testing knowledge portal, Allocate sufficient funds in the budget to provide proper SQA resources and Familiarize and adapt security testing fundamentals, protocols, tools, and methods to fit within existing processes were some of the suggestions made by the domain experts, which they have successfully tried while addressing those problems.

This research delivers several valuable results that can be useful for SQA professionals to grow in software security testing gradually. By properly adopting the strategy, we expect to develop the security testing mindset of SQA professionals inside the organization as well as the industry as a whole. Improved SQA professionals will enhance software security.

ACKNOWLEDGMENT

I need to expose my true thankfulness to everybody who pushed for the successful completion of my research study on "Leveraging the power of SQA to enhance software security."

Primarily, I acknowledge my supervisor, Dr. Shantha Fernando, for the direction and support provided during the research study. Also, I would like to thank Dr. Shehan Perera, Head of the CSE department and Dr. Dilum Bandara, the former coordinator of the MBA in IT for assistance and teaching, and all other faculty associates in the CSE department at the University of Moratuwa.

Moreover, the tremendous support I got from the software professionals who supported me by filling out the surveys, participating interviews, and discussions are greatly admired. Without your help, this research study is not likely to finish successfully.

Exceptional acknowledgments must go to my mother 'Nilanka' father 'Dayananda' brother 'Dinith' and Sister 'Sasheeka' who continuously understanding of the sacrifices that I had to make this study success.

I'm deeply thankful to my friend, Kasuni Gunasekera, for the encouragement, and full support is given to me throughout this research.

The subsequent research dissertation accomplished with the support of people well-known and unknown to me. Finally, I appreciate those who encouraged me in various ways, even if they were not stated above but helpful for the success of this study.

TABLE OF CONTENTS

DECLARATION I
COPYRIGHT STATEMENT II
ABSTRACT III
ACKNOWLEDGMENT IV
TABLE OF CONTENTS V
LIST OF FIGURES VII
LIST OF TABLES X
LIST OF ABBREVIATIONS XII
1. INTRODUCTION 1
1.1. Background 1
1.2. Motivation 2
1.3. Problem Statement and Research Question 3
1.4. Research Objectives 3
1.5. Research Contribution 3
1.6. Organization of Thesis 5
2. LITERATURE REVIEW 6
2.1. Software Security Testing 6
2.2. Importance of Software Security Testing 7
2.3. Software Security Testing as SQA Profession 8
2.4. SQA Problems in Software Security Testing 10
2.5. Overcoming SQA Problems in Software Security Testing 12
2.6. Strategies Used to Develop the Security Testing Mindset of SQA 14
2.7. Summary 16
3. RESEARCH METHODOLOGY 17
3.1. Research Methodology 17
3.2. Measurement and Measures 19
3.3. Sampling Design 21
3.3.1. Population 21
3.3.2. Sampling Procedure 22
4. DATA ANALYSIS AND INTERPRETATION 23
4.1. Preliminary Survey Results 23

4.2. Online Survey Results 27
4.2.1. Analysis Based on Gender 30
4.2.2. Analysis Based on Organization Hierarchy 32
4.2.3. Analysis Based on Size of QA Department 37
4.2.4. Analysis Based on Company Type 40
4.2.5. Analysis Based on Target Market 43
4.2.6. Analysis of Identified Significant Problems 47
4.3. Interview Results 82
4.4. Summary of Results 100
5. CONCLUSION AND RECOMMENDATIONS 104
5.1. Evaluating the Objectives 104
5.2. Summary of Contributions 105
5.3. Limitations 106
5.4. Recommendations 106
5.5. Future Work 110
REFERENCES 111
APPENDIX A: PRELIMINARY SURVEY QUESTIONNAIRE 114
APPENDIX B: ONLINE SURVEY QUESTIONNAIRE 116
APPENDIX C: INTERVIEW QUESTIONNAIRE 126
APPENDIX D: INTERVIEW RESULTS 129

LIST OF FIGURES

Figure 3.1: The research methodology. 18
Figure 3.2: Total ICT workforce by work category (SLICTA, 2013). 22
Figure 4.1: Percentage-wise analysis of problems in the online survey. 29
Figure 4.2: Weighted score analysis of problems in the online survey..... 30
Figure 4.3: Problems distribution for the male category..... 30
Figure 4.4: Problems distribution for the female category. 31
Figure 4.5: Problems distribution for the engineering category. 33
Figure 4.6: Problems distribution for the tactical management category. 34
Figure 4.7: Problems distribution for the middle management category. 35
Figure 4.8: Problems distribution for the executive management category..... 36
Figure 4.9: Problems distribution for QA department size less than 10 categories. 37
Figure 4.10: Problems distribution for QA department size less than 50 categories. 38
Figure 4.11: Problems distribution for QA department size more than 50 categories. 39
Figure 4.12: Problems distribution for the product development category..... 40
Figure 4.13: Problems distribution for the IT service category. 41
Figure 4.14: Problems distribution for both product development and IT service category.. 42
Figure 4.15: Problems distribution for the overseas category..... 44
Figure 4.16: Problems distribution for the local market category..... 45
Figure 4.17: Problems distribution for the overseas and local market category. 46
Figure 4.18: The agreeable extent of the participants for the “Lack of specialized SQA people in security testing” as a problem. 48
Figure 4.19: Gender-wise analysis on the distribution of “Lack of specialized SQA people in security testing” in the online survey. 49
Figure 4.20: Organization level-wise analysis on the distribution of “Lack of specialized SQA people in security testing” in the online survey. 49
Figure 4.21: Size of the QA department wise analysis on the distribution of “Lack of specialized SQA people in security testing” in the online survey. 50
Figure 4.22: The agreeable extent of the participants for the ‘Budget’ as a problem. 51
Figure 4.23: Gender-wise analysis of the distribution of ‘Budget’ in the online survey. 52
Figure 4.24: Organization level-wise analysis of the distribution of ‘Budget’ in the online survey. 52
Figure 4.25: Size of the QA department wise analysis on the distribution of ‘Budget’ in the online survey. 53
Figure 4.26: The agreeable extent of the participants for the ‘Lack of knowledge about security testing fundamentals’ as a problem. 54
Figure 4.27: Gender-wise analysis on the ‘Lack of knowledge about security testing fundamentals’ in the online survey distribution. 54
Figure 4.28: Organization level-wise analysis on the ‘Lack of knowledge about security testing fundamentals’ in the online survey distribution. 55
Figure 4.29: Size of the QA department wise analysis on the ‘Lack of knowledge about security testing fundamentals’ in the online survey distribution. 55
Figure 4.30: The agreeable extent of the participants for the ‘Lack of detailed information and advice’ as a problem. 57
Figure 4.31: Gender-wise analysis on the ‘Lack of detailed information and advice’ in the online survey distribution. 57

Figure 4.32: Organization level-wise analysis of the ‘Lack of detailed information and advice’ in the online survey distribution. 58
Figure 4.33: Size of the QA department wise analysis on the ‘Lack of detailed information and advice’ in the online survey distribution. 58
Figure 4.34: The agreeable extent of the participants for the ‘No security testing training’ as a problem. 60
Figure 4.35: Gender-wise analysis of the ‘No security testing training’ in the online survey distribution. 60
Figure 4.36: Organization-level wise analysis of the ‘No security testing training’ in the online survey distribution. 61
Figure 4.37: QA department wise analysis on the ‘No security testing training’ in the online survey distribution. 61
Figure 4.38: The agreeable extent of the participants for the ‘Lack of time’ as a problem. .. 63
Figure 4.39: Gender-wise analysis of the ‘Lack of time’ in the online survey distribution... 63
Figure 4.40: Organization level-wise analysis of the distribution of ‘Lack of time’ in the online survey. 64
Figure 4.41: Size of the QA department wise analysis on the ‘Lack of time’ in the online survey distribution. 64
Figure 4.42: The agreeable extent of the participants for the ‘Complexity’ as a problem..... 66
Figure 4.43: Gender wise analysis of the ‘Complexity’ in the online survey distribution..... 66
Figure 4.44 Organization level-wise analysis of the ‘Complexity’ in the online survey distribution. 67
Figure 4.45 Size of the QA department wise analysis on the ‘Complexity’ in the online survey distribution. 67
Figure 4.46: The agreeable extent of the participants for the ‘Less SQA involvement in system design, requirement gathering, and code review phases’ as a problem. 68
Figure 4.47: Gender wise analysis of the ‘Less SQA involvement in system design, requirement gathering, and code review phases’ in the online survey distribution. 68
Figure 4.48: Organization level-wise analysis on the ‘Less SQA involvement in system design, requirement gathering, and code review phases’ in the online survey distribution... 69
Figure 4.49: Size of the QA department wise analysis on the ‘Less SQA involvement in system design, requirement gathering, and code review phases’ in the online survey distribution. 69
Figure 4.50: The agreeable extent of the participants for the ‘Lack of management support’ as a problem. 71
Figure 4.51: Gender-wise analysis of the ‘Lack of management support’ in the online survey distribution. 71
Figure 4.52: Organization level-wise analysis of the ‘Lack of management support’ in the online survey distribution. 72
Figure 4.53: Size of the QA department wise analysis on the ‘Lack of management support’ in the online survey distribution..... 72
Figure 4.54: The agreeable extent of the participants for the ‘No project requirements’ as a problem. 74
Figure 4.55: Gender-wise analysis of the ‘No project requirements’ in the online survey distribution. 74
Figure 4.56: Organization level-wise analysis of the ‘No project requirements’ in the online survey distribution. 75

Figure 4.57: Size of the QA department wise analysis on the ‘No project requirements’ in the online survey distribution. 75
Figure 4.58: The agreeable extent of the participants for the ‘Lack of motivation t’ as a problem. 76
Figure 4.59: Gender-wise analysis of the ‘Lack of motivation’ in the online survey distribution. 77
Figure 4.60: Organization level-wise analysis of the ‘Lack of motivation’ in the online survey distribution. 77
Figure 4.61: Size of the QA department wise analysis on the ‘Lack of motivation’ in the online survey distribution. 78
Figure 4.62: The agreeable extent of the ‘Lower salary scale compared to other IT professions’ as a problem. 79
Figure 4.63: Gender-wise analysis of the ‘Lower salary scale compared to other IT professions’ in the online survey distribution. 80
Figure 4.64: Organization level-wise analysis of the ‘Lower salary scale compared to other IT professions’ in the online survey distribution. 80
Figure 4.65: Size of the QA department wise analysis on the ‘Lower salary scale compared to other IT professions’ in the online survey distribution. 81

LIST OF TABLES

Table 3.1: Mapping problems to the questions. 19
Table 3.2: Mapping suggestions to the questions to overcome problems. 20
Table 4.1: Distribution of the organization levels in the preliminary survey. 24
Table 4.2: Distribution of the organization type in the preliminary survey. 24
Table 4.3: Distribution of the target market in the preliminary survey. 24
Table 4.4: Distribution of the department size in the preliminary survey. 24
Table 4.5: Problems encountered by SQA professionals in software security testing. 25
Table 4.6: Suggestions to overcome the security testing problems faced by SQA professionals. 26
Table 4.7: Distribution of the organization levels in the online survey. 27
Table 4.8: Distribution of the organization type in the online survey. 27
Table 4.9: Distribution of the target market in the online survey. 28
Table 4.10: Distribution of the department size in the online survey. 28
Table 4.11: Distribution of gender in the online survey. 28
Table 4.12: Problems ranking for the male category. 31
Table 4.13: Problems ranking for the female category. 32
Table 4.14: Problems ranking for the engineering category. 33
Table 4.15: Problems ranking for the tactical management category. 34
Table 4.16: Problems ranking for the middle management category. 35
Table 4.17: Problems ranking for the executive management category. 36
Table 4.18: Problems ranking for QA department size less than 10 categories. 37
Table 4.19: Problems ranking for QA department size less than 50 categories. 38
Table 4.20: Problems ranking for QA department size more than 50 categories. 39
Table 4.21: Problems ranking for the product development category. 41
Table 4.22: Problems ranking for the IT service category. 42
Table 4.23: Problems ranking for product development and IT service category. 43
Table 4.24: Problems ranking for the overseas category. 44
Table 4.25: Problems ranking for the local market category. 45
Table 4.26: Problems ranking for the overseas and local market category. 46
Table 4.27: Ranking summary of the identified problems based on demographic data. 47
Table 4.28: Suggestions distribution made to overcome ‘Lack of specialized SQA people in security testing’ problem in the online survey. 50
Table 4.29: Suggestion distribution made to overcome the ‘Budget’ problem in the online survey. 53
Table 4.30: Suggestions distribution made to overcome ‘Lack of knowledge about security testing fundamentals’ problem in the online survey. 56
Table 4.31: Suggestions distribution made to overcome ‘Lack of detailed information and advice’ problem in the online survey. 59
Table 4.32: Suggestion distribution made to overcome ‘No security testing training’ problem in the online survey. 62
Table 4.33: Suggestion distribution made to overcome ‘Lack of time’ problem in the online survey. 65
Table 4.34: Suggestion distribution made to overcome the ‘Complexity’ problem in the online survey. 67
Table 4.35: Suggestion distribution made to overcome ‘Less SQA involvement in system design, requirement gathering, and code review phases’ problem in the online survey. 70

Table 4.36: Suggestions distribution made to overcome ‘Lack of management support’ problem in the online survey. 73
Table 4.37: Suggestion distribution made to overcome ‘No project requirements’ problem in the online survey. 76
Table 4.38: Suggestion distribution made to overcome ‘Lack of motivation’ problem in the online survey. 78
Table 4.39: Suggestion distribution made to overcome ‘Lower salary scale compared to other IT professions’ problem in the online survey. 81
Table 4.40: Summary of online survey problems. 100
Table 4.41: Summary of online survey suggessions. 101
Table 4.42: Mapping of problems with suggestions as to the strategy. 102

LIST OF ABBREVIATIONS

Abbreviation	Description
BeEF	Browser Exploitation Framework
CR	Change Request
CWE	Common Weakness Enumeration
HTTP	Hyper Text Transfer Protocol
ICT	Information and Communications Technology
IT	Information Technology
MBA	Master of Business Administration
OWASP	Open Web Application Security Project
POC	Proof of Concept
QA	Quality Assurance
QC	Quality Control
QMS	Quality Management System
Q&A	Question and Answer
ROI	Return on Investment
SANS	SysAdmin, Audit, Network and Security
SDLC	Software Development Life Cycle
SQA	Software Quality Assurance
SQL	Structured Query Language
XSRF	Cross-Site Request Forgery
XSS	Cross Site Scripting
ZAP	Zed Attack Proxy

1. INTRODUCTION

1.1. Background

Software quality assurance is a key element in all ICT businesses (Frankk, 2014) since they are responsible for verifying the quality of a software product. A very competitive ICT business is there in the global market. Hence, companies should assure the product quality they present to the customers to survive in the business. A conventional quality assurance process may do for IT companies to improve their products while maintaining quality at various stages over the production process.

SQA aims to sustain the highest quality of a software product (AshfaqQazi et al., 2012). Software security is an increasing matter for most software organizations because security breaks continue to receive breaking news. Involving testers can help to resolve most of the issues that harm most software companies today. SQA professionals could play a significant part in software security testing, as they know wherewith an application matches all-together and where to find the linking points. This overall view is essential for the implementation of the security testing basics.

Although software development companies have separate QA departments to identify functional errors, the majority do not have proper security testing procedures (Abela, 2017). Most of the security experts spend many hours considering how to build secure software. Still, not many reveal about reaching the intention of wealthy software security testing (Whittaker, 2006). Therefore, SQA professionals encounter many problems in changing software environments. Companies force them to testify software applications for security but give limited or no comprehensive guidance on how to succeed that objective.

Thus, it can conclude that SQA professionals faced many problems in software security testing, and suggestions and recommendations can implement to overcome these problems.

1.2. Motivation

As a senior software automation engineer and worked as a security tester for many years, I saw that SQA people in IT enterprises encounter many problems when achieving the purpose of wealthy software security testing. There are several security-related works of literature in the software industry to support software vendors, for instance, books like Software Security-Building Security by Mcgraw (2006) and Craft of System Security by Smith et al. (2008), but a few guidance on reaching the intention of wealthy software security testing. Testers need to develop a primary security testing mindset, but they usually receive no training in software security testing (Lent, 2013).

Security and software quality are not isolated worlds, but two glances at the same coin (Woody et al., 2014). However, when it comes to software security testing, SQA professionals have an exciting idea saying that security testing is not QA's job. It is untrue because QA engineers are bound to the quality of the software in which security is a component. If they failed to ensure the secureness of a software/application, it is the same as they failed in their job (Los, 2011).

Therefore, security testing is an essential part of the software testing process, which implies the particular talents are becoming as required for quality assurance teams. But software security is a challenging task even for an experienced tester. With the above findings, it motivates us to identify the problems faced by SQA professionals in software security testing and a proper strategy that can use to overcome the identified problems.

1.3. Problem Statement and Research Question

As per Sections 1.1 and 1.2 findings, it is imperative to say that most of the SQA professionals do not know how to succeed in the aim of a wealthy software security testing. So, they face many problems in the current dynamic software environments. Organizations force them to testify software applications for security but give limited or no comprehensive guidance on how to succeed that objective.

It is necessary to identify a strategy to solve those problems so that it can use to develop the security testing mindset of SQA professionals. Therefore, we raised the research question as:

What strategy can be used to develop the security testing mindset of SQA professionals?

1.4. Research Objectives

It is needed to recognize the significant problems faced by SQA professionals in software security testing and suggestions to overcome those problems since the mapping of those problems to the suggestions can be used to find out the strategy. Hence, the research objectives are:

- To distinguish the significant problems encountered by SQA professionals in software security testing.
- To determine and present recommendations and suggestions to accomplish the identified problems.

1.5. Research Contribution

The research method of qualitative content analysis used to conduct the study by picking a sample of SQA professionals. Semi-structured questionnaires and interviews used to collect appropriate information. Preliminary and online surveys carried to identify the significant problems faced by SQA professionals in software security testing. Interviews among SQA higher management utilized to determine the recommendations and suggestions to conquer the identified problems at the last stage of this study. The research has distinguished significant problems encountered by SQA professionals in software security testing and suggestions and recommendations to

accomplish those problems from the management viewpoint. Two essential findings are:

- **Problem 01**: Lack of specialized SQA people in security testing

  Suggestion: Form a dedicated QA security taskforce to develop and confine the security testing mentality between SQA professionals.

  This task force will have one main goal: to develop and maintain a security testing mentality between SQA professionals. Members of this task force will be able to continuously learn new tools and methods, use their security testing capabilities, and distribute their experience among other group members.

- **Problem 02**: Budget

  Suggestion: Allocate sufficient funds in the budget to provide proper SQA resources. (e.g., people, tools, environments)

  When offering estimates, all security risk factors need to identified, highlighted, quantified, and communicated to the client. The QA manager needs to provide the pros and cons of the available possibilities to the top management and allow them to choose the greater worthwhile option.

Concerning the ICT workforce survey 2013, 8% is classified into the SQA job category. Therefore, we consider a strong implementation of the recommended suggestions will be useful for SQA professionals to grow in the industry by developing their security testing mindset.

1.6. Organization of Thesis

Chapter one provides the motivation, research background, research problem, and the research objectives. Chapter two will provide the related work associated with SQA Professionals and Security Testing.

Chapter three explains the adopted research methodology for the study including questionnaire development, survey approach, and interviews. The fourth chapter will examine a detailed analysis and discussions of the observations and results obtained.

Based on the data analysis and interpretation, Chapter five concludes the total research outcome, including the limitations and recommendations along with the directions for future work.

2. LITERATURE REVIEW

2.1. Software Security Testing

Software security testing is a software test type that determines potential security risks in a software project. It can divide into functional and vulnerability security testing. Functional security testing assures that security functions are correctly implemented correctly and meet security requirements based on the security requirements spec. Security requirements include data availability, authorization, control, confidentiality, privacy protection access, integrity, authentication, auditing, and security management. Security vulnerability tests used to detect losses assigns to the design, implementation, operation, and management of the system as an attacker (Tian-yang et al., 2016).

Software malfunctions happen directly in the world, without intentional damages. Conventional software testing discussions refer to what seek when software crashes, despite its purpose. Thus the distinction within software security and safety is the appearance of a clever attacker trying to hack. So that security testing involves two ways. Standard organizations practicing the usual way can achieve functional security testing. For instance, assuring that the access control devices work as told is a standard test.

Differently, the usual SQA employees will find it harder to conduct risk-based security testing. First, security tests (especially those that lead to full use) are challenging to create, as the designer needs to think like an attacker. Second, security tests not usually cause direct security accomplishment and, thus, generate a detectable problem. It can lead to unexpected results that require further complex analysis from the tester. Bottom line: risk-based security testing count further on knowledge and the experience (Potter et al., 2004).

2.2. Importance of Software Security Testing

Initially, the internet considered a brave new world, and now that the internet is everywhere. As a result, cybersecurity is a serious problem not only for individuals but also for enterprises that are trusted to securely store data, ranging from customer names and email addresses to even more classified information such as credit card numbers and trade secrets. Nowadays, data is the currency, and many vile people are willing to spend and risk almost anything to get them. Considering all this, now more than ever, companies must implement a healthy approach to secure their applications, websites, and any other digital products that can receive or store essential data from customers, clients, and partners (Wysopal et al., 2006).

Software quality and security testing have a clear connection. The fact that software meets functionality and performance-related quality requirements do not necessarily indicate that it is secure. Security testing measure that protects disclosure of data to parties, not to the expected recipient, which is not the only way to ensure security. The efficient method to produce secure software is to strictly align the development of life cycle processes with the principles and practices of safe development, deployment, and support. Ensure software security is the process of determining that a data system shields data and supports functionality as expected. Software testing is to ensure that the systems meet the requirements. It expected those functional specifications and would, but not significantly, reflect the functionality that potential users need, especially aspects that users may not know or have not asked to consider (Sanksoft, 2017).

2.3. Software Security Testing as SQA Profession

Software quality assurance professionals will carefully examine the software product with project managers and developers to realize what each module should meet, what are the main functions, and who will be the end-users.To accomplish this, QA's will go after multiple test strategies and processes. The aim intends to reduce errors as much as possible and enhance the quality of the final product.

Although, software security ensures that security is taken into account at every stage of software development to protect an application. Security defects in any form should also be considered a quality assurance problem. It can argue that there can be security vulnerabilities in software that already consist of quality issues. Bad software code quality can lead to unexpected behavior. For users, this often results in poor usage. For a hacker, this makes possible to subject the system to stress unexpectedly. Development teams that pay great attention to quality, as a rule, had less vulnerability in their code. Security and software quality are not isolated worlds, but two glances at the same coin, an error that shows itself today as a failure of the system, could become a vulnerability that can exploit tomorrow (Rosenberg, 2002).

Software security and quality assurance are associated with risk elimination. Software security groups are working to eliminate security risks, and quality assurance groups are working to reduce quality risks (Wisseman, 2018). The combination of software quality assurance (QA) and IT security leads to a symbiotic relationship. Still, few organizations have begun to recognize the benefits of working together between these two separate groups. The IT Quality Assurance and Security Alliance are natural because IT security is a form of quality assurance at a basic level. Exposure to safety in any kind is a quality assurance issue. QA teams have long figured out that quality assurance professionals required in the early stages of the software/application development life cycle for several reasons. The first reason is that the detection of quality problems immediately before deployment causes big headaches. However, QA also found that finding issues at the requirements gathering stage is much easier and cheaper to fix at the requirements stage than after the developers wrote something. Early participation in the SDLC is also useful, as quality assurance teams can begin to create test scenarios and test scenarios before building software. Security should do

the same, working hand in hand with the QA team, system analysts, and developers to help system analysts gather requirements and build security design in software development. IT security analysts should also develop the types of tests that should perform at each stage. IT security specialists, together with the quality assurance team, must write test suites and plans for each test that the IT security team wants (Laskowski, 2011).

SQA professionals could play a significant part in software security testing, as they know wherewith an application matches all-together and where to find the linking points. This overall view is essential for the implementation of the security testing basics. So the involvement of testers helps to solve most of the problems that harm most ICT organizations (Lent, 2013). Although development companies have departments designed to identify functional errors, most of them do not have any security testing procedures. When a software developer implements a new button to a web application, there are lots of test procedures to check its functionality, but very limited or no procedures are to test the feature behind that button, and it perhaps forged. It is mainly since lots of companies still fail to distinguish between security testing and functional testing, or management is not aware of the indications that a security problem that exploited may have for customers' businesses. So, the security test for any software product should be included in the SDLC with routine quality assurance testing (Abela, 2017).

SQA professionals usually focused on functional testing to verify the correct behavior of a software product. But, as software security becomes more serious, the company quality control unit is more feasible to become an essential player in software security testing (English, 2014).

2.4. SQA Problems in Software Security Testing

Bonver et al. (2012) analyzed and recognized the resulting problems faced by SQA professionals in security testing:

- Lack of knowledge about security testing fundamentals
- Lack of detailed information and advice
- Do not accurately gather security requirements or do not gather at all
- Lack of management support

There are many security literatures in the software industry that helps software vendors; for instance, books like Software Security-Building Security by Mcgraw (2006) and Craft of System Security by Smith et al. (2008) are trying to solve these problems. Besides, many providers nowadays offer on-the-spot security testing training. Most experts who engaged in similar works contribute many hours considering how to build secure software. Still, not many reveal about reaching the intention of wealthy software security testing. Therefore, SQA professionals encounter many problems in changing software environments. Organizations force them to testify software applications for security but give limited or no comprehensive guidance on how to succeed that objective. Furthermore, since security testing is more of an art than a science, conducting one-time training on software security testing will not surely afford a durable return on investment for a software vendor (Wysopal et al., 2006; Dowd et al., 2006).

As mentioned by Rosenberg (2002), SQA faces a higher amount of problems when determining the quality of a software product. It is necessary to have a thorough understanding of a quality software definition, but the software usage environment usually influences the final description. Moreover, this stage is a critical and challenging area that seriously affects the final project outcome. SQA's face difficulties in costs and time when they are delivering software products for customer usage. Most of the time organizations should deliver the highest quality with a lower budget. Since they fail to achieve it, numerous software corporations encounter legal issues and lose

valuable clients. Not only that but also the business is going down from the market (Sigrid, 2006).

As mentioned by Los (2011), when it comes to software security testing, SQA professionals have few false opinions. Rely on whom you questioned, you might get many different answers. He identified the following problems:

- Security testing is very difficult [complicated]
- Testing software for security is not a QA's work
- SQA professionals cannot be useful at security testing
- QA analysts do not understand security testing

Takenen (2008) analyzed how quality assurance is relevant to the topic of security testing? The author says that the traditional security testing process took place late in the software cycle. They designed to protect software from obvious attacks and to identify obvious vulnerabilities in already launched systems. Even if the usual security evaluation be found in launching vulnerability detection tools are not trying to discover something fresh and unusual. It is still well suited for post-deployment processes, but for beneficial quality goals, we need something else.

AshfaqQazi et al. (2012) have driven research into the Pakistan software industry to analyze the possibilities to improve SQA in developing countries. They addressed consecutive general SQA problems which also credible for SQA's when involving in security testing:

- Deficiency of experts (not having decent testers/test teams)
- Team forming for requirement collection (no QA participation for requirement gathering)
- Budget (unrealistic project budget)

Furthermore, Iqbal and Qureshi (2012) identified the subsequent fundamental difficulties:

- Test time reduction
- Inadequate management support
- Inadequate domain knowledge

2.5. Overcoming SQA Problems in Software Security Testing

Javed et al. (2012) presented a few results for the established issue areas to the Pakistan software industry. They have address following general SQA solutions which also can be relevant when SQA involving in security testing:

- Certified and specialized SQA team
- Specialized domain knowledge

Mostly, the intention of security testing poorly relates only to the field of functional security testing, as portrayed in the declared functional requirements. Yet, a simple examination of security features does not solve the vulnerability problem. Thus, the natural starting point for a quality assurance specialist is to think like an attacker whose task is to determine how the systems' functions, technologies used, the logic of the business, the implementation, as well as in the configurations. SQA personal should be included in the software design stage, implementation, and customization of the system, working together with architects and developers to identify potential vulnerabilities (Basu, 2013).

Another aspect of security testing is knowledge of security fundamentals. Like, QA professional who tests web apps, will begin with standard attacks, like cross-site scripting and SQL injections, which are well-grounded and have multiple illustrations to pursue (Abela, 2017).

Knowledge of security fundamentals is insufficient to do a successful security test. Quality assurance specialists should have an in-depth understanding of the design and implementation of the system, as well as an absolute knowledge of the primary environment. The idea is that the SQA team should participate in non-QA phases of

the development life cycle, such as requirements, design, and code checks. An overview of general requirements are the best way to understand how the system should have worked and to ensure that security requirements are clearly and correctly defined. Performing code verification from a security point of view is the best way to understand how the code works (Takenen, 2008). Security tools are designed to improve performance, either by facilitating testing or by detecting vulnerabilities in fully automatic mode. SQA specialists should precisely inform with such tools. They should know when and how to use each tool (English, 2014).

QA personnel training in risk management improves their analysis of where to look for vulnerabilities, based on business and technical aims of the system, and the risks that may violate these goals as a result of a malicious attack. Developing a security testing mindset requires more than one or more security training sessions; it is a continuous, iterative process. It is unfair to expect from QA for continuously reading related literature, monitoring relevant websites, and correctly applying new knowledge gained during the normal project cycle. Therefore Bonver et al. (2012) suggest a solution to form a security testing taskforce. This particular group will have one main goal in mind: to develop and retain the security testing mindset among SQA professionals. Target group members will have the opportunity to continually learn new tools and methods of security testing, use their security testing capabilities, and share their experience with the rest of the target group. To determine the basics of security testing, SQA professionals should be motivated to participate. Management can cope with motivation on several fronts, primarily by making participation in the working group exciting and making it worthy of the attention of an employee of the quality assurance department (Lent, 2013).

Adopting a security testing task force is an essential step in helping the QA teams to develop their thinking about security testing. An essential element of enhancing knowledge about security testing is the use of the knowledge portal. The portal primarily helps QA staff to keep up with the latest training given by the security testing taskforce. Still, it is also an excellent place to advertise relevant resources for security testing on the Internet, including information about security tools, attacks, vulnerabilities, and training materials (Bonver et al., 2012).

2.6. Strategies Used to Develop the Security Testing Mindset of SQA

According to Hrynczak (2016), security testing is a very vital part when testing software applications, which means that these skills are becoming in demand for QA teams. So, he outlines the following steps to follow when starting building up security testing skills even for an experienced tester:

- Understand the application
- Understand security terms and definitions
- Use online training tools
- Learn from others
- Learn to use an automated vulnerability scanner
- Share what you are learning
- Convince people that security is important
- Communicate security issues in the context
- Use useful default test data
- Practice
- Use automation when appropriate
- Read books
- External training

Bonver et al. (2012), declaring that adopting a security taskforce is a vital step in helping the QA teams to develop their thinking about security testing. An essential element of enhancing knowledge about security testing is the use of the knowledge portal. The portaliprimarily helps QA staff to keep up with the latest training. Still, it is also an excellent place to advertise relevant resources for security testing on the Internet, including information about security tools, attacks, vulnerabilities, and training materials.

According to Borodina (2019), security testing is a difficult skill to learn. A professional will not become a good security tester by just doing a few online courses. Part of becoming a confident security tester is building their library of tools. QA must have a comfortable setup. Once they have learned and practiced the fundamental principles, they can move on to leading some nifty tricks. Once they have some theory down, they can start practicing by doing hacking challenges. Tools do not make a good security tester. However, they are not going to get too far without them. So, it recommended starting with a leading couple of necessary tools. Security testing is confusing and overall just frustrating, so try to join communities to learn from others.

Erikson (2018) stating that reading a lot is a necessity, simply because it is too much, QA needs to know in software security testing. Signup and complete legal hacking trials so that they will learn to be innovative in finding ways to break security. Going further understand networking, getting a QA position at a company that has a penetration testing opportunity is a solid choice.

According to ThinkSys (2017), to make an application vulnerability-resistant, it is vital to have a strong strategy for security testing. Learning tools that help to do vulnerability scanning tests and the testing includes targeted testing where the QA team and the security testers work together, help to gather security testing fundamentals. Testers should handle security scans to evaluate network weakness to improve the scope of security testing. Also, having a security testing plan that works in order with the speed of software development becomes necessary.

2.7. Summary

There are various security-related works of literature to assist software vendors, but only a few describe how to achieve the goal of successful software security testing. In the literature, we found significant SQA problems in software security testing, such as inadequate knowledge about security testing, lack of advice, lack of management support, lack of time, unrealistic project budget, and not having proper testers. Furthermore, introducing more security testing training, recruit certified, and specialized SQA professionals, participation in traditionally non-QA phases of the project life cycle, adapting security testing task force, and use of a knowledge portal are some notable suggestions we found in the literature that can use to overcome SQA problems in security testing. Identified security testing strategies focused on developing technical skills (see Section 2.6). In this study, we address what it takes to conduct successful software security testing preliminary by identifying a strategy that can use to develop the security testing mindset of SQA professionals by exploring different problems faced by them in software security testing and providing suitable suggestions to overcome those problems.

3. RESEARCH METHODOLOGY

This chapter explains the procedure used to conduct the surveys and interviews, as well as analyzing the collected data during the surveys and interviews. Section 3.1 outlined the research methodology. Section 3.2 represents measurements and measures, while Section 3.3 illustrates the sample design.

3.1. Research Methodology

Figure 3.1 shows the research methodology used for the study. The research problem was determined based on a literature review. Based on a literature review and preliminary analysis, a list of problems identified which are faced by SQA professionals in security testing. 15 SQA professionals were used to verifying the clarity and validity of the preliminary questionnaire.

A preliminary survey conducted as the next step to find new problems and to verify the identified problems as a result of the literature review. The results of the preliminary study used to determine the list of problems and the most frequently used methods for overcoming them. Based on these results, an online survey was created and launched in 2019. The results of the online survey were analyzed and taken as input data for a semi-structured interview questionnaire for senior managers and industry experts of SQA.

Interviews conducted to find suggestions to overcome the identified problems based on the experience and expertise of the interviewed professionals. Interview results and the online survey results analyzed and used to build the strategy as the final result of this study.

Results of the preliminary survey, online survey, and interviews were analyzed using the qualitative content analysis research methodology. The selected methodology helped to identify essential aspects of the content and supporting arguments, as well as a clear and valid presentation of results.

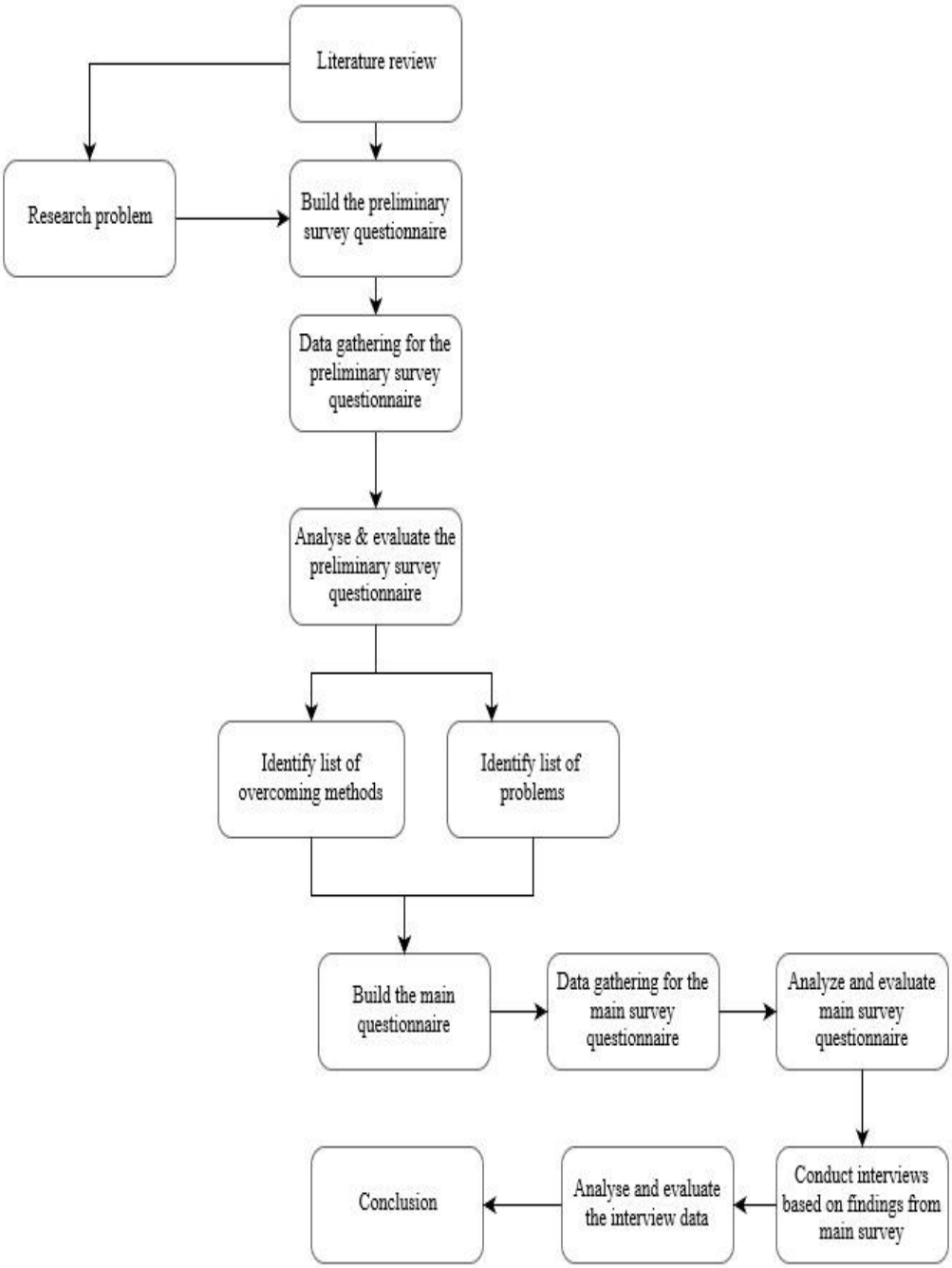

Figure 3.1: The research methodology.

3.2. Measurement and Measures

A preliminary survey conducted to a limited audience to ensure the clarity and validity of the questions. After that, it used to identify new problems and new suggestions to overcome those problems.

The questions used in the survey are list in Appendix A. Twelve problems, and fifteen suggestions to overcome those problems recognized and verified as the result of the preliminary survey.

The identified problems and suggestions then included in the online survey questionnaire (see Appendix B). The semi-structured questionnaires used for both preliminary and online surveys. Table 3.1 lists the mapping of questions to significant problems, and Table 3.2 contains the suggestions to overcome those problems. Each item in Tables 3.1 and 3.2 designed to get a 5-point Likert scale value from " strongly agree" to "strongly disagree."

Table 3.1: Mapping problems to the questions.

(This table continues to the next page.)

Problems	Scale	Measure	Question #
Complexity (e.g., hard to understand)	Likert	5-pointiscale	1.1
Lack of motivation	Likert	5-point scale	1.2
Lack of knowledge about security testing fundamentals (e.g., testing tools, frequent attacks like SQL injection)	Likert	5-point scale	1.3
Lack of detailed information's and advice	Likert	5-point scale	1.4
No security testing training	Likert	5-point scale	1.5
Lack of specialized SQAipeople in security testing	Likert	5-point scale	1.6
Lack of management support	Likert	5-point scale	1.7
Less SQA involvement in system design, requirement gathering and code review phases	Likert	5-point scale	1.8

Lack of time (e.g., due to regular project cycle)	Likert	5-point scale	1.9
No project requirements	Likert	5-point scale	1.10
Lower salary scale compared to other IT professions	Likert	5-point scale	1.11
Budget (e.g., less allocation of SQA people, tools,ienvironments)	Likert	5-point scale	1.12

Table 3.2: Mapping suggestions to the questions to overcome problems.

(This table continues to the next page.)

Suggestions	**Scale**	**Measure**	**Question #**
Form a dedicated QA security taskforce to develop and retain the security testing mindset among SQA professionals	Likert	5-point scale	3.1
Maintain a security testing knowledge portal	Likert	5-point scale	3.2
Motivate SQA people to do security testing sessions during project idle times	Likert	5-point scale	3.3
Familiarize and adapt security testing fundamentals, protocols, tools and methods to fit within existing processes	Likert	5-point scale	3.4
Introduced more security testing meet-ups and training for SQA people	Likert	5-point scale	3.5
Reduce SQA individual's lack of exposure to security testing by providing awareness	Likert	5-point scale	3.6
Advice SQA professionals to approach security testing with a risk management mindset	Likert	5-point scale	3.7
Recruit detail-oriented and experienced SQA professionals	Likert	5-point scale	3.8
Provide strong management support	Likert	5-point scale	3.9

Keep the higher management informed by having weekly, monthly progress review or awareness meetings	Likert	5-point scale	3.10
Working in tandem with architects and IT security teams to map out security vulnerabilities	Likert	5-point scale	3.12
Facilitate SQA participation in non-QAirelated phases of the development life cycle (e.g., System design, Requirement gathering, Code review)	Likert	5-point scale	3.11
Have SQA pool of people to service projects which are having security testing requirements	Likert	5-point scale	3.13
Increased standard of living for the skilled SQA resources	Likert	5-point scale	3.14
Provide proper SQA resources regardless of profit margin (e.g., people, tools, environments)	Likert	5-point scale	3.15

3.3. Sampling Design

3.3.1. Population

Research conducted for the SQA professionals in the Sri Lankan IT industry. According to the IT workforce survey conducted by the Sri Lanka Information and Communication Technology Agency (SLICTA) in November 2013, the total number of IT workforce projected for 2014 was 82,854 (SLICTA 2013). Out of this, 40.8% of the professionals were in ICT companies, 47.1% were in the non-ICT private sector, 7.8% were in government organizations, and 4.2% were in BPO companies. Since we are interested in SQA professionals related to software development, only ICT companies included in the study. Hence, the selected population was approximately 33,918. Out of this population, 8% (see Figure 3.2) of the professionals are estimated to be in theSQA profession (SLICTA 2013). Hence, the specific population under the study is 2,714.

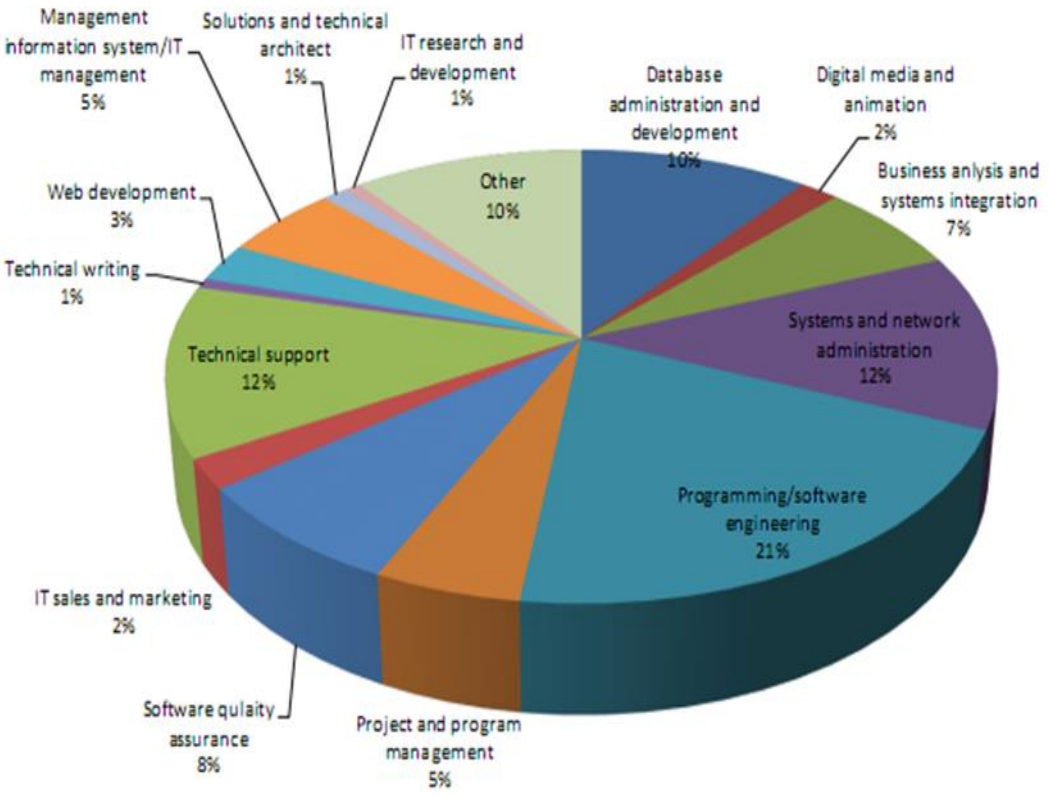

Figure 3.2: Total ICT workforce by work category (SLICTA, 2013).

3.3.2. Sampling Procedure

Initially, 15 SQA professionals (who worked in five different software companies) were used to test the clarity and validity of the preliminary questionnaire associated with the online survey. After that, an online survey provided to a random sample of SQA professionals selected from different software companies. While invitations to participate in the study were sent by e-mail to known SQA professionals, these professionals also invited to share the survey with their colleagues.

The sample size chose to achieve a confidence level of 95% and a confidence interval of 5, based on this value, 337 samples required for a population of 2714. Based on this, an online survey submitted to 337 SQA professionals. However, 337 replies were received; therefore, the response rate is 100%. A semi-structured interview was conducted with five SQA experts to find methods/solutions to overcome the problems identified in the questionnaire. These industry experts were reached either through a known party or through a self-introduction.

4. DATA ANALYSIS AND INTERPRETATION

This chapter analyzes the data collected through online survey questionnaires and interviews. The results of the preliminary study used to determine the list of problems that SQA professionals face in software security testing. The results of the online survey used to filter the significant problems identified during the preliminary study. Nevertheless, the results of the online survey used to verify the suggestions provided to overcome those identified problems according to the views of the survey participants. Semi-structured interviews were analyzed separately to determine the leader's opinions and recommendations related to the problems identified. Section 4.1 describes the distribution and the results of data for the preliminary survey. Section 4.2 provides data distribution and the analysis of the online survey. Section 4.3 provides interview results according to the views of the industry experts. Section 4.4 provides a summary of the online survey and interview results on significant problems and suggestions to overcome those problems.

4.1. Preliminary Survey Results

As mentioned above, the preliminary survey conducted to identify the problems faced by the SQA professionals in software security testing. Fifteen participants from five different software companies selected for this study. The participants of this survey categorized into four different levels in a typical organization. Table 4.1 depicts the distribution of the survey participants among the selected levels. Most of the survey participants belong to the engineering level. The participants in this survey divided into four types of organizations. Most of the survey participants were from organizations involved in both product development and IT services (see Table 4.2). As seen in Table 4.3, the participants divided into three different types depending on the target markets. Most of the participants were from the companies that focused on both local and foreign markets. Besides, participants divided into three groups depending on the size of their SQA department. As can be seen from Table 4.4, most of the participants worked in the SQA department, which employed more than 50 people.

Table 4.1: Distribution of the organization levels in the preliminary survey.

Level	Responses	Percentage
Executive Management	3	20%
Middle Management	3	20%
Tactical Management	4	27%
Engineer / Executive	5	33%
Total	**15**	**100%**

Table 4.2: Distribution of the organization type in the preliminary survey.

Organization Type	Responses	Percentage
Product Development	3	20%
IT Services	4	27%
Both	8	53%
Total	**15**	**100%**

Table 4.3: Distribution of the target market in the preliminary survey.

Target Market	Responses	Percentage
Local Market	3	20%
Overseas Market	4	27%
Both	8	53%
Total	**15**	**100%**

Table 4.4: Distribution of the department size in the preliminary survey.

Size of the QA department	Responses	Percentage
Less than 10	2	13%
10-50	6	40%
More than 50	7	47%
Total	**15**	**100%**

The preliminary questionnaire included open-ended questions to gather details related to security testing problems faced by SQA professionals. Significant problems identified based on the participants' comments. Table 4.5 summarizes the problems which are faced by the SQA professionals in software security testing. Further, the questionnaire included open-ended questions to gather details on suggestions to overcome the problems mentioned above. Suggestions by participants are in Table 4.6. These identified problems and suggestions used as input to the online survey questionnaire.

Table 4.5: Problems encountered by SQA professionals in software security testing.

(This table continues to the next page.)

Problems	Participant's Responses
Complexity	"Too hard to understand." "Quite difficult since there is no expressive and well-established framework or process is utilizing tools and documents." "Some security teams trying to impose a whole new suite of testing tools and methodologies on the QA analysts."
Lack of motivation	"Lack of motivation due to the complexity of the work." "Lack of motivation due to lack of information and support." "No motivation to participate for training."
Lack of knowledge about security testing fundamentals	"Do not have the basic knowledge to start the work."
Lack of detailed information's and advice	"Give little or no detailed advice on how to achieve the objective of successful software security testing."
No security testing training	"No proper training sessions and meet-ups."
Lack of specialized SQA people in security testing	"Do not Keep up to date knowledge on new technologies and how to provide SQA solutions to new trends." "The main challenge is to find QA Engineers who are highly technical and capable of working as team members and individual members." "Lack of security testing experts."
Less SQA involvement in system design, requirement gathering and code review phases	"To successfully test software for security, QA personnel do not have a deep knowledge of the system'sides gn and implementation, as well as a solid understanding of the underlying environment."
Lack of management support	"Neglecting the SQA role by the higher management. They do not pay for QA's to participate in a valuable meet up sessions." "Trying to rely on development team suggestions and approaches most of the time, which will lead to delivering incomplete successful QA tasks." "Management thinks QA cannot be effective at security testing since they are not experts."
Lack of time	"Limited Time duration to complete regular tasks." "Not enough time to focus on security testing due to regular project cycle." "It is unfair to ask QA testers to deal with continuously reading related literature, monitoring

	related web sites, and correctly applying the new knowledge learned during the regular project cycle."
No project requirements	"There are no project requirements to do security testing."
Lower salary scale compared to other IT professions	"QA professionals are paid less, even if they provide a valuable contribution to the project. Because of that, they are paying a lack of interest in doing tasks like security testing."
Budget	"Limited budget for QA related activities."

Table 4.6: Suggestions to overcome the security testing problems faced by SQA professionals.

#	**Suggestions**
1	Form a dedicated QA security taskforce to develop and retain the security testing mindset among SQA people.
2	Maintain a security testing knowledge portal.
3	Motivate SQA people to do security testing sessions during project idle times.
4	Familiarize and adapt security testing fundamentals, protocols, tools, and methods to fit within existing processes.
5	Introduced more security testing meet-ups and training for SQA people.
6	Reduce SQA individual's lack of exposure to security testing by providing awareness.
7	Advice SQA professionals to approach security testing with a risk management mindset.
8	Recruit detail-oriented and experienced SQA professionals.
9	Facilitate SQA participation in non-QA related phases of the development life cycle (e.g., System design, Requirement gathering, Code review).
10	Working in tandem with architects and developers to map out security vulnerabilities.
11	Provide strong management support.
12	Keep the higher management informed by having weekly, monthly progress reviews or awareness meetings.
13	Have an SQA pool of people to service projects which are having security testing requirements.
14	Increased standard of living for the skilled SQA resources.
15	Allocate sufficient funds in the budget to provide proper SQA resources. (e.g., people, tools, environments)

4.2. Online Survey Results

The online survey conducted to confirm the two main objectives of this research study, mentioned in section 1.4. We limit the analysis to determine the following:

- To identify the significant problems faced by SQA professionals in software security testing.
- To identify and present recommendations and suggestions to overcome the identified problems.

Participants categorized into four different levels in the organization. The majority of the participants were in the Engineer/Executive level. It is an advantage for the researcher, since most of the problems faced during this age of the profession. More than 50% of the participants were working as an engineer. Table 4.7 portrays the distribution of the survey participants among the selected levels. Participants further categorized into three different types of organizations. Their distribution represented in Table 4.8. The majority of the participants were working for an organization where their types are both IT services and product development. Participants also categorized into three different types of target markets. The majority of the participants were involved in Overseas market companies. Table 4.9 portrays the distribution of the survey participants among the selected target market.

Table 4.7: Distribution of the organization levels in the online survey.

Level	Responses	Percentage
Executive Management	31	9%
Middle Management	69	20%
Tactical Management	29	9%
Engineer / Executive	208	62%
Total	**337**	**100%**

Table 4.8: Distribution of the organization type in the online survey.

Organization Type	Responses	Percentage
Product Development	107	32%
IT Services	95	28%
Both	135	40%
Total	**337**	**100%**

Table 4.9: Distribution of the target market in the online survey.

Target Market	Responses	Percentage
Local Market	18	5%
Overseas Market	180	53%
Both	139	41%
Total	**337**	**100%**

Also, as can be seen from Table 4.10, the participants were divided into three groups depending on the size of their QA department. The majority of the participants were working in departments has more than 50 team members. Finally, the survey participants were identified by gender to ensure that both categories received equal opportunities to participate in the questionnaire. Table 4.11 portrays the distribution of survey participants.

Table 4.10: Distribution of the department size in the online survey.

Size of the QA department	Responses	Percentage
Less than 10	80	24%
10-50	95	28%
More than 50	162	48%
Total	**337**	**100%**

Table 4.11: Distribution of gender in the online survey.

Gender	Responses	Percentage
Male	179	53%
Female	158	47%
Total	**337**	**100%**

Next, we analyze the identified problems and viable solutions. Below two approaches have been used to identify the significant problems:

- Percentage scale
- Weighted scale

In the percentage-based approach, both "Strongly Agreed" and "Agreed" were considered as "Agreed." Similarly, "Disagree" and "Strongly Disagree" were

considered as “Disagree.” “Neither Agree nor Disagree” responses considered as “Neutral.” Figure 4.1 shows the results of the percentage-wise analysis of problems. The following figure used to identify the significant problems in software security testing.

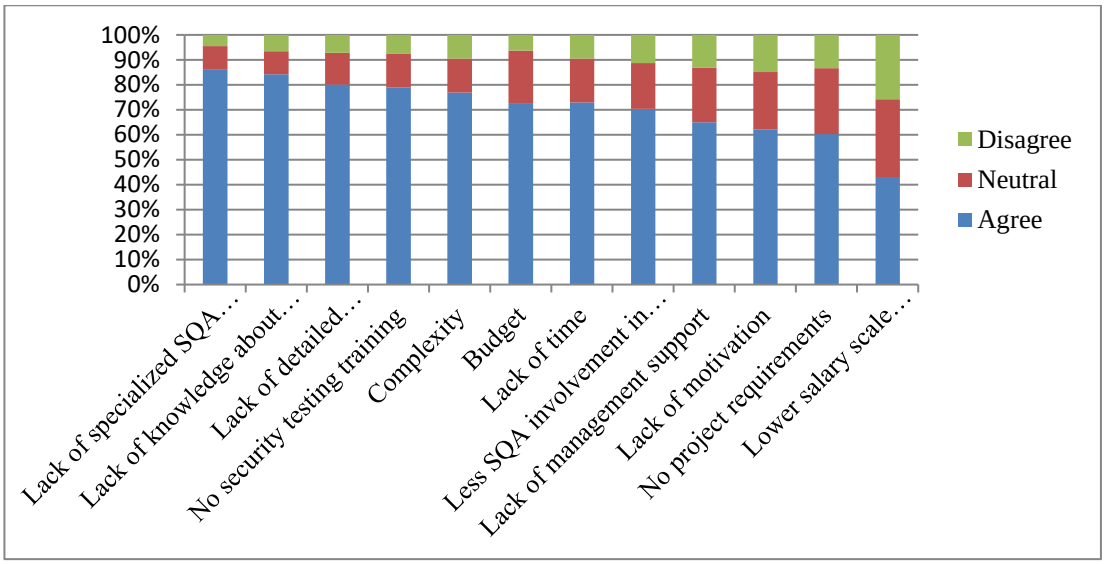

Figure 4.1: Percentage-wise analysis of problems in the online survey.

In the weighted scoring approach, weights for each Likert’s scale value assigned as follows:

- Strongly Agreed - 5
- Agreed - 4
- Neither Agree nor Disagree - 3
- Disagree - 2
- Strongly Disagree - 1

Figure 4.2 shows the weighted results of the problems in the online survey. This approach also shows results similar to the significant problems identified, using the percentage scale approach. The latter part of the study, ‘Strongly Agreed,’ ‘Agree’ and ‘Neither Agree nor Disagree’ was considered as ‘Agree.’ Also, ‘Disagree’ and ‘Strongly Disagree’ were considered ‘Disagree’ for the ease of analysis. It applies from Figure 4.3 onwards.

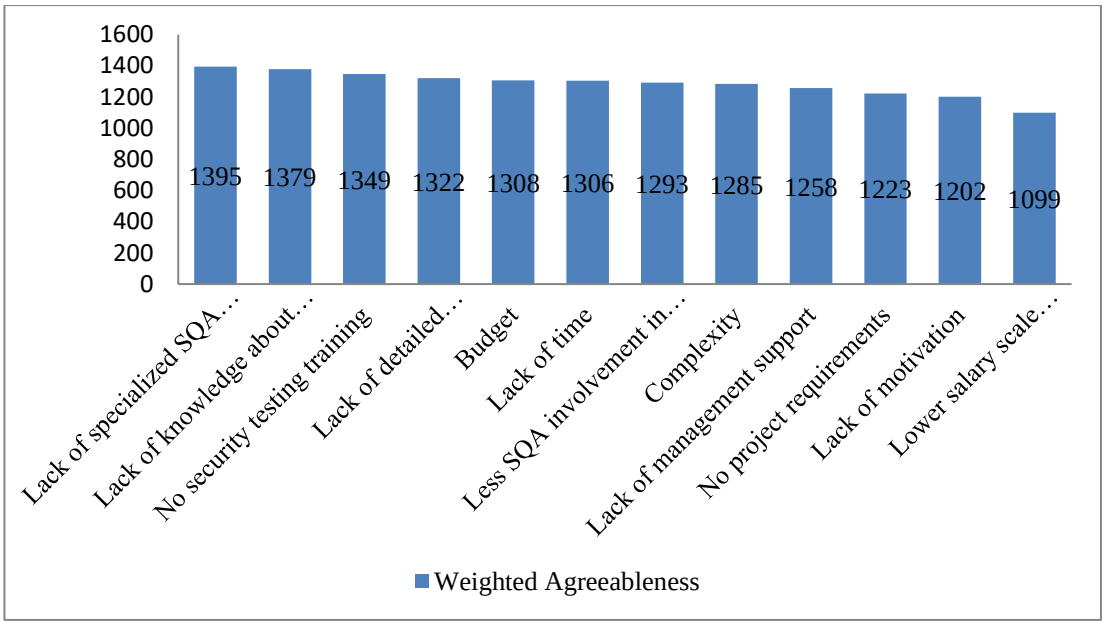

Figure 4.2: Weighted score analysis of problems in the online survey.

4.2.1. Analysis Based on Gender

Figure 4.3 depicts the agreeableness percentage towards the security testing problems faced by SQA males. Table 4.12 showing the stack ranking of the identified problems for the male category.

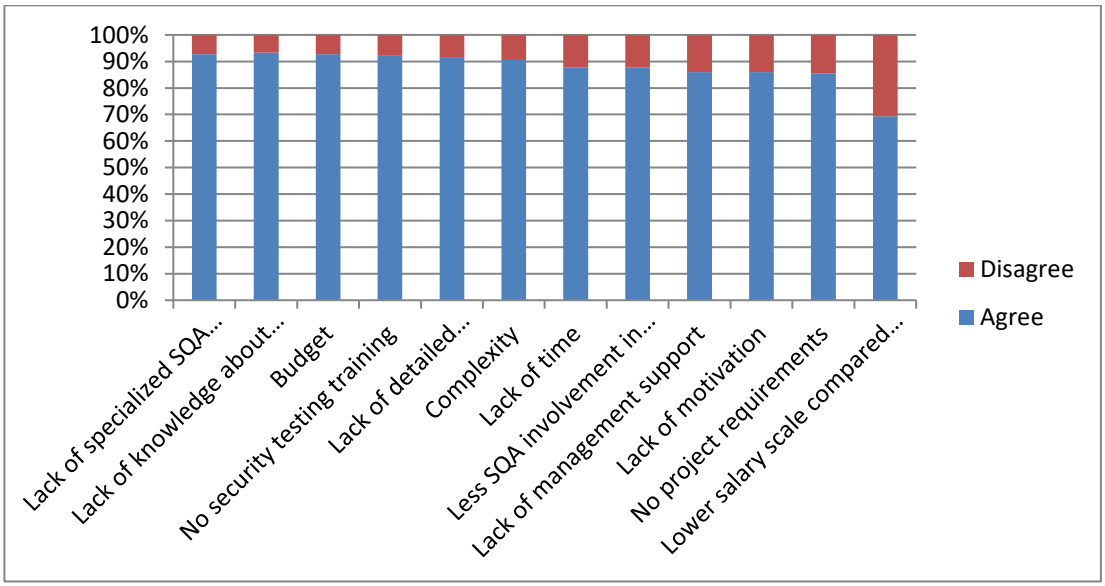

Figure 4.3: Problems distribution for the male category.

Table 4.12: Problems ranking for the male category.

Problem Description	Rank
Complexity	6
Lack of motivation	10
Lack of knowledge about security testing fundamentals	2
Lack of detailed information's and advice	5
No security testing training	4
Lack of specialized SQA people in security testing	1
Less SQA involvement in system design, requirement gathering and code review phases	8
Lack of management support	9
Lack of time	7
No project requirements	11
Lower salary scale compared to other IT professions	12
Budget	3

In the female-only distribution, the ranking positions and the agreeableness percentage towards the problems are different from the male distribution. However, the identified significant problems remained the same for both males and females. Figure 4.4 depicts the distribution of problems for the female category. Table 4.13 shows the stack ranking for females.

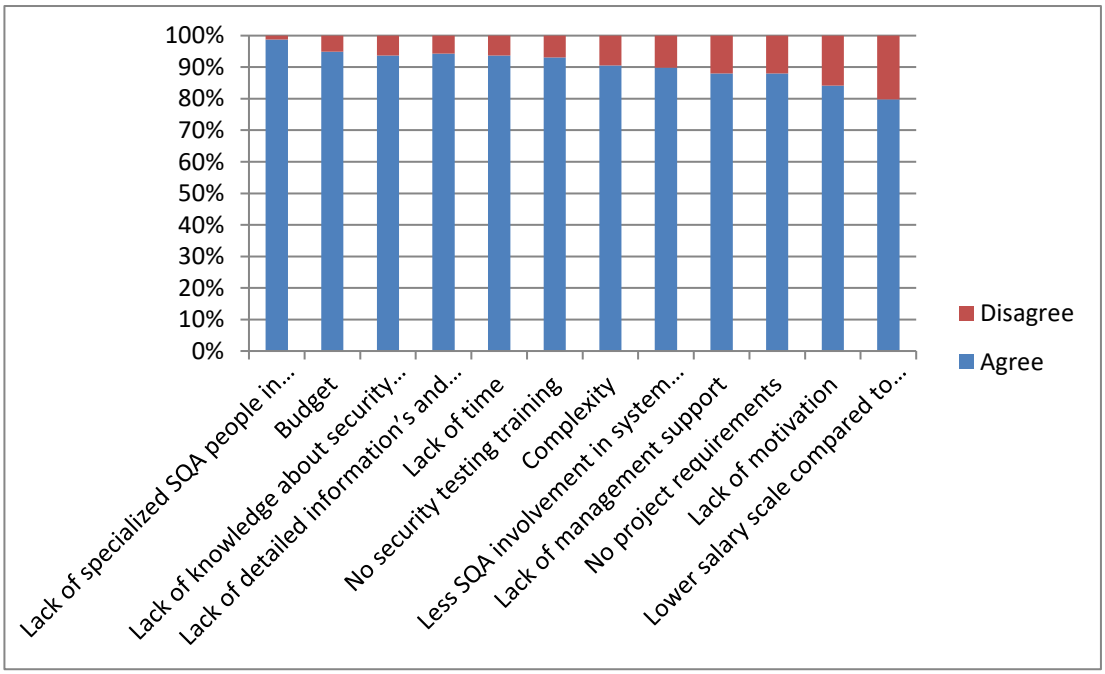

Figure 4.4: Problems distribution for the female category.

Table 4.13: Problems ranking for the female category.

Problem Description	Rank
Complexity	7
Lack of motivation	11
Lack of knowledge about security testing fundamentals	3
Lack of detailed information's and advice	4
No security testing training	6
Lack of specialized SQA people in security testing	1
Less SQA involvement in system design, requirement gathering and code review phases	8
Lack of management support	9
Lack of time	5
No project requirements	10
Lower salary scale compared to other IT professions	12
Budget	2

When analyzing both Tables 4.12 and 4.13, the problems remain the same, and only the order goes changed. Hence, this study has proven that both male and female categories agreed to the identified significant problems.

4.2.2. Analysis Based on Organization Hierarchy

When analyzing the engineer distribution, significant problems are similar to the general distribution. Only the positions and agreeableness percentage towards the problems are different; problems remained the same for engineer only distribution. Figure 4.5 depicts the distribution of problems for the engineering category. Table 4.14 shows the stack rankings of problems for the engineering category.

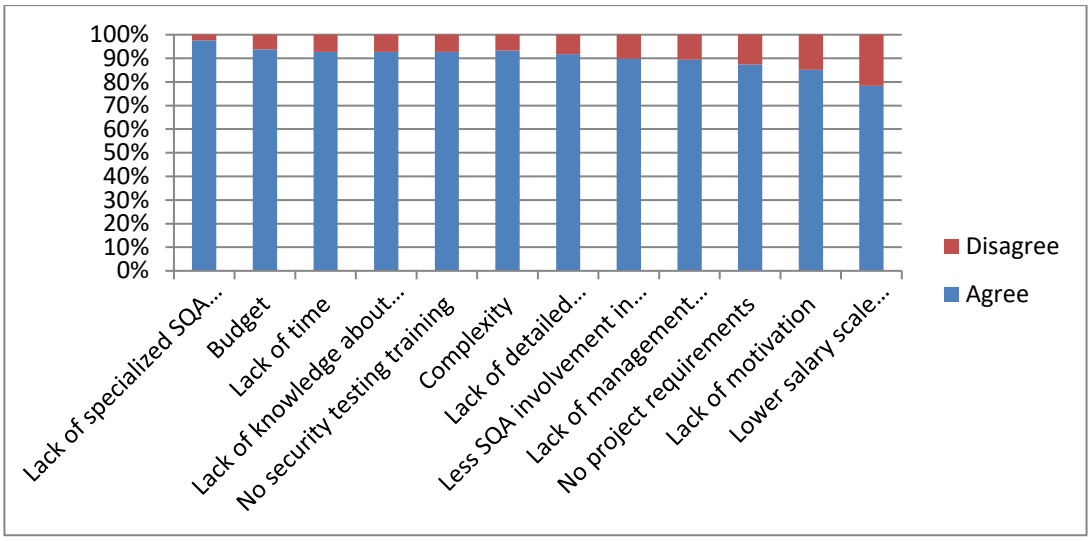

Figure 4.5: Problems distribution for the engineering category.

Table 4.14: Problems ranking for the engineering category.

Problem Description	Rank
Complexity	6
Lack of motivation	11
Lack of knowledge about security testing fundamentals	4
Lack of detailed information's and advice	7
No security testing training	5
Lack of specialized SQA people in security testing	1
Less SQA involvement in system design, requirement gathering and code review phases	8
Lack of management support	9
Lack of time	3
No project requirements	10
Lower salary scale compared to other IT professions	12
Budget	2

When analyzing the tactical management distribution, significant problems are similar to the general distribution. Only the positions and agreeableness percentage towards the problems are different; problems remained the same for tactical management only distribution. Figure 4.6 depicts the distribution of problems for the tactical management category. Table 4.15 shows the stack rankings of problems for the tactical management category.

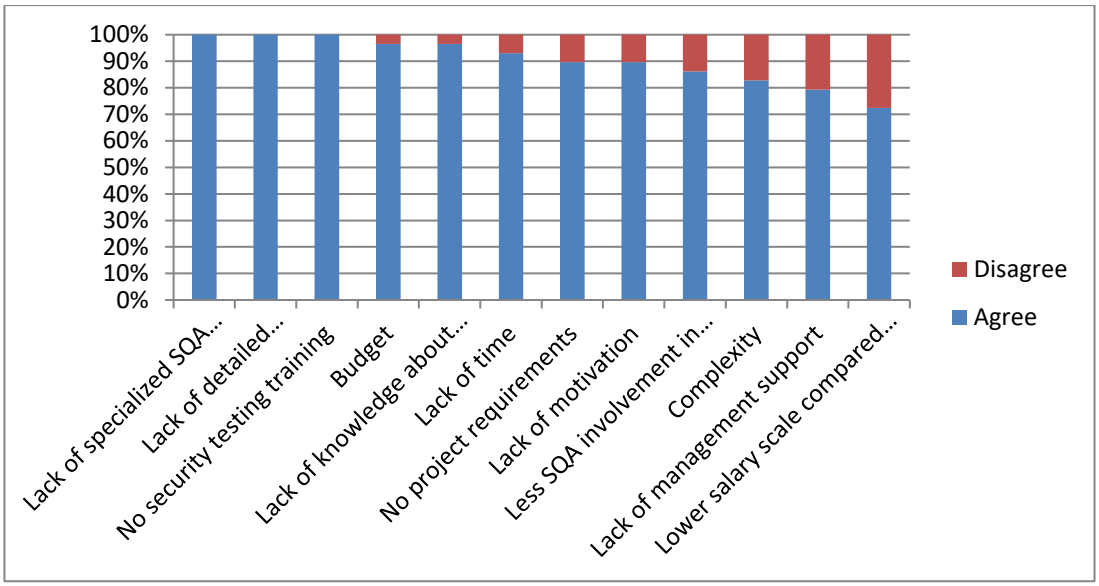

Figure 4.6: Problems distribution for the tactical management category.

Table 4.15: Problems ranking for the tactical management category.

Problem Description	Rank
Complexity	10
Lack of motivation	8
Lack of knowledge about security testing fundamentals	5
Lack of detailed information's and advice	2
No security testing training	3
Lack of specialized SQA people in security testing	1
Less SQA involvement in system design, requirement gathering and code review phases	9
Lack of management support	11
Lack of time	6
No project requirements	7
Lower salary scale compared to other IT professions	12
Budget	4

When analyzing the middle management distribution, significant problems are similar to the general distribution. Only the positions and agreeableness percentage towards the problems are different; problems remained the same for middle management only distribution. Figure 4.7 depicts the distribution of problems for the middle management category. Table 4.16 shows the stack rankings of problems for the middle management category.

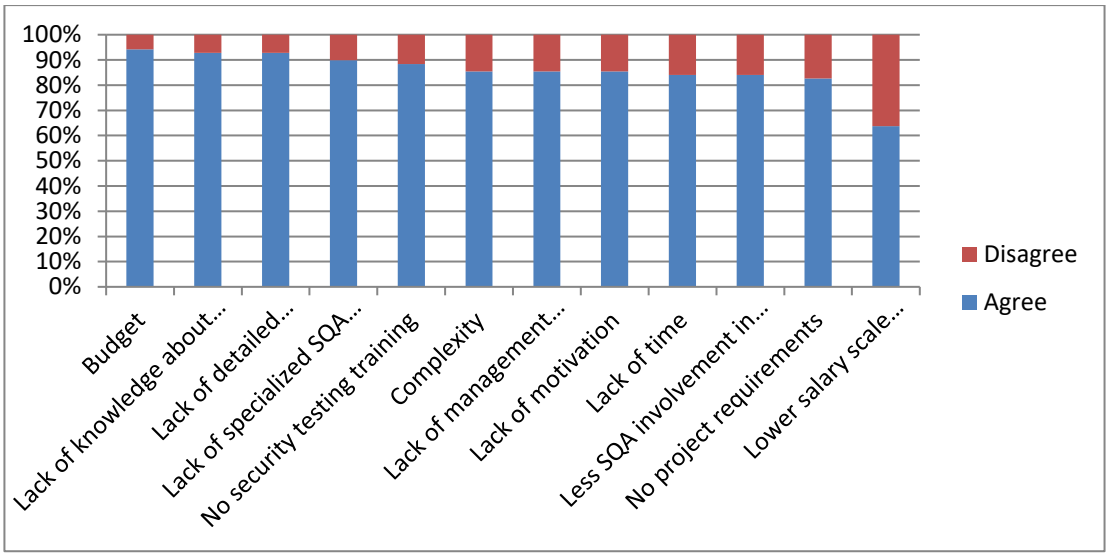

Figure 4.7: Problems distribution for the middle management category.

Table 4.16: Problems ranking for the middle management category.

Problem Description	Rank
Complexity	6
Lack of motivation	8
Lack of knowledge about security testing fundamentals	2
Lack of detailed information's and advice	3
No security testing training	5
Lack of specialized SQA people in security testing	4
Less SQA involvement in system design, requirement gathering and code review phases	10
Lack of management support	7
Lack of time	9
No project requirements	11
Lower salary scale compared to other IT professions	12
Budget	1

When analyzing the executive management distribution, significant problems are similar to the general distribution. Only the positions and agreeableness percentage towards the problems are different; problems remained the same for executive management only distribution. Figure 4.8 depicts the distribution of problems for the executive management category. Table 4.17 shows the stack rankings of problems for the executive management category.

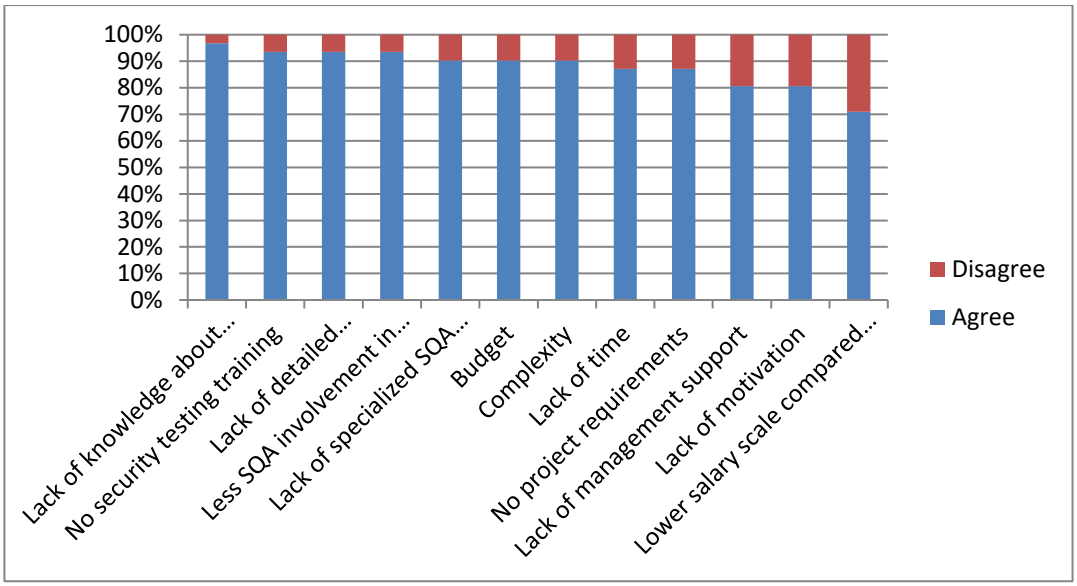

Figure 4.8: Problems distribution for the executive management category.

Table 4.17: Problems ranking for the executive management category.

Problem Description	Rank
Complexity	7
Lack of motivation	11
Lack of knowledge about security testing fundamentals	1
Lack of detailed information's and advice	3
No security testing training	2
Lack of specialized SQA people in security testing	5
Less SQA involvement in system design, requirement gathering and code review phases	4
Lack of management support	10
Lack of time	8
No project requirements	9
Lower salary scale compared to other IT professions	12
Budget	6

When analyzing Table 4.14, 4.15, 4.16, and 4.17, the problems remain the same, and only the order goes changed. Hence, this study has proven that all the organizational hierarchy level categories agreed to the identified significant problems.

4.2.3. Analysis Based on Size of QA Department

When analyzing the QA department size less than 10 distribution, significant problems are similar to the general distribution. Only the positions and agreeableness percentage towards the problems are different; problems remained the same for department size less than 10 distribution. Figure 4.9 depicts the distribution of problems for the department size of fewer than 10 categories. Table 4.18 shows the stack rankings of problems for the department size of fewer than 10 categories.

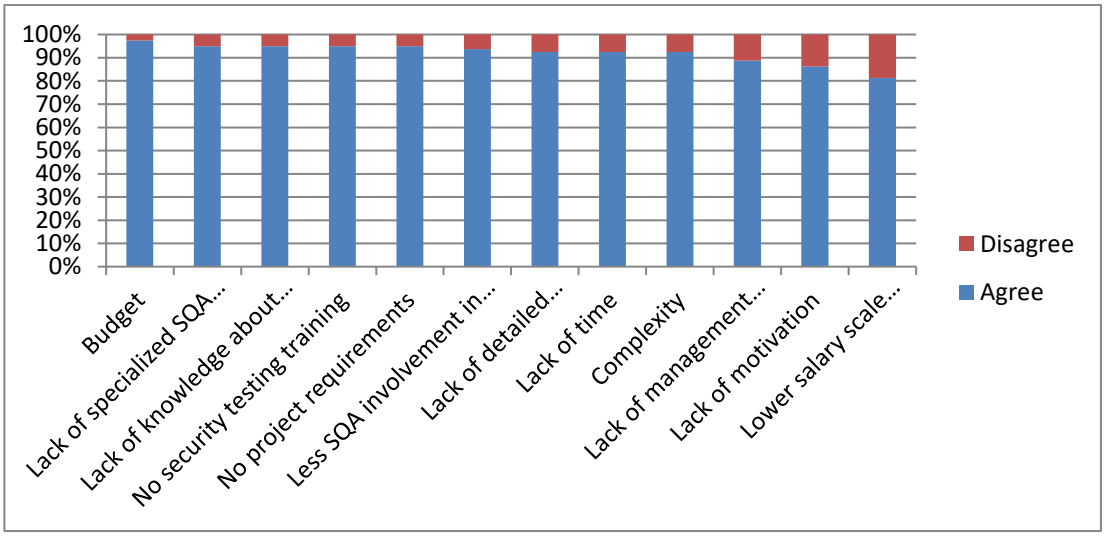

Figure 4.9: Problems distribution for QA department size less than 10 categories.

Table 4.18: Problems ranking for QA department size less than 10 categories.

Problem Description	Rank
Complexity	9
Lack of motivation	11
Lack of knowledge about security testing fundamentals	3
Lack of detailed information's and advice	7
No security testing training	4
Lack of specialized SQA people in security testing	2
Less SQA involvement in system design, requirement gathering and code review phases	6
Lack of management support	10
Lack of time	8
No project requirements	5
Lower salary scale compared to other IT professions	12
Budget	1

When analyzing the QA department size less than 50 distribution, significant problems are similar to the general distribution. Only the positions and agreeableness percentage towards the problems are different; problems remained the same for department size less than 50 distribution. Figure 4.10 depicts the distribution of problems for the department size of fewer than 50 categories. Table 4.19 shows the stack rankings of problems for the department size of fewer than 50 categories.

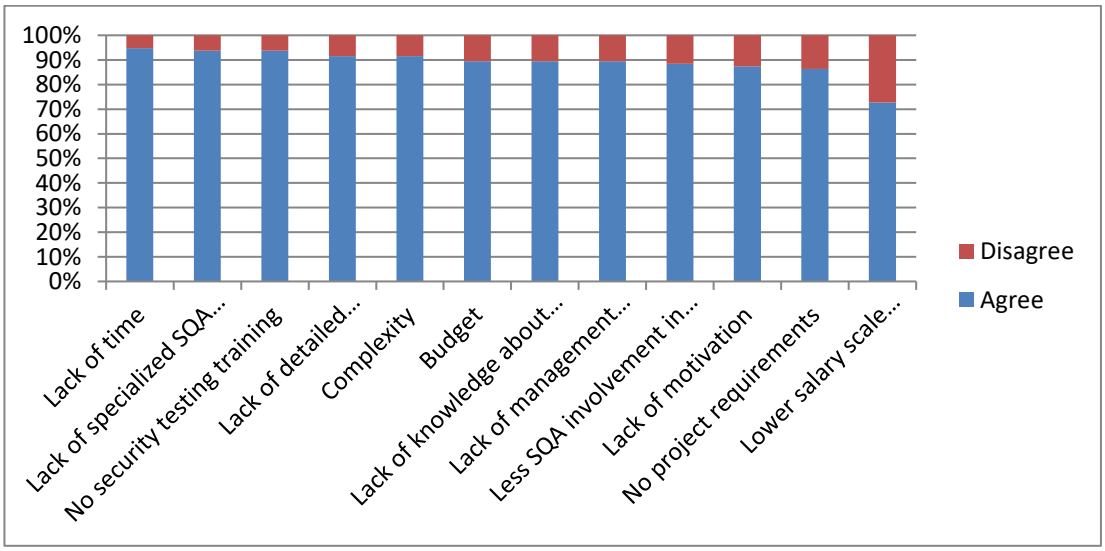

Figure 4.10: Problems distribution for QA department size less than 50 categories.

Table 4.19: Problems ranking for QA department size less than 50 categories.

Problem Description	Rank
Complexity	5
Lack of motivation	10
Lack of knowledge about security testing fundamentals	7
Lack of detailed information's and advice	4
No security testing training	3
Lack of specialized SQA people in security testing	2
Less SQA involvement in system design, requirement gathering and code review phases	9
Lack of management support	8
Lack of time	1
No project requirements	11
Lower salary scale compared to other IT professions	12
Budget	6

When analyzing the QA department size more than 50 distribution, significant problems are similar to the general distribution. Only the positions and agreeableness percentage towards the problems are different; problems remained the same for department size more than 50 distribution. Figure 4.11 depicts the distribution of problems for the department size of more than 50 categories. Table 4.20 shows the stack rankings of problems for the department size of more than 50 categories.

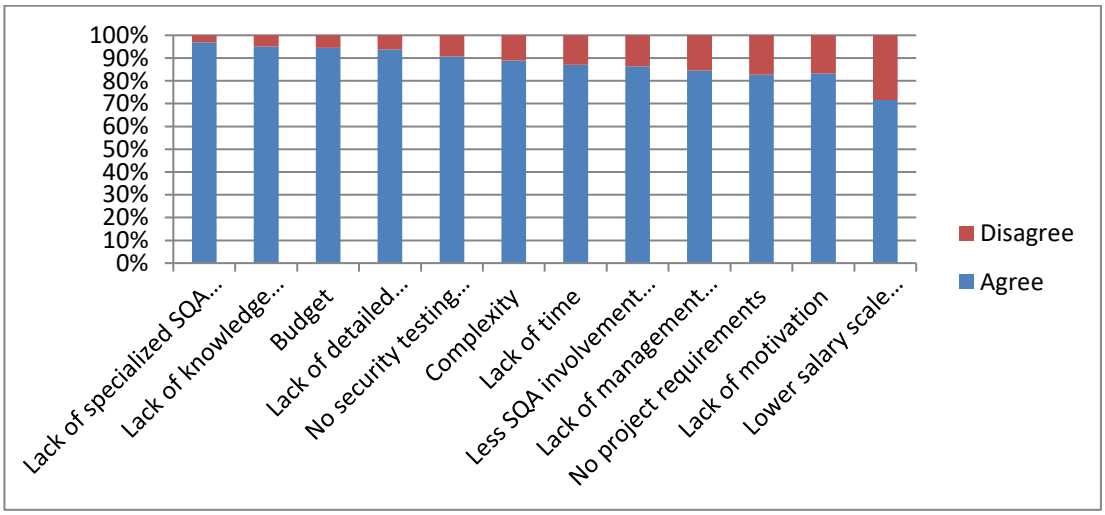

Figure 4.11: Problems distribution for QA department size more than 50 categories.

Table 4.20: Problems ranking for QA department size more than 50 categories.

Problem Description	Rank
Complexity	6
Lack of motivation	11
Lack of knowledge about security testing fundamentals	2
Lack of detailed information's and advice	4
No security testing training	5
Lack of specialized SQA people in security testing	1
Less SQA involvement in system design, requirement gathering and code review phases	8
Lack of management support	9
Lack of time	7
No project requirements	10
Lower salary scale compared to other IT professions	12
Budget	3

When analyzing Table 4.18, 4.19, and 4.20, problems remain the same, and only the order goes changed. Hence, this study has proven all the QA department size categories agreed to the identified significant problems.

4.2.4. Analysis Based on Company Type

When analyzing the product development distribution, significant problems are similar to the general distribution. Only the positions and agreeableness percentage towards the problems are different; problems remained the same for product development distribution. Figure 4.12 depicts the distribution of problems for the product development category. Table 4.21 shows the stack rankings of problems for the product development category.

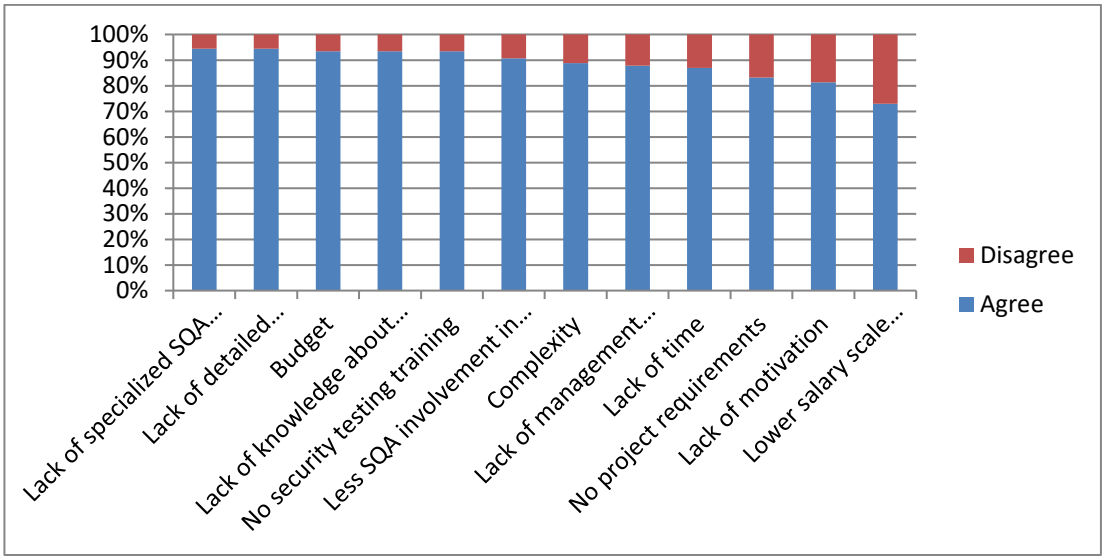

Figure 4.12: Problems distribution for the product development category.

Table 4.21: Problems ranking for the product development category.

Problem Description	Rank
Complexity	7
Lack of motivation	11
Lack of knowledge about security testing fundamentals	4
Lack of detailed information's and advice	2
No security testing training	5
Lack of specialized SQA people in security testing	1
Less SQA involvement in system design, requirement gathering and code review phases	6
Lack of management support	8
Lack of time	9
No project requirements	10
Lower salary scale compared to other IT professions	12
Budget	3

When analyzing the IT services distribution, significant problems are similar to the general distribution. Only the positions and agreeableness percentage towards the problems are different; problems remained the same for IT services distribution. Figure 4.13 depicts the distribution of problems for the IT services category. Table 4.22 shows the stack rankings of problems for the IT services category.

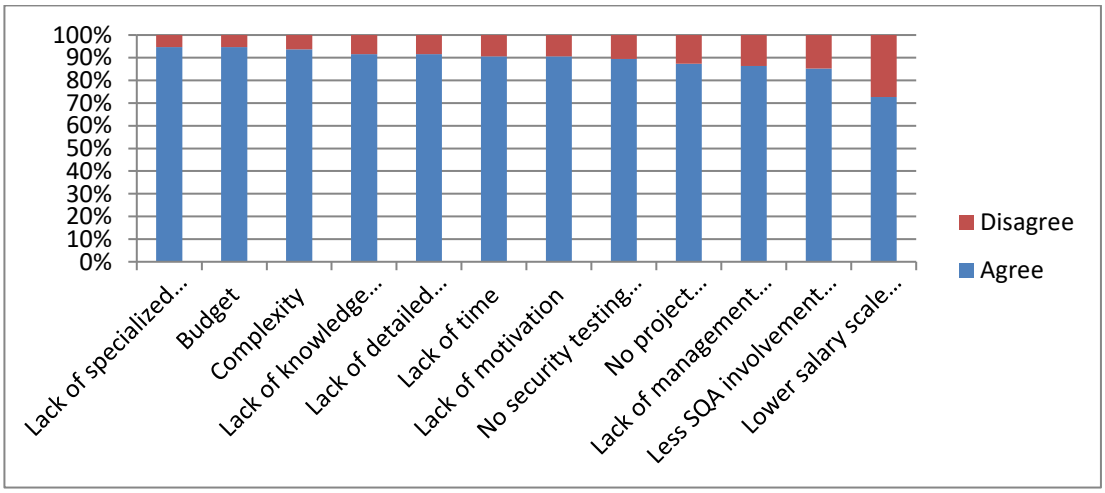

Figure 4.13: Problems distribution for the IT service category.

Table 4.22: Problems ranking for the IT service category.

Problem Description	Rank
Complexity	3
Lack of motivation	7
Lack of knowledge about security testing fundamentals	4
Lack of detailed information's and advice	5
No security testing training	8
Lack of specialized SQA people in security testing	1
Less SQA involvement in system design, requirement gathering and code review phases	11
Lack of management support	10
Lack of time	6
No project requirements	9
Lower salary scale compared to other IT professions	12
Budget	2

When analyzing product development and IT services distribution, significant problems are similar to the general distribution. Only the positions and agreeableness percentage towards the problems are different; problems remained the same for product development and IT services distribution. Figure 4.14 depicts the distribution of problems for the product development and IT services category. Table 4.23 shows the stack rankings of problems for the product development and IT services category.

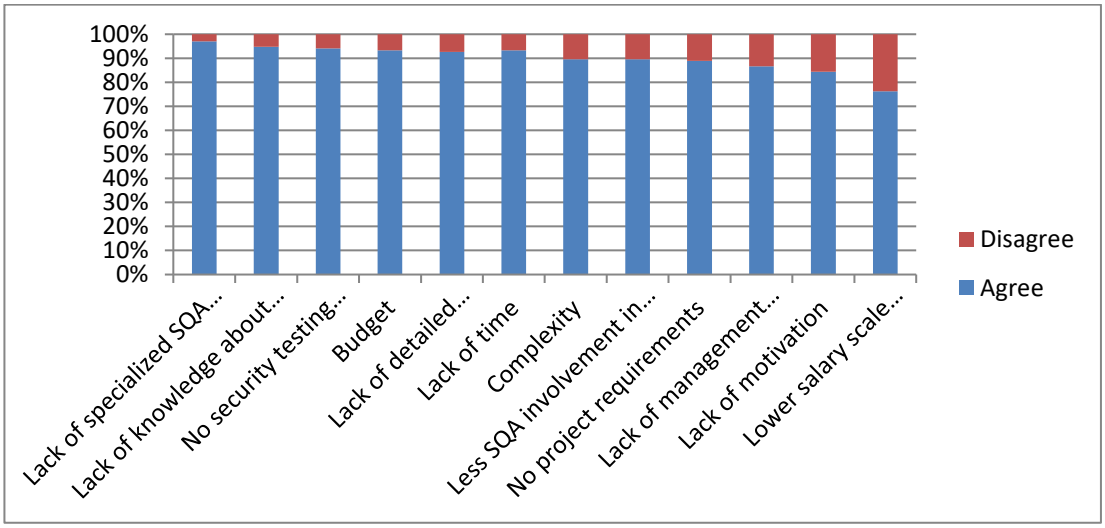

Figure 4.14: Problems distribution for both product development and IT service category.

Table 4.23: Problems ranking for product development and IT service category.

Problem Description	Rank
Complexity	7
Lack of motivation	11
Lack of knowledge about security testing fundamentals	2
Lack of detailed information's and advice	5
No security testing training	3
Lack of specialized SQA people in security testing	1
Less SQA involvement in system design, requirement gathering and code review phases	8
Lack of management support	10
Lack of time	6
No project requirements	9
Lower salary scale compared to other IT professions	12
Budget	4

When analyzing Table 4.21, 4.22, and 4.23, problems remain the same, and only the order go changed. Hence, this study has proven that all the company type categories agreed to the identified significant problems.

4.2.5. Analysis Based on Target Market

When analyzing the overseas market distribution, significant problems are similar to the general distribution. Only the positions and agreeableness percentage towards the problems are different; problems remained the same for the overseas market distribution. Figure 4.15 depicts the distribution of problems for the overseas market category. Table 4.24 shows the stack rankings of problems for the overseas market category.

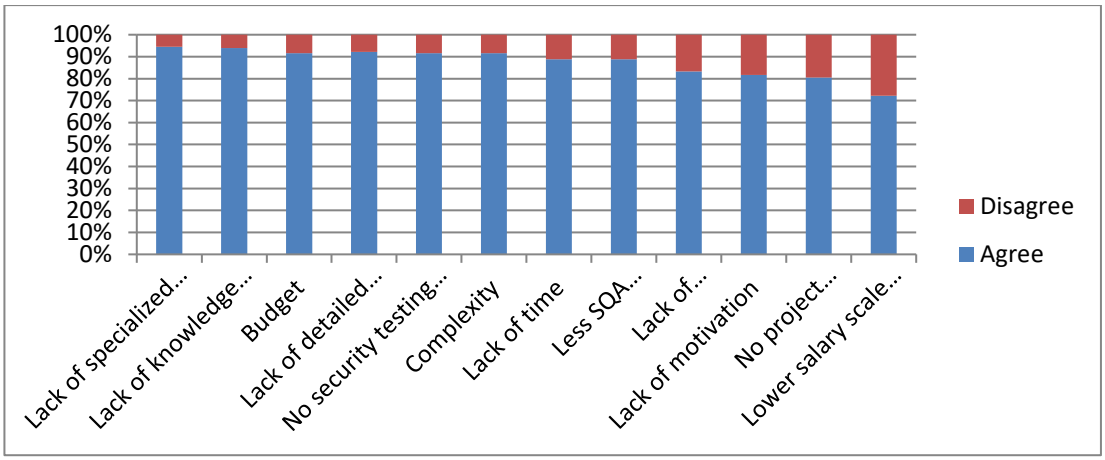

Figure 4.15: Problems distribution for the overseas category.

Table 4.24: Problems ranking for the overseas category.

Problem Description	Rank
Complexity	6
Lack of motivation	10
Lack of knowledge about security testing fundamentals	2
Lack of detailed information's and advice	4
No security testing training	5
Lack of specialized SQA people in security testing	1
Less SQA involvement in system design, requirement gathering and code review phases	8
Lack of management support	9
Lack of time	7
No project requirements	11
Lower salary scale compared to other IT professions	12
Budget	3

When analyzing the local market distribution, significant problems are similar to the general distribution. Only the positions and agreeableness percentage towards the problems are different; problems remained the same for the local market distribution. Figure 4.16 depicts the distribution of problems for the local market category. Table 4.25 shows the stack rankings of problems for the local market category.

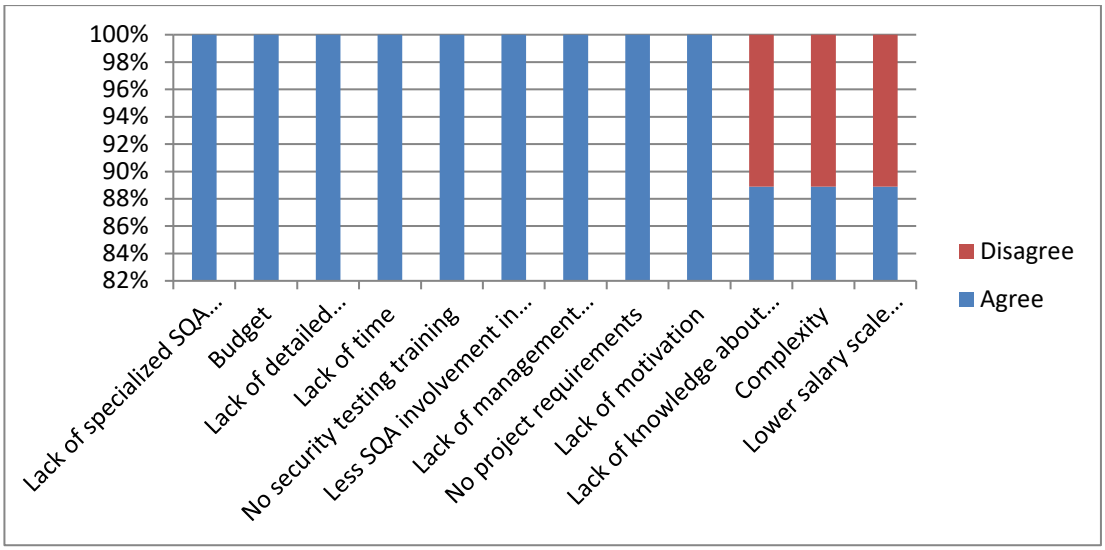

Figure 4.16: Problems distribution for the local market category.

Table 4.25: Problems ranking for the local market category.

Problem Description	Rank
Complexity	11
Lack of motivation	9
Lack of knowledge about security testing fundamentals	10
Lack of detailed information's and advice	3
No security testing training	5
Lack of specialized SQA people in security testing	1
Less SQA involvement in system design, requirement gathering and code review phases	6
Lack of management support	7
Lack of time	4
No project requirements	8
Lower salary scale compared to other IT professions	12
Budget	2

When analyzing both overseas and local market distribution, significant problems are similar to the general distribution. Only the positions and agreeableness percentage towards the problems are different; problems remained the same for both overseas and local market distribution. Figure 4.16 depicts the distribution of problems for the overseas and local market category. Table 4.25 shows the stack rankings of problems for the overseas and local market category.

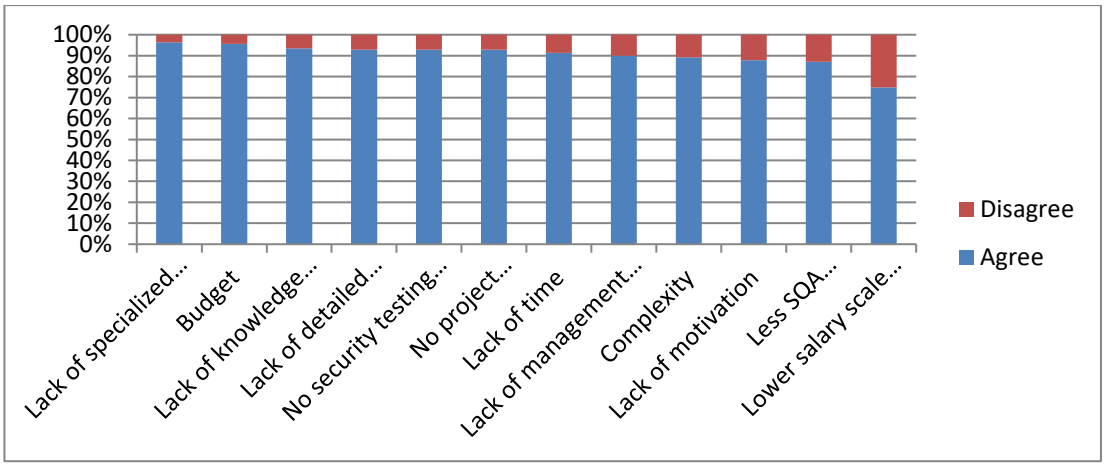

Figure 4.17: Problems distribution for the overseas and local market category.

Table 4.26: Problems ranking for the overseas and local market category.

Problem Description	Rank
Complexity	9
Lack of motivation	10
Lack of knowledge about security testing fundamentals	3
Lack of detailed information's and advice	4
No security testing training	5
Lack of specialized SQA people in security testing	1
Less SQA involvement in system design, requirement gathering and code review phases	11
Lack of management support	8
Lack of time	7
No project requirements	6
Lower salary scale compared to other IT professions	12
Budget	2

When analyzing Table 4.24, 4.25, and 4.26, the problems remain the same, and only the order goes changed. Hence, this study has proven that all the target market categories agreed to the identified significant problems.

Identify ngisignificant problems faced by SQA professionals in software security testing and the suggestions to overcome those problems is essential to create the final strategy. Table 4.27 shows the average ranking summary of the identified significant problems based on demographic data.

Table 4.27: Ranking summary of the identified problems based on demographic data.

Problem Description	Rank
Lack of specialized SQA people in security testing	1
Budget	2
Lack of knowledge about security testing fundamentals	3
Lack of detailed information's and advice	4
No security testing training	5
Lack of time	6
Complexity	7
Less SQA involvement in system design, requirement gathering and code review phases	8
Lack of management support	9
No project requirements	10
Lack of motivation	11
Lower salary scale compared to other IT professions	12

4.2.6. Analysis of Identified Significant Problems

"Lack of specialized SQA people in security testing" is considered as the most crucial problem. It is standing at the first rank on the significant problem list. As per the respondent's comments, there is a low resource availability for specialize QA activities. In the present, customers are more focused on non-functional activities than functional activities. Due to this matter, the QA department should contain security, performance, and automation, specialized team members. However, still, the companies do not have the required number of SQA resources to cater to the above requirement. Hence, 86% of the respondents agreed that 'Lack of specialized SQA people in security testing' is a problem for SQA professionals in software security testing. Figure 4.18 shows the distribution of the agreeableness towards the problem. Two critical suggestions made to overcome this problem. The first one was to "Form a dedicated QA security taskforce to develop and retain the security testing mindset among SQA people." 91% of the respondents agreed to the suggestion made by the

researcher. The second one was to “Recruit detail-oriented and experienced SQA professionals.” 78% of the respondents agreed to the suggestion made by the researcher. Table 4.28 shows the distribution of the agreeableness towards the researcher’s suggestions to overcome the problem. Figure 4.19 shows the gender-wise distribution of the agreeableness towards ‘Lack of specialized SQA people in security testing’ as a problem. Figure 4.20 shows the organization level-wise distribution of the agreeableness towards ‘Lack of specialized SQA people in security testing’ as a problem. Figure 4.21 shows the size of the QA department wise distribution of the agreeableness towards ‘Lack of specialized SQA people in security testing’ as a problem.

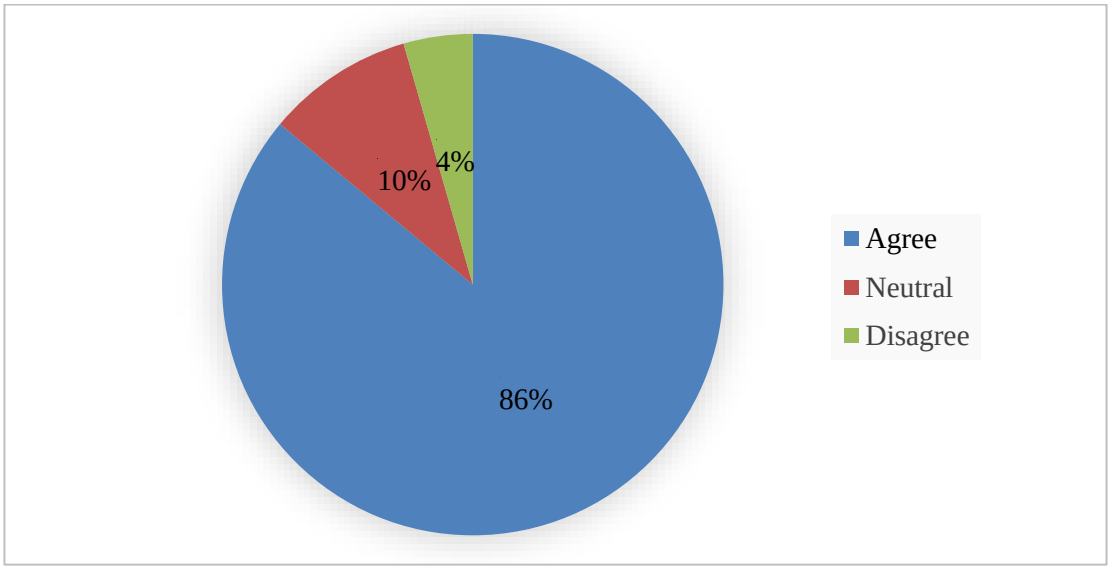

Figure 4.18: The agreeable extent of the participants for the “Lack of specialized SQA people in security testing” as a problem.

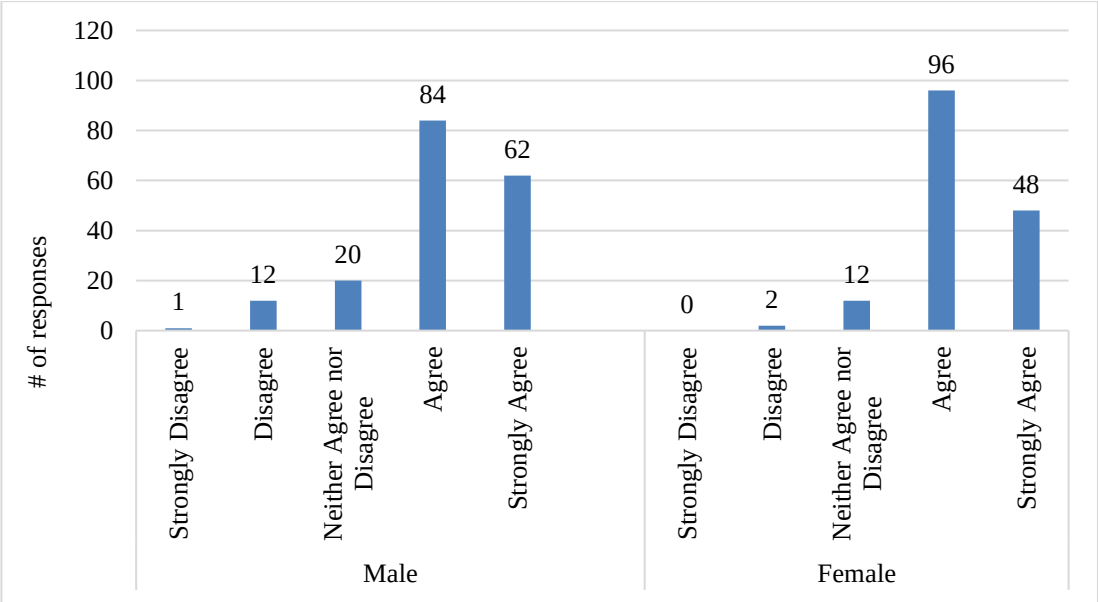

Figure 4.19: Gender-wise analysis on the distribution of "Lack of specialized SQA people in security testing" in the online survey.

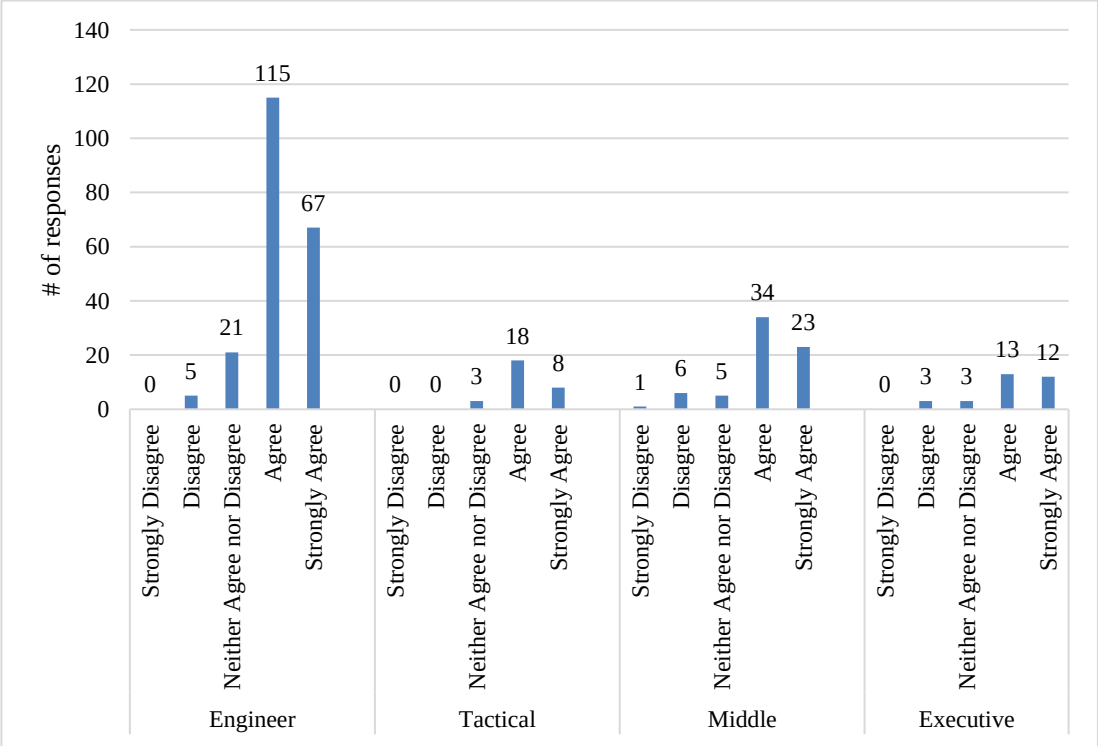

Figure 4.20: Organization level-wise analysis on the distribution of "Lack of specialized SQA people in security testing" in the online survey.

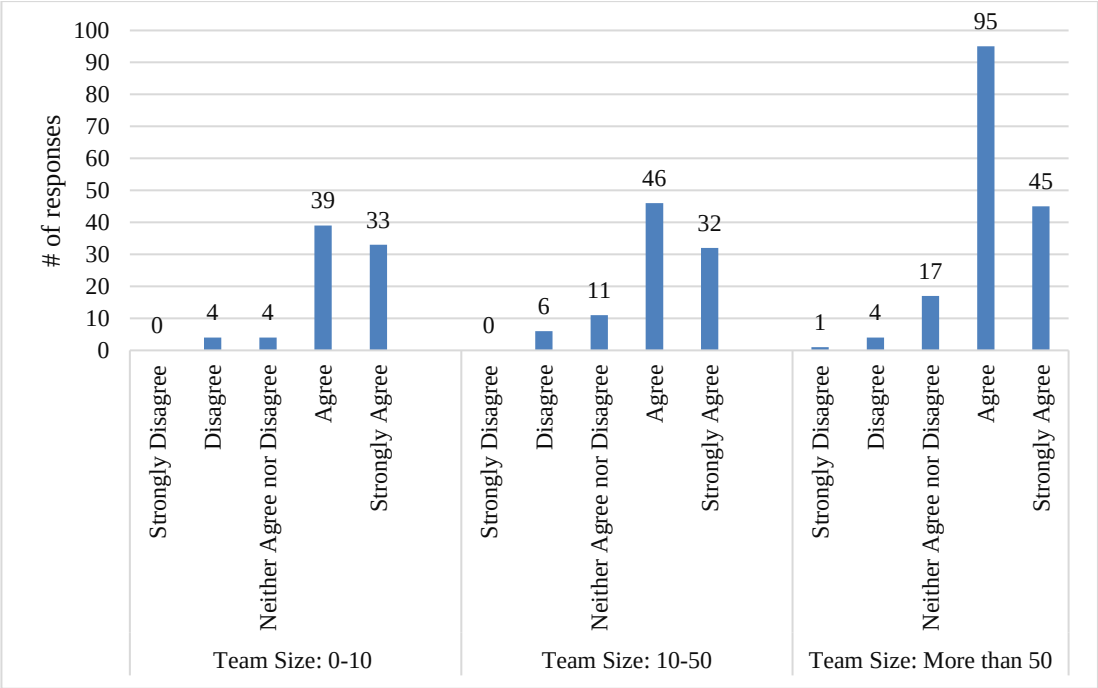

Figure 4.21: Size of the QA department wise analysis on the distribution of "Lack of specialized SQA people in security testing" in the online survey.

"Form a dedicated QA security taskforce to develop and retain the security testing mindset among SQA people" and "Recruit detail-oriented and experienced SQA professionals" were the researcher's suggestions to over the problem on 'Lack of specialized SQA people in security testing.' Table 4.28 shows the distribution of the agreeableness towards the researcher's suggestions to overcome the problem.

Table 4.28: Suggestions distribution made to overcome 'Lack of specialized SQA people in security testing' problem in the online survey.

Solution Description	Agree	Neutral	Disagree
Form a dedicated QA security taskforce to develop and retain the security testing mindset among SQA people.	91%	8%	1%
Recruit detail-oriented and experienced SQA professionals.	78%	19%	4%

The problem with "Budget" has a different standing in the online survey. Only 73% of the respondents agreed that 'Budget' is a problem for SQA professionals in software security testing. 21% of respondents are neither agreed nor disagreed with the problem. Since the research focus is to find out a specific strategy to develop the SQA mindset in software security testing, the researcher considers that 21% as a positive response.

Hence the problem is standing at the second place of the significant problem list. The reason behind this problem is that the management tends to invest less budget on SQA related activities. Also, there is a low demand for investing in different SQA tools, separate test environments. As per a respondent's comment, most of the testing budget now investing in new development projects based on the higher management's prioritization. However, 6% of respondents think that 'Budget' is not a problem for SQA professionals. Figure 4.22 shows the distribution of the agreeableness towards 'Budget' as a problem. The suggestion made to overcome this problem in the online survey was 'Allocate sufficient funds in the budget to provide proper SQA resources.' 85% of the respondents agreed to the suggestion made by the researcher. Figure 4.23 shows the gender-wise distribution of the agreeableness towards 'Budget' as a problem. Figure 4.24 shows the organization level-wise distribution of the agreeableness towards 'Budget' as a problem. Figure 4.25 shows the size of the QA department wise distribution of the agreeableness towards 'Budget' as a problem.

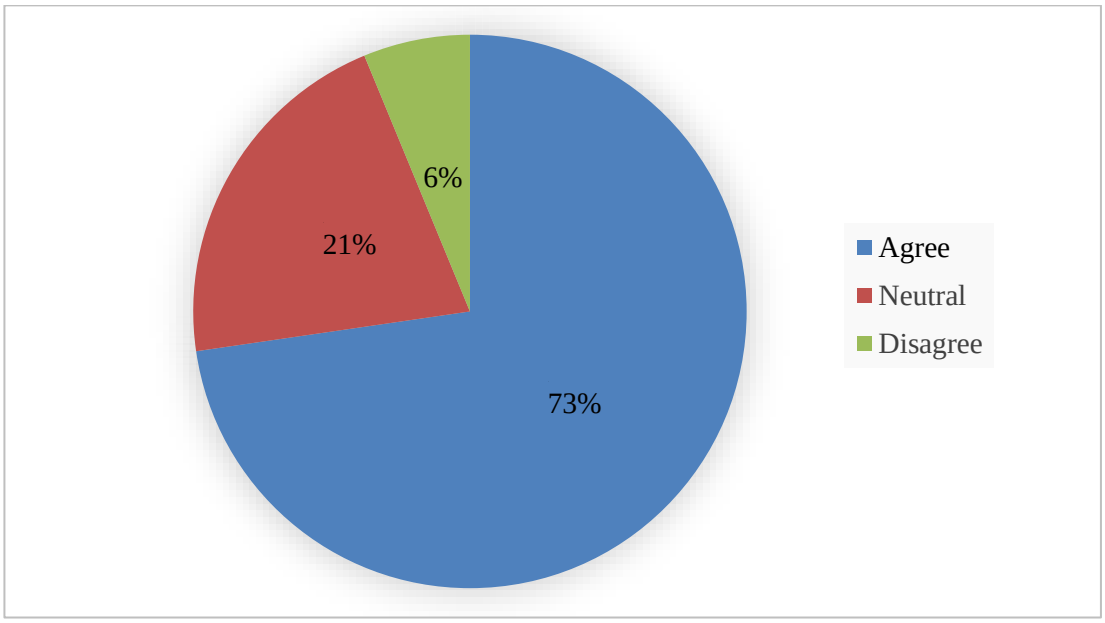

Figure 4.22: The agreeable extent of the participants for the 'Budget' as a problem.

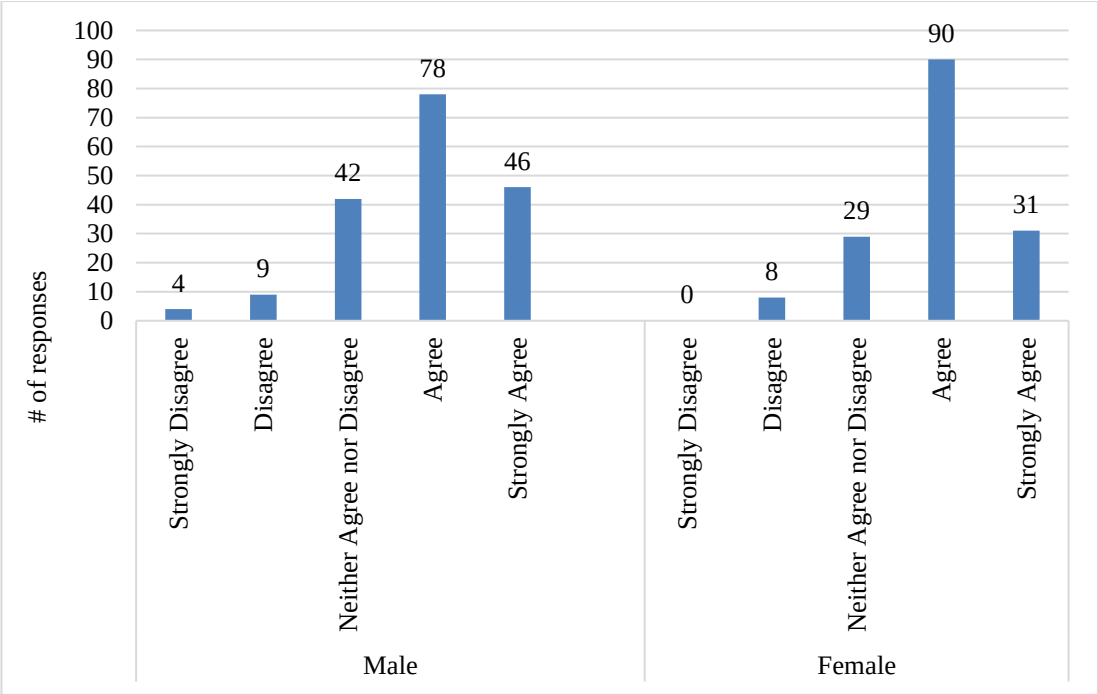

Figure 4.23: Gender-wise analysis of the distribution of 'Budget' in the online survey.

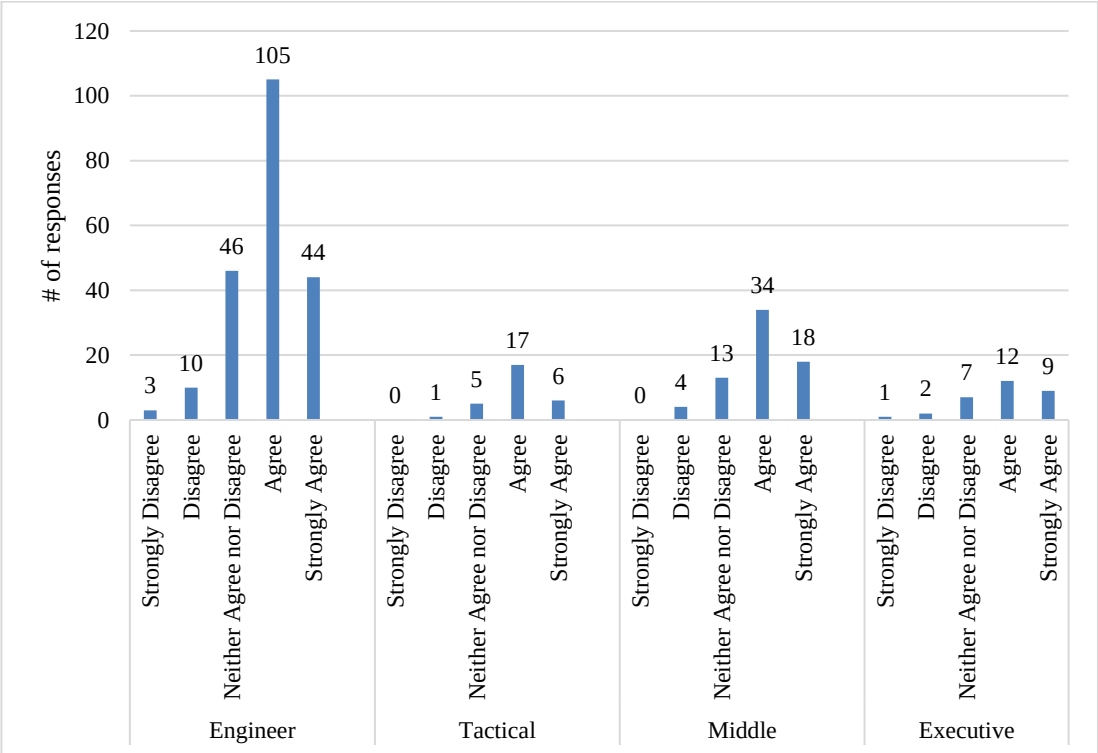

Figure 4.24: Organization level-wise analysis of the distribution of 'Budget' in the online survey.

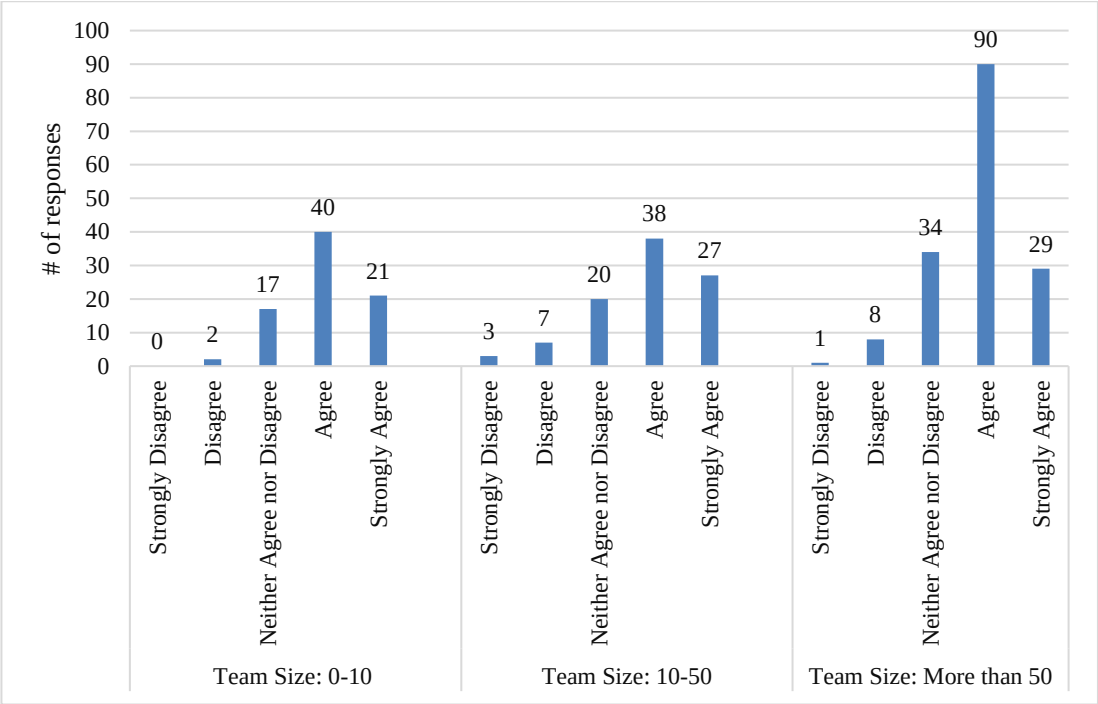

Figure 4.25: Size of the QA department wise analysis on the distribution of 'Budget' in the online survey.

'Provide proper SQA resources regardless of profit margin.' was the researcher's suggestion to over the problem on 'Budget.' Table 4.29 shows the distribution of the agreeableness towards the researcher's suggestion to overcome the problem.

Table 4.29: Suggestion distribution made to overcome the 'Budget' problem in the online survey.

Solution Description	Agree	Neutral	Disagree
Provide proper SQA resources regardless of the profit margin.	85%	12%	3%

Problem on "Lack of knowledge about security testing fundamentals" was standing third place while having 84% of agreed responses. As per the respondent's comments, there are many types of security vulnerabilities that cannot and will not be detected using tools, and the use of a scanning tool does not at all replace the need for manual security testing. However, 7% of respondents think that 'Lack of knowledge about security testing fundamentals' is not a problem for SQA professionals in software security testing. Figure 4.26 shows the distribution of the agreeableness towards 'Lack

of knowledge about security testing fundamentals.' 94% of the respondents equally agreed to the two suggestions made by the researcher. Figure 4.27 shows the gender-wise distribution of the agreeableness towards 'Lack of knowledge about security testing fundamentals' as a problem. Figure 4.28 shows the organization level-wise distribution of the agreeableness towards 'Lack of knowledge about security testing fundamentals' as a problem. Figure 4.29 shows the size of the QA department wise distribution of the agreeableness towards 'Lack of knowledge about security testing fundamentals' as a problem.

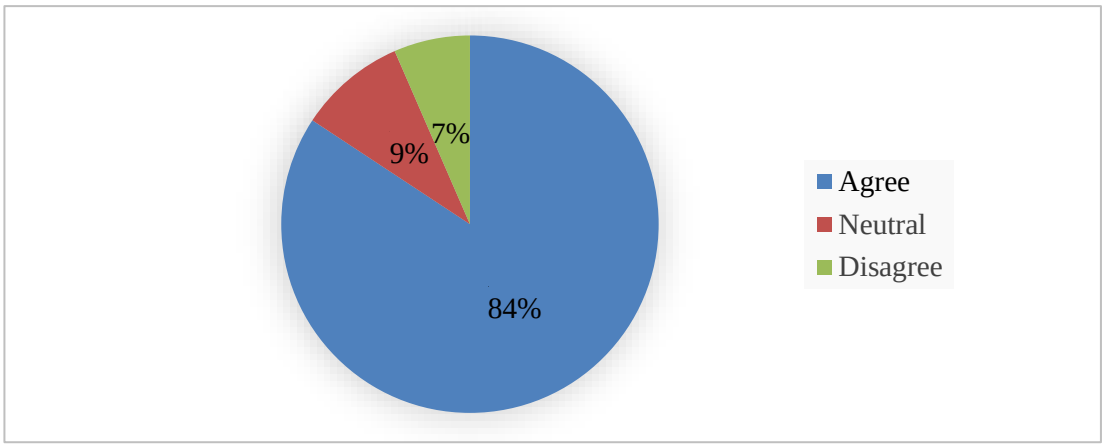

Figure 4.26: The agreeable extent of the participants for the 'Lack of knowledge about security testing fundamentals' as a problem.

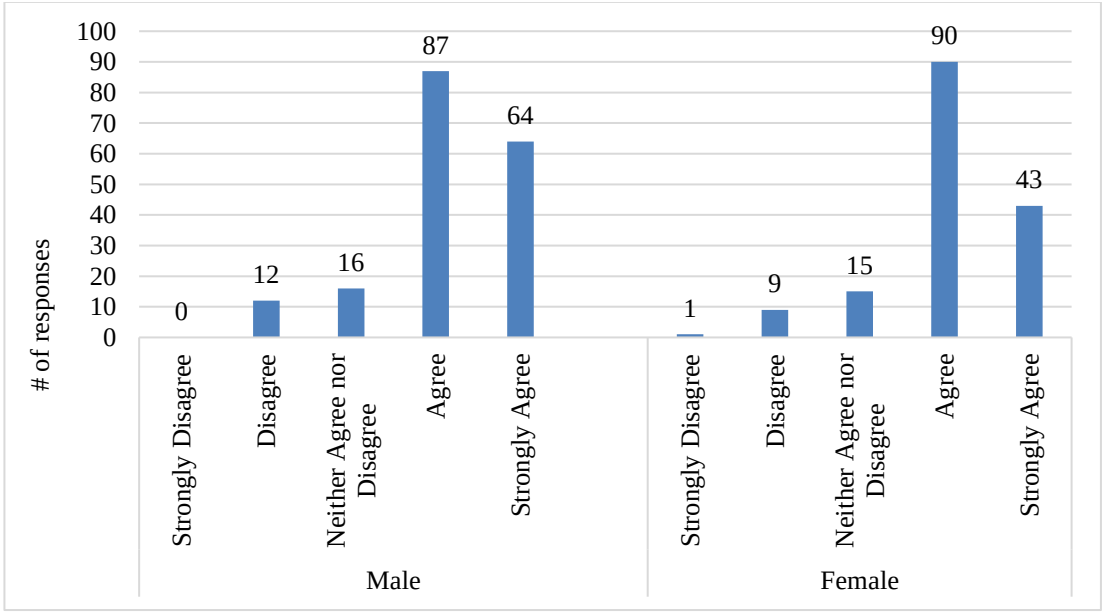

Figure 4.27: Gender-wise analysis on the 'Lack of knowledge about security testing fundamentals' in the online survey distribution.

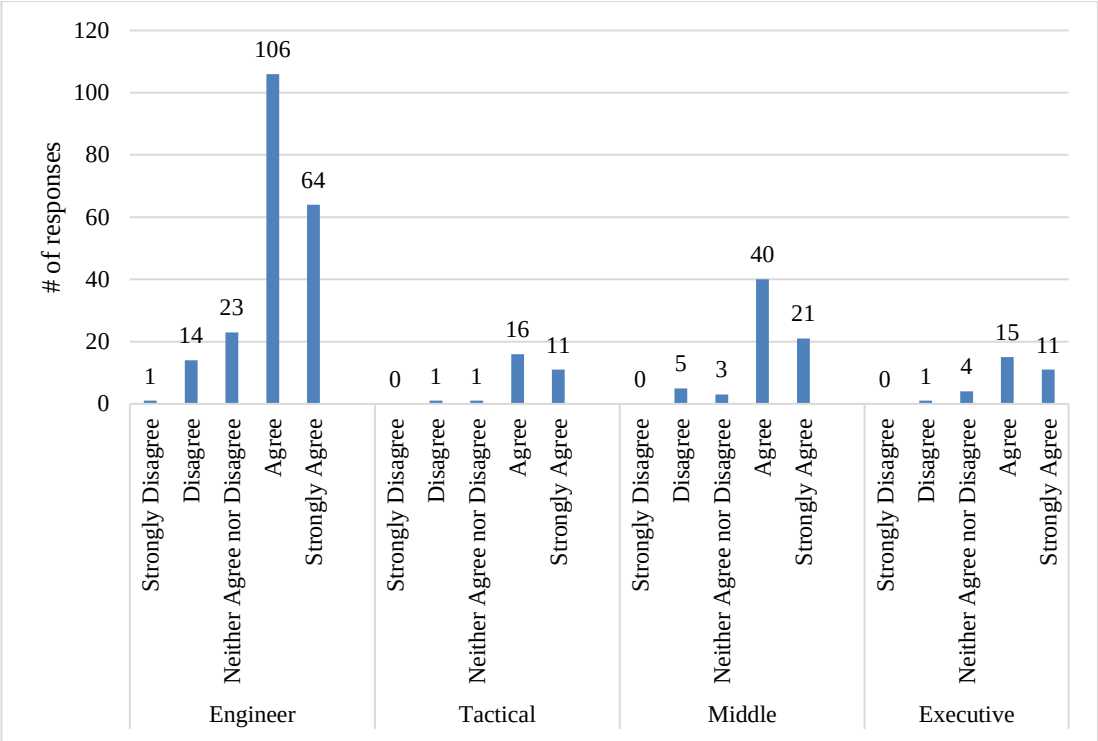

Figure 4.28: Organization level-wise analysis on the 'Lack of knowledge about security testing fundamentals' in the online survey distribution.

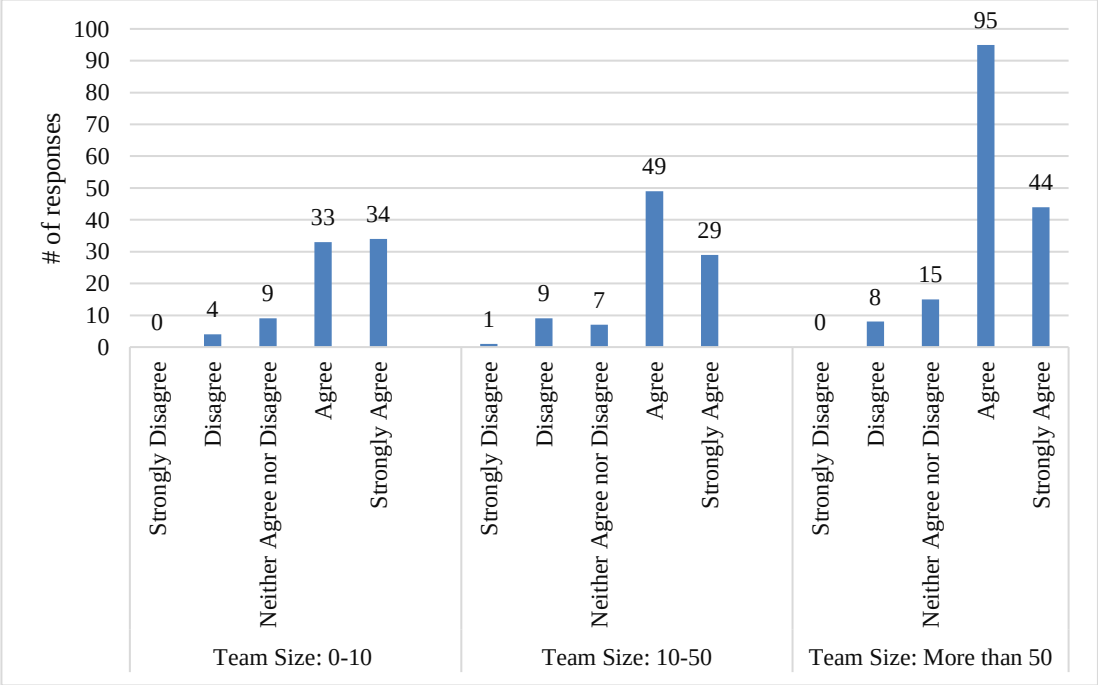

Figure 4.29: Size of the QA department wise analysis on the 'Lack of knowledge about security testing fundamentals' in the online survey distribution.

The suggestions made to overcome this problem in the online survey were 'Familiarize and adapt security testing fundamentals, protocols, tools, and methods to fit within existing processes' and 'Maintain a security testing knowledge portal.' Table 4.30 shows the distribution of the agreeableness towards the researcher's suggestion to overcome the problem.

Table 4.30: Suggestions distribution made to overcome 'Lack of knowledge about security testing fundamentals' problem in the online survey.

Solution Description	**Agree**	**Neutral**	**Disagree**
Familiarize and adapt security testing fundamentals, protocols, tools, and methods to fit within existing processes.	94%	6%	0%
Maintain a security testing knowledge portal.	94%	5%	1%

"Lack of detailed information and advice" is standing at the fourth rank on the significant problem list. As per the respondent's comments, when QA begins to accumulate knowledge, there is no much-detailed information on basic security testing concepts. No lessons on how to use the automatic scanners. Hence, 80% of the respondents agreed that 'Lack of detailed information and advice' is a problem for SQA professionals in software security testing. Figure 4.30 shows the distribution of the agreeableness towards the problem. The suggestions made to overcome this problem were "Advice SQA professionals to approach security testing with a risk management mindset" and "Working in tandem with architects and IT security teams to map out security vulnerabilities." 87% of the respondents agreed to the first suggestion made by the researcher. 82% of the respondents agreed to the second suggestion made by the researcher. Figure 4.31 shows the gender-wise distribution of the agreeableness towards 'Lack of detailed information and advice' as a problem. Figure 4.32 shows the organization level-wise distribution of the agreeableness towards 'Lack of detailed information and advice' as a problem. Figure 4.33 shows the size of the QA department wise distribution of the agreeableness towards 'Lack of detailed information and advice' as a problem.

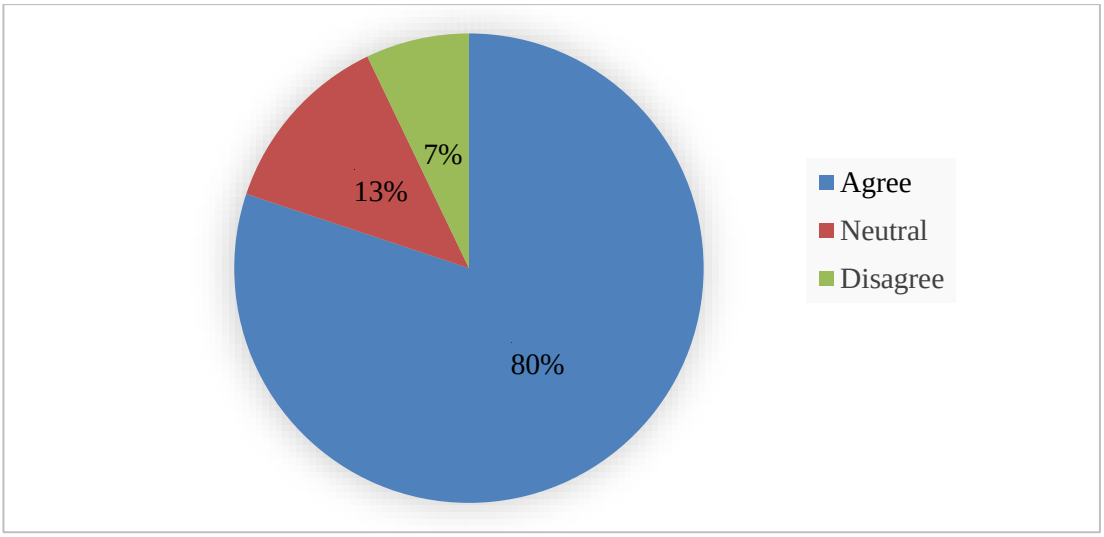

Figure 4.30: The agreeable extent of the participants for the ‘Lack of detailed information and advice’ as a problem.

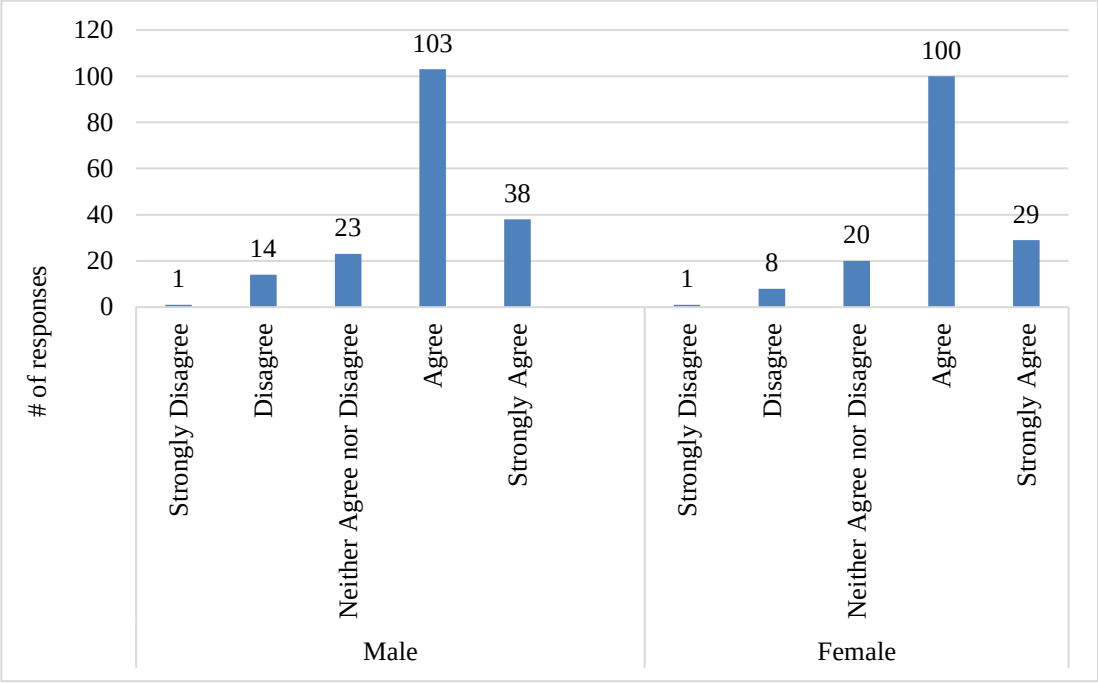

Figure 4.31: Gender-wise analysis on the ‘Lack of detailed information and advice’ in the online survey distribution.

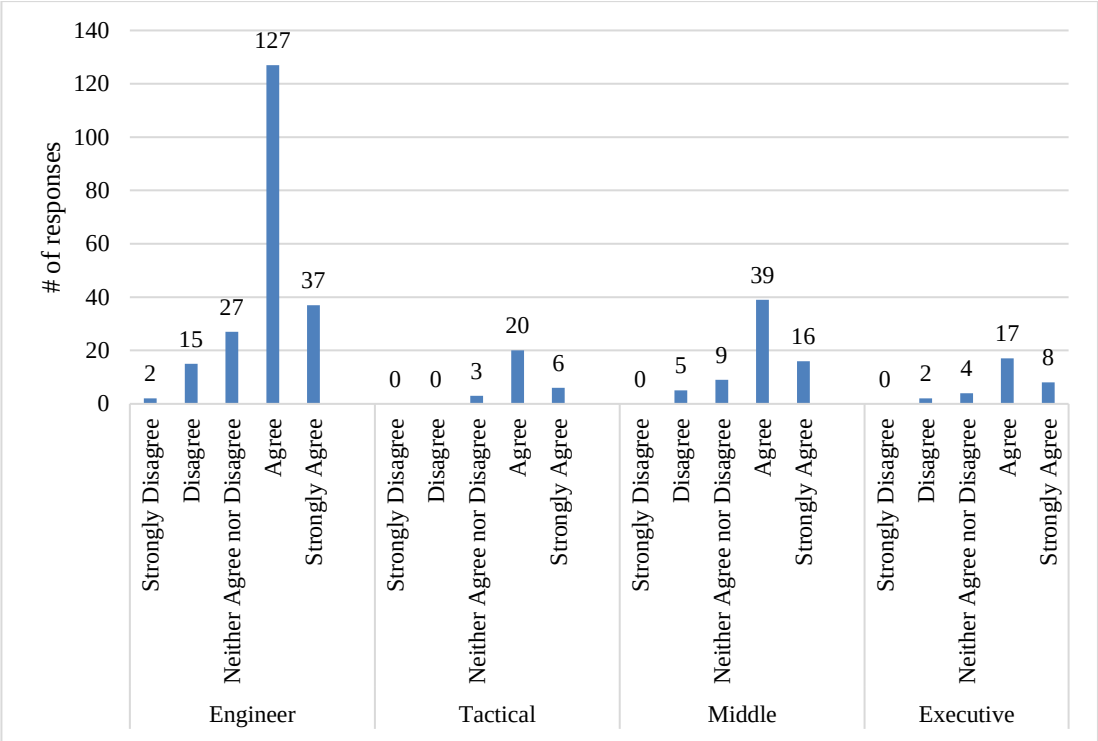

Figure 4.32: Organization level-wise analysis of the 'Lack of detailed information and advice' in the online survey distribution.

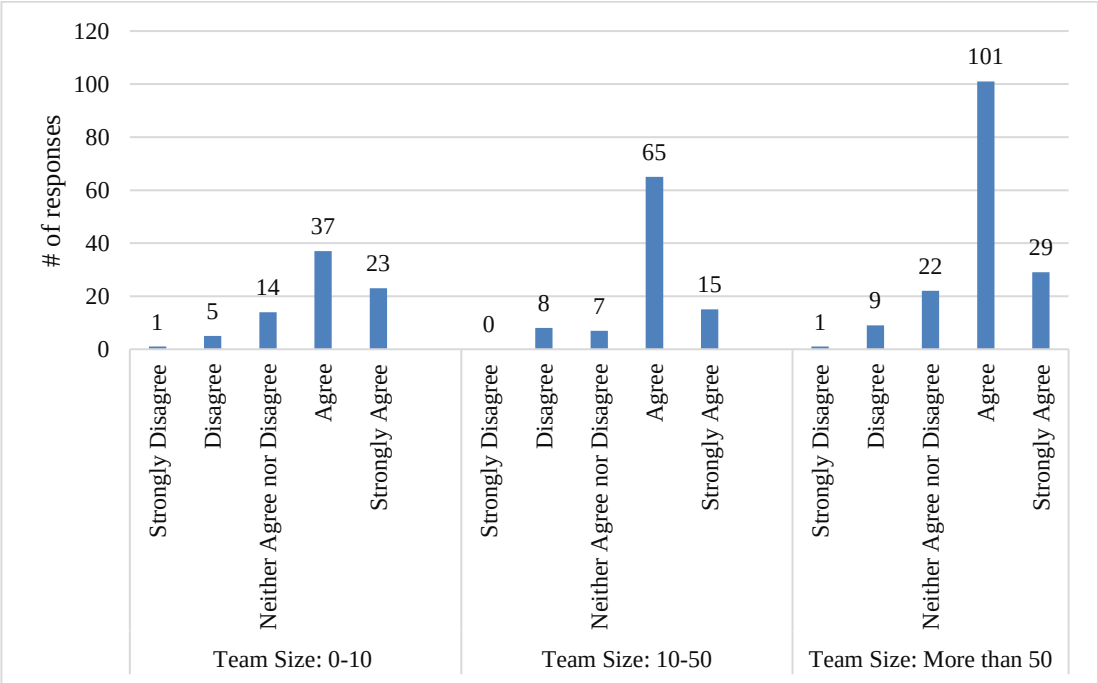

Figure 4.33: Size of the QA department wise analysis on the 'Lack of detailed information and advice' in the online survey distribution.

The suggestions made to overcome this problem in the online survey were "Advice SQA professionals to approach software security testing with a risk management mindset" and "Working in tandem with architects and IT security teams to map out security vulnerabilities." Table 4.31 shows the distribution of the agreeableness towards the researcher's suggestion to overcome the problem.

Table 4.31: Suggestions distribution made to overcome 'Lack of detailed information and advice' problem in the online survey.

Solution Description	**Agree**	**Neutral**	**Disagree**
Advice SQA professionals to approach security testing with a risk management mindset.	87%	12%	1%
Working in tandem with architects and IT security teams to map out security vulnerabilities.	82%	15%	3%

"No security testing training" is standing at the fifth rank on the significant problem list. As per the respondent's comments, without enough experience, it is complicated to understand why we need all the security testing tools, what are the advantages and disadvantages of each of them, and, most importantly, when we should use one instead of the other. Hence, 79% of the respondents agreed that 'No security testing training' is a problem for SQA professionals in software security testing. Figure 4.34 shows the distribution of the agreeableness towards the problem. The suggestion made to overcome this problem was, "Introduced more security testing meet-ups and training for SQA people." 93% of the respondents agreed to the suggestion made by the researcher. Figure 4.35 shows the gender-wise distribution of the agreeableness towards 'No security testing training' as a problem. Figure 4.36 shows the organization level-wise distribution of the agreeableness towards 'No security testing training' as a problem. Figure 4.37 shows the size of the QA department wise distribution of the agreeableness towards 'No security testing training' as a problem.

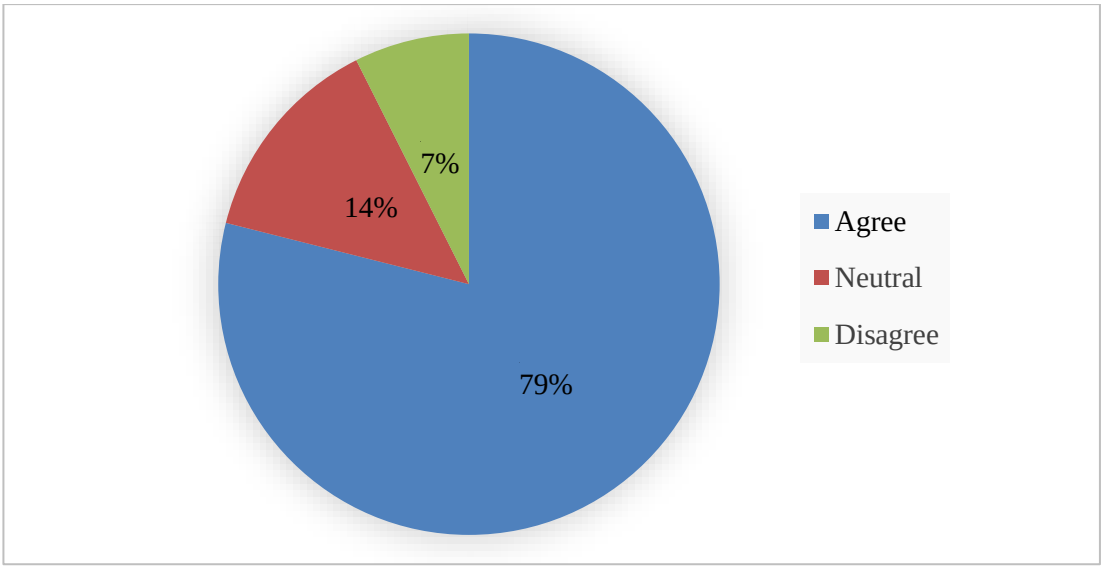

Figure 4.34: The agreeable extent of the participants for the 'No security testing training' as a problem.

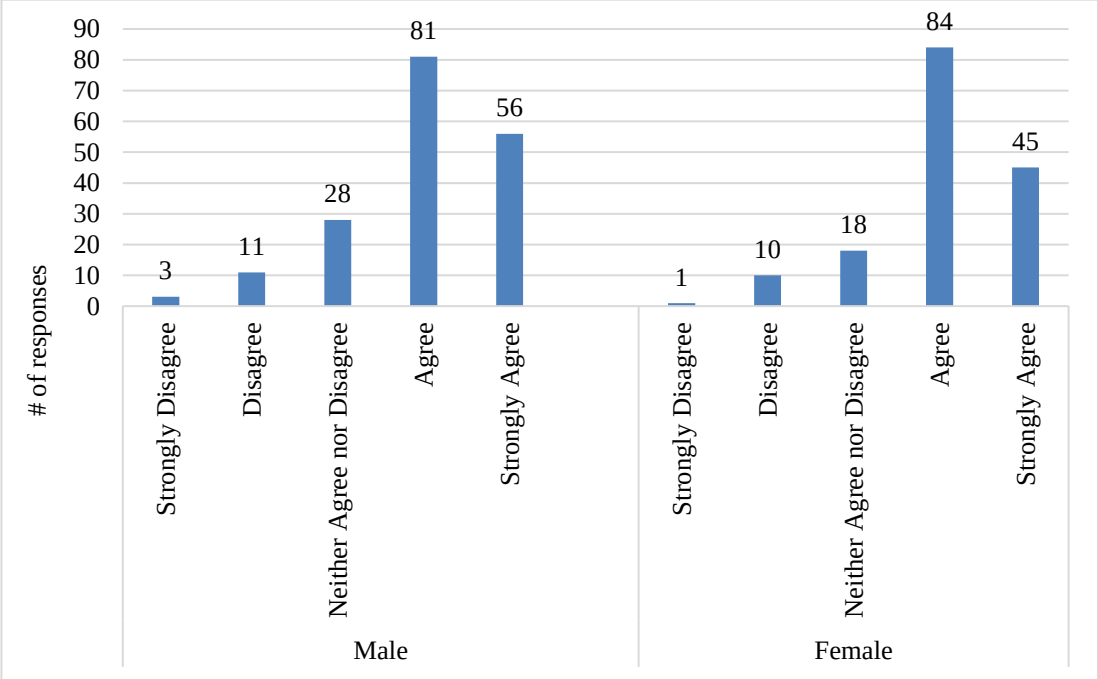

Figure 4.35: Gender-wise analysis of the 'No security testing training' in the online survey distribution.

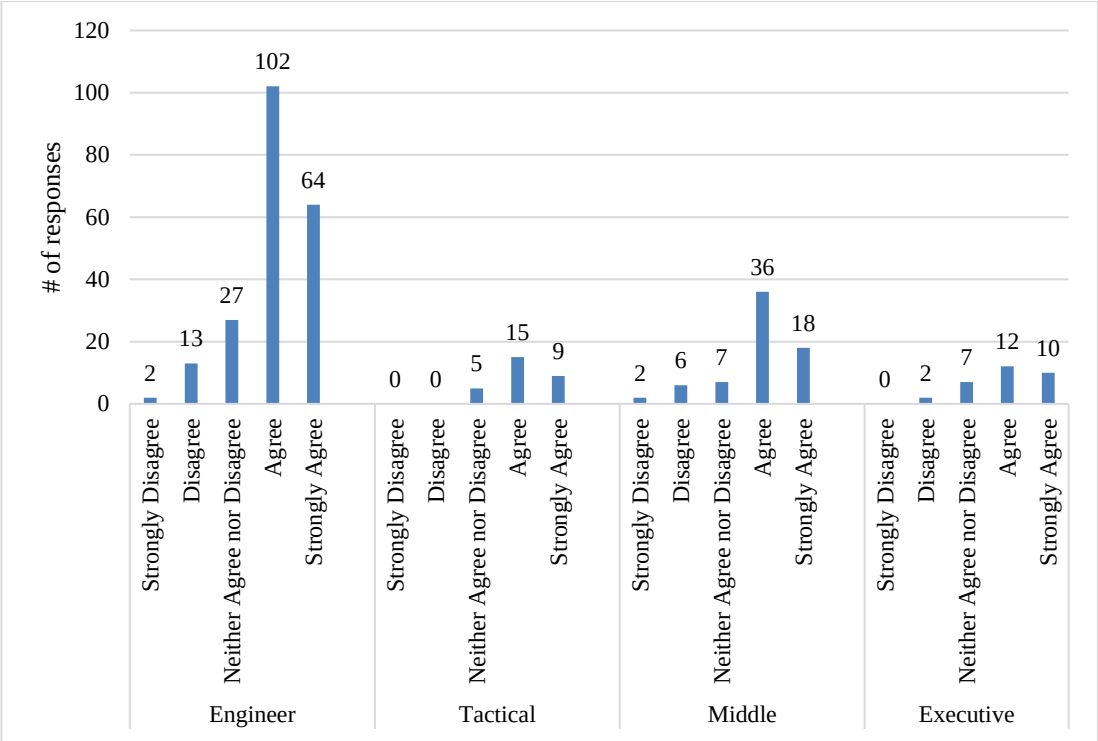

Figure 4.36: Organization-level wise analysis of the 'No security testing training' in the online survey distribution.

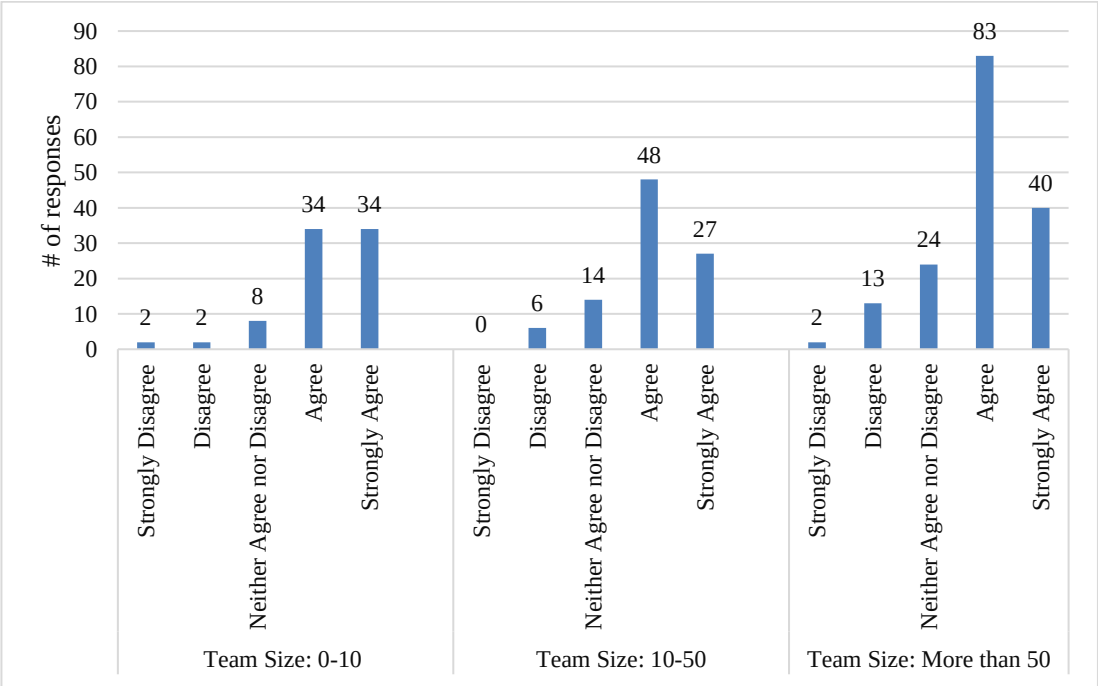

Figure 4.37: QA department wise analysis on the 'No security testing training' in the online survey distribution.

The suggestion made to overcome this problem in the online survey was, “Introduced more security testing meet-ups and training for SQA people.” Table 4.32 shows the distribution of the agreeableness towards the researcher’s suggestion to overcome the problem.

Table 4.32: Suggestion distribution made to overcome ‘No security testing training’ problem in the online survey.

Solution Description	**Agree**	**Neutral**	**Disagree**
Introduced more security testing meet-ups and training for SQA people.	93%	6%	1%

“Lack of time” is standing at the sixth rank on the significant problem list. 73% of the respondents agreed that ‘Lack of time’ is a problem for SQA professionals in software security testing. However, 9% of respondents think that it is not a problem for SQA professionals in software security testing. When considering the ‘Lack of time’ problem, this is not limiting to the SQA professionals. Time has become a problem due to the presence of unrealistic project deadlines. Based on the respondents’ thoughts, this is a problem across the company. Figure 4.38 shows the distribution of the agreeableness towards the problem. The suggestion made to overcome this problem was, “Form a dedicated QA security taskforce to develop and retain the security testing.” 91% of the respondents agreed to the suggestion made by the researcher. Figure 4.39 shows the gender-wise distribution of the agreeableness towards ‘Lack of time’ as a problem. Figure 4.40 shows the organization level-wise distribution of the agreeableness towards ‘Lack of time’ as a problem. Figure 4.41 shows the size of the QA department wise distribution of the agreeableness towards ‘Lack of time’ as a problem.

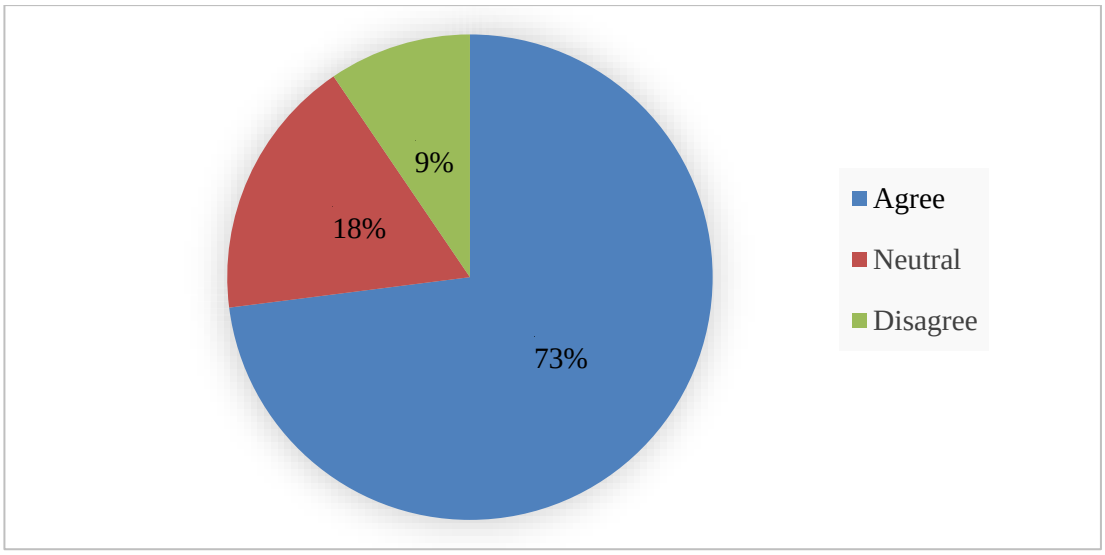

Figure 4.38: The agreeable extent of the participants for the 'Lack of time' as a problem.

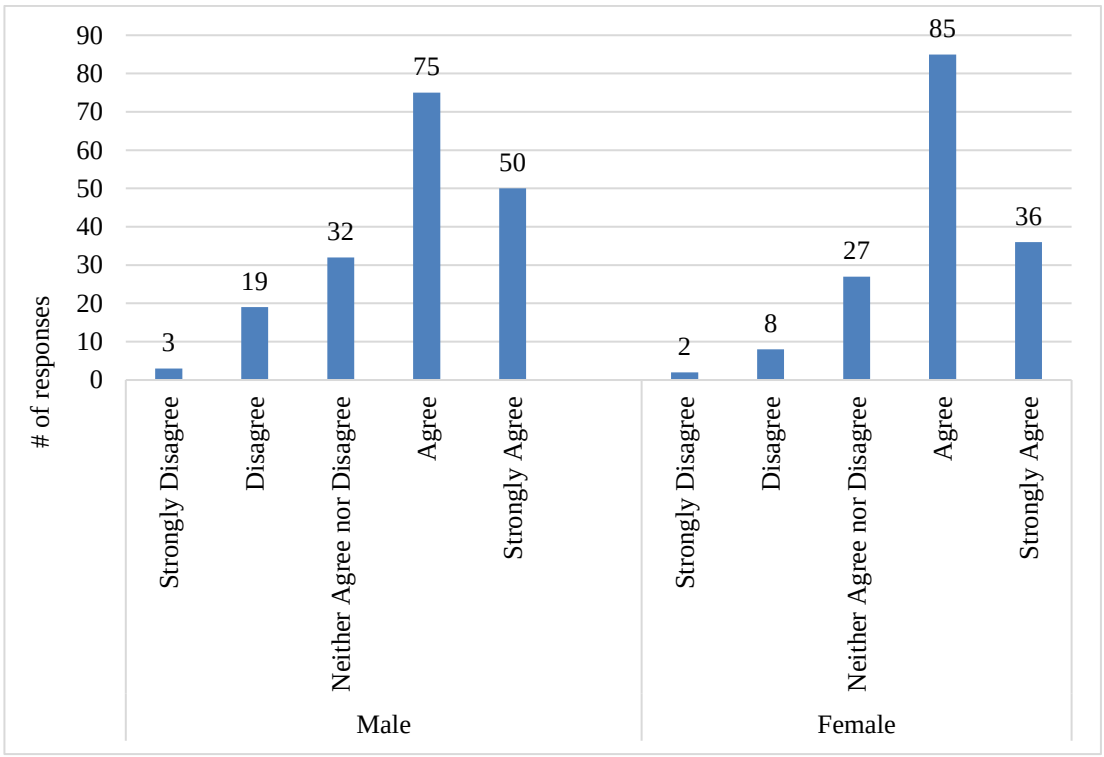

Figure 4.39: Gender-wise analysis of the 'Lack of time' in the online survey distribution.

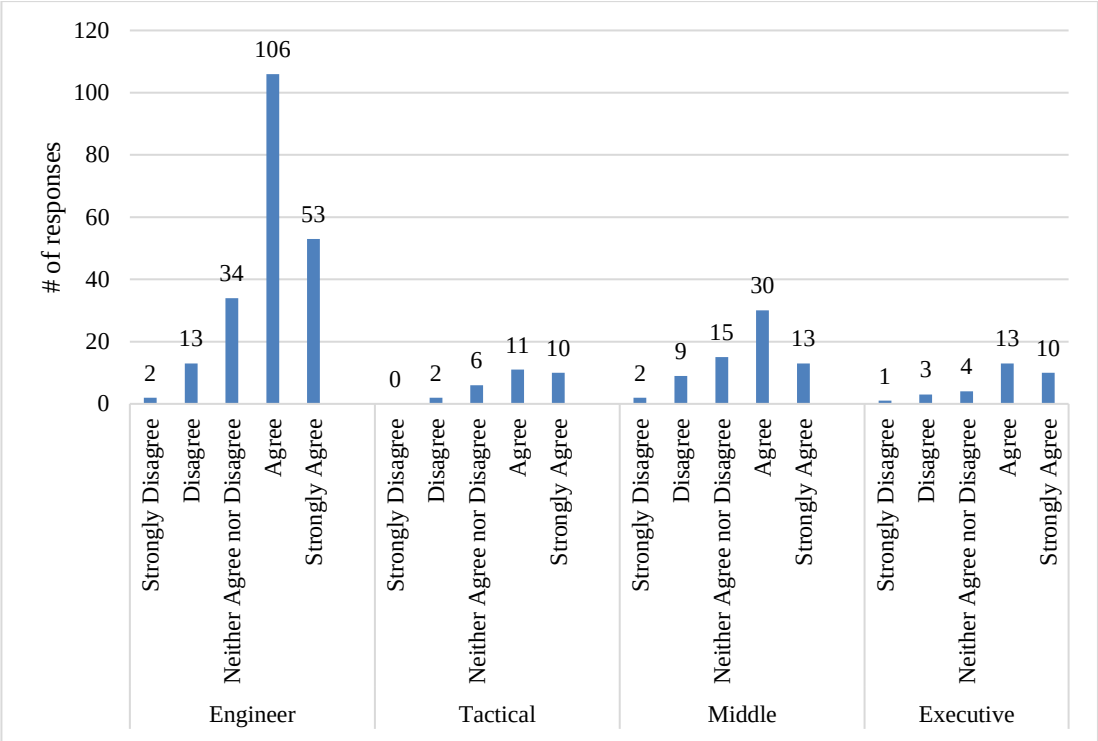

Figure 4.40: Organization level-wise analysis of the distribution of 'Lack of time' in the online survey.

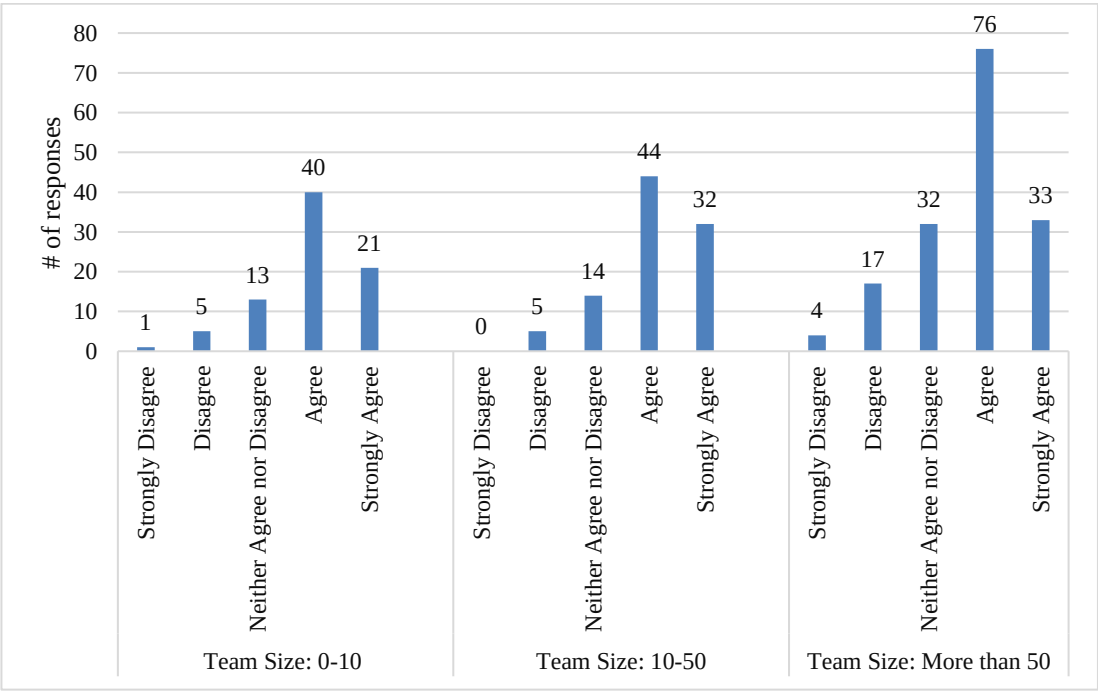

Figure 4.41: Size of the QA department wise analysis on the 'Lack of time' in the online survey distribution.

“Form a dedicated QA security taskforce to develop and retain security testing” was the researcher’s suggestion. Table 4.33 shows the distribution of the agreeableness towards the researcher’s suggestion to overcome the problem.

Table 4.33: Suggestion distribution made to overcome ‘Lack of time’ problem in the online survey.

Solution Description	**Agree**	**Neutral**	**Disagree**
Form a dedicated QA security taskforce to develop and retain security testing.	91%	8%	1%

The problem of “Complexity” was standing seventh place of the significant problem list. As per the respondent’s comments, Security testing can be quite tricky if there is no dramatic and well-functioning structure or process. Hence, 77% of respondents agreed that ‘Complexity’ is a problem for SQA professionals in software security testing. However, 9% of respondents think that ‘Complexity’ is not a problem for SQA professionals in software security testing. Figure 4.42 shows the distribution of the agreeableness towards ‘Complexity.’ 86% of the respondents agreed to the suggestion made by the researcher. Figure 4.43 shows the gender-wise distribution of the agreeableness towards ‘Complexity’ as a problem. Figure 4.44 shows the organization level-wise distribution of the agreeableness towards ‘Complexity’ as a problem. Figure 4.45 shows the size of the QA department wise distribution of the agreeableness towards ‘Complexity’ as a problem.

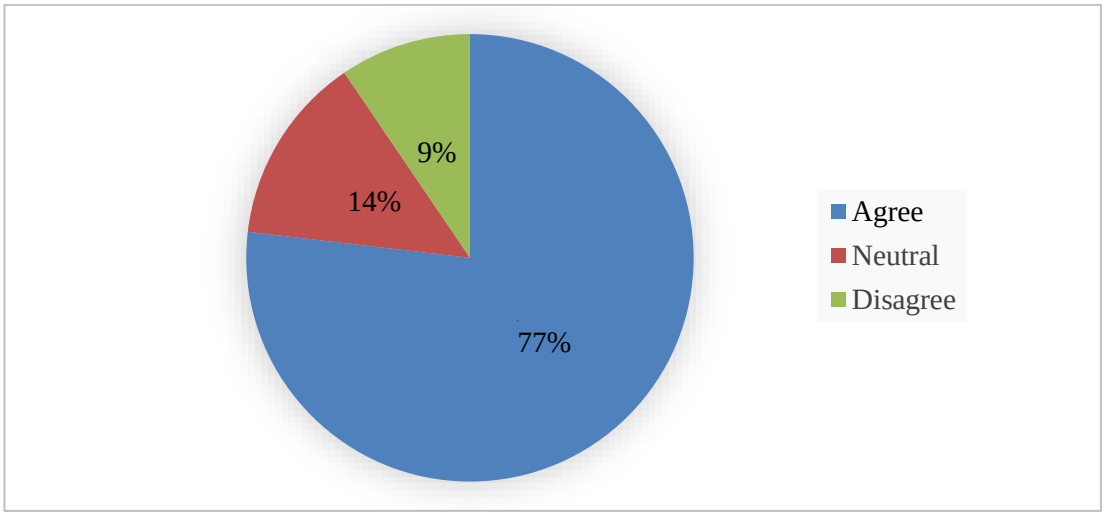

Figure 4.42: The agreeable extent of the participants for the ‘Complexity’ as a problem.

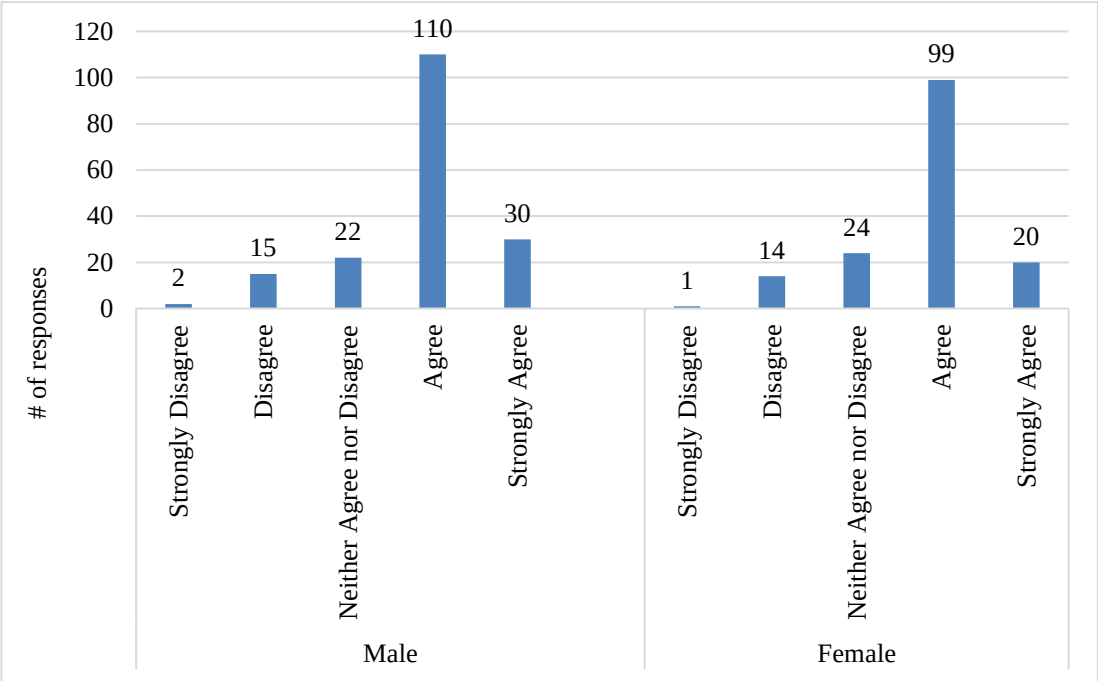

Figure 4.43: Gender wise analysis of the ‘Complexity’ in the online survey distribution.

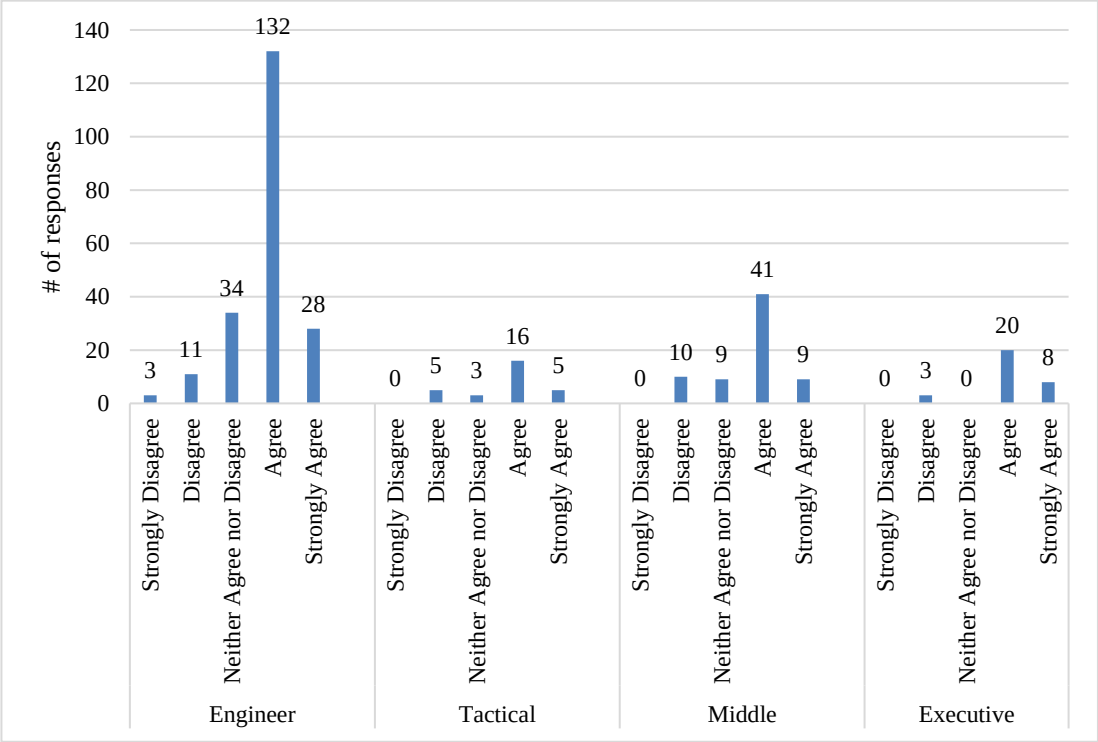

Figure 4.44 Organization level-wise analysis of the ‘Complexity’ in the online survey distribution.

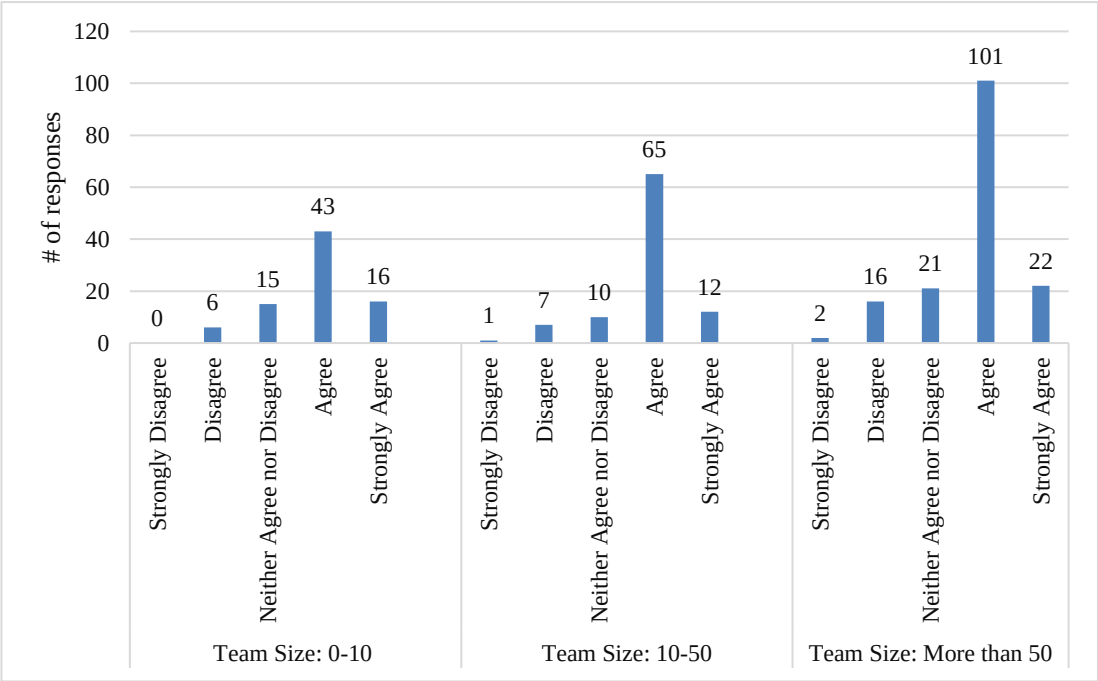

Figure 4.45 Size of the QA department wise analysis on the ‘Complexity’ in the online survey distribution.

“Reduce SQA individual’s lack of exposure to security testing by providing awareness” was the researcher’s suggestion. Table 4.34 shows the distribution of the agreeableness towards the researcher’s suggestion to overcome the problem.

Table 4.34: Suggestion distribution made to overcome the ‘Complexity’ problem in the online survey.

Solution Description	**Agree**	**Neutral**	**Disagree**
Reduce SQA individual’s lack of exposure to security testing by providing awareness.	86%	12%	2%

Another problem is "Less SQA involvement in system design, requirement gathering and code review phases." It is very critical and helps to destroy the concept ‘prevention is better than cure.’ Top management must be aware of where to start QA. The problem has ranked eighth in the significant problem list. 70% of respondents agreed with this as a problem. Moreover, 11% of respondents consider that 'Less SQA involvement in system design, requirement gathering and code review phases' is not a problem for the SQA professionals in software security testing.

Figure 4.46 shows the distribution of the agreeableness towards Less SQA involvement in system design, requirement gathering, and code review phases' as a problem. Figure 4.47 shows the gender-wise distribution of the agreeableness towards 'Less SQA involvement in system design, requirement gathering, and code review phases' as a problem. Figure 4.48 shows the organization level-wise distribution of the agreeableness towards 'Less SQA involvement in system design, requirement gathering, and code review phases' as a problem. Figure 4.49 shows the size of the QA department wise distribution of the agreeableness towards 'Less SQA involvement in system design, requirement gathering, and code review phases' as a problem.

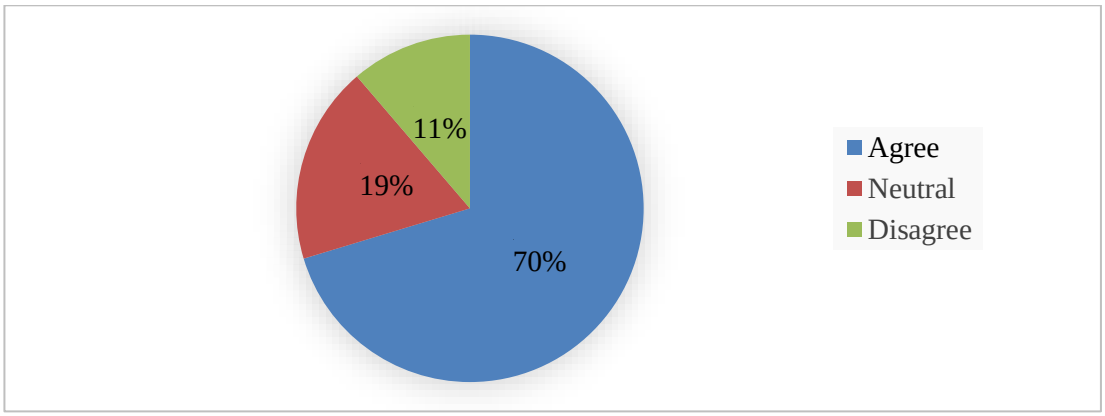

Figure 4.46: The agreeable extent of the participants for the 'Less SQA involvement in system design, requirement gathering, and code review phases' as a problem.

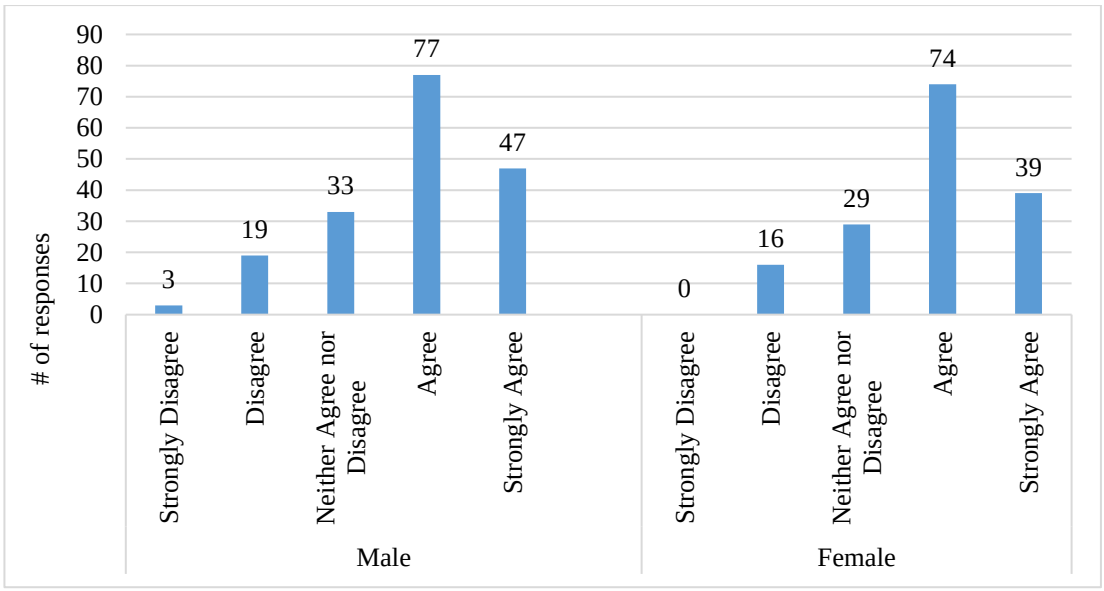

Figure 4.47: Gender wise analysis of the 'Less SQA involvement in system design, requirement gathering, and code review phases' in the online survey distribution.

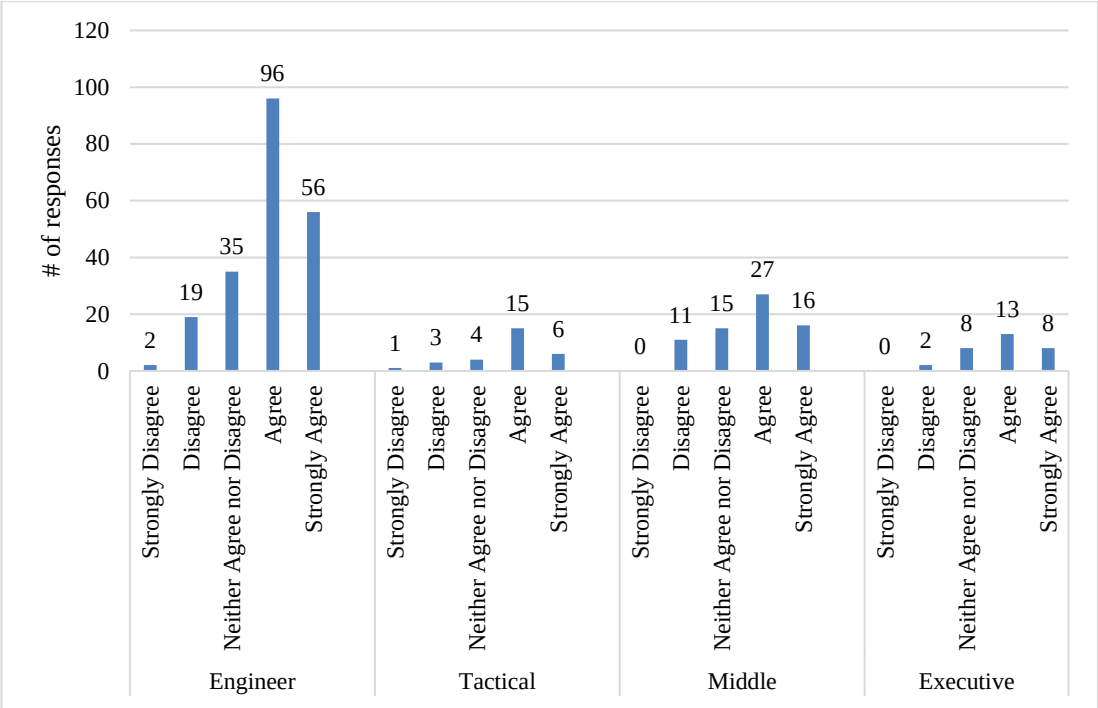

Figure 4.48: Organization level-wise analysis on the 'Less SQA involvement in system design, requirement gathering, and code review phases' in the online survey distribution.

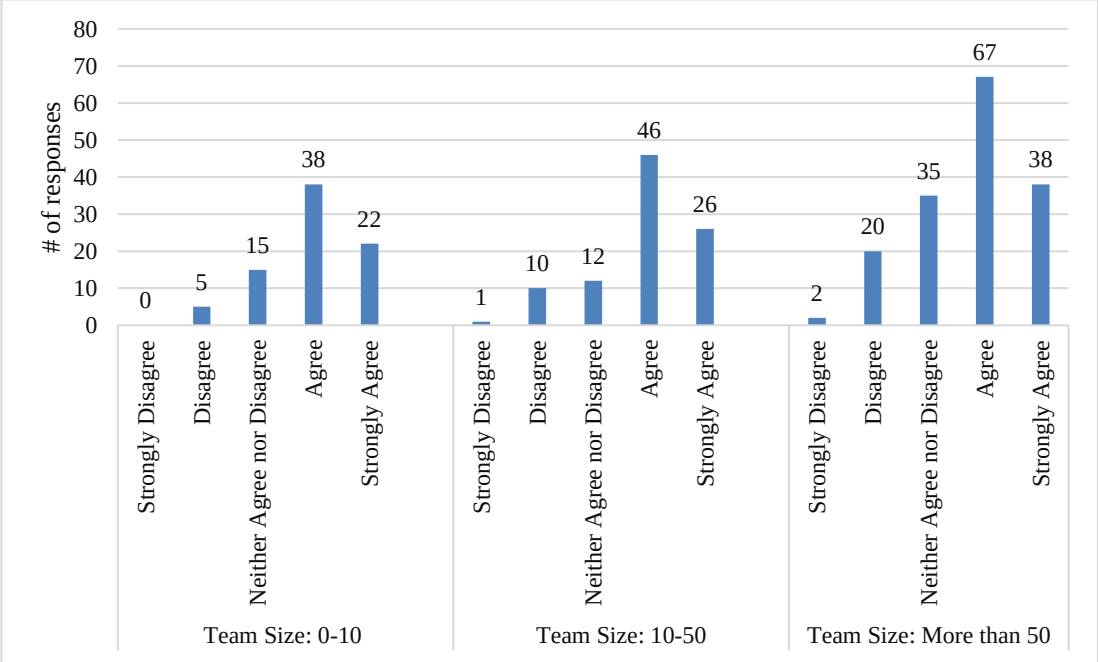

Figure 4.49: Size of the QA department wise analysis on the 'Less SQA involvement in system design, requirement gathering, and code review phases' in the online survey distribution.

“Facilitate SQA participation in non-QA related phases of the development life cycle” was the researcher’s suggestion.Table 4.35 shows the distribution of the agreeableness towards the researcher’s suggestion to overcome the problem.

Table 4.35: Suggestion distribution made to overcome ‘Less SQA involvement in system design, requirement gathering, and code review phases’ problem in the online survey.

Solution Description	**Agree**	**Neutral**	**Disagree**
Facilitate SQA participation in non-QA related phases of the development life cycle.	91%	8%	1%

The next problem is “Lack of management support.” As per the respondent’s comments, some managers do not want to accept QA, which adds significant value to the organization. Most of them consider QA as a bottleneck for client deliverables. The problem has ranked ninth due to this situation. 65% of respondents agreed that ‘Lack of management support’ is a problem for SQA professionals in software security testing. However, 13% of respondents think that ‘Lack of management support’ is not a problem for SQA professionals in software security testing. Figure 4.50 shows the distribution of the agreeableness towards ‘Lack of management support.’ 87% of the respondents agreed to “Provide strong management support’ as a suggestion, and 91% agreed to “Keep the higher management informed by having weekly, monthly progress review or awareness meetings” as a suggestion made by the researcher. Figure 4.51 shows the gender-wise distribution of the agreeableness towards ‘Lack of management support’ as a problem. Figure 4.52 shows the organization level-wise distribution of the agreeableness towards ‘Lack of management support’ as a problem. Figure 4.53 shows the size of the QA department wise distribution of the agreeableness towards ‘Lack of management support’ as a problem.

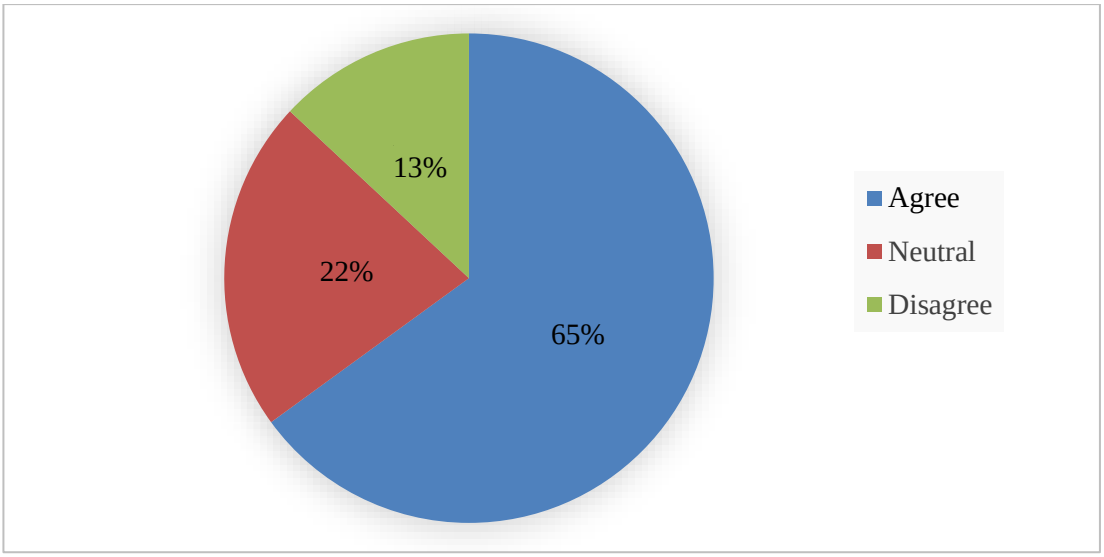

Figure 4.50: The agreeable extent of the participants for the 'Lack of management support' as a problem.

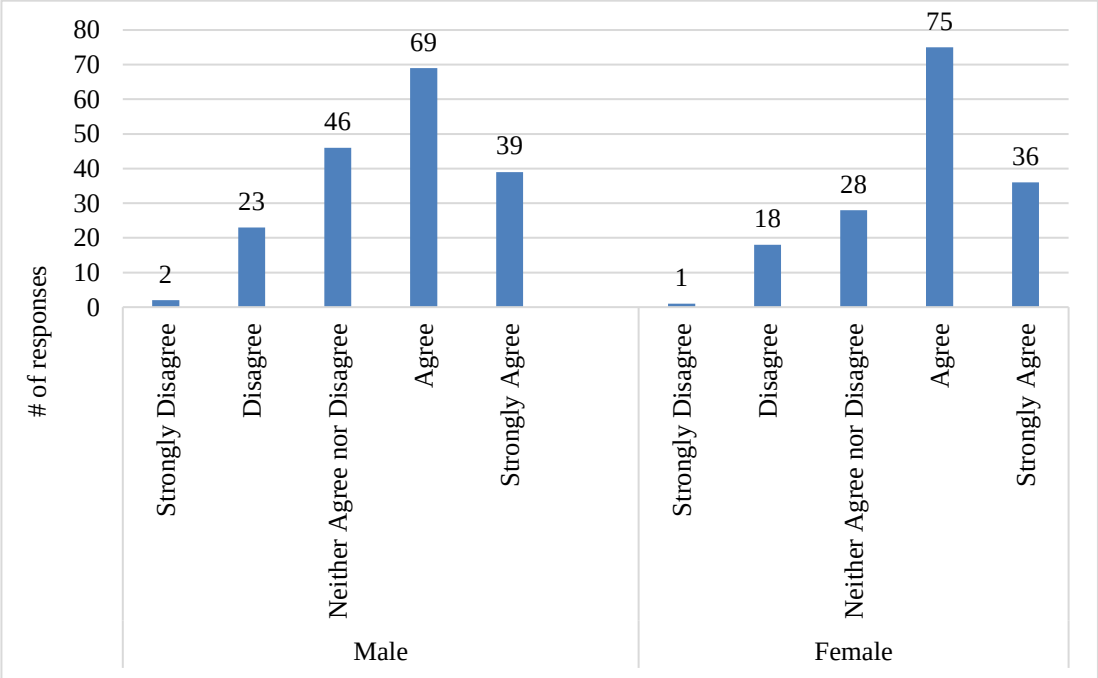

Figure 4.51: Gender-wise analysis of the 'Lack of management support' in the online survey distribution.

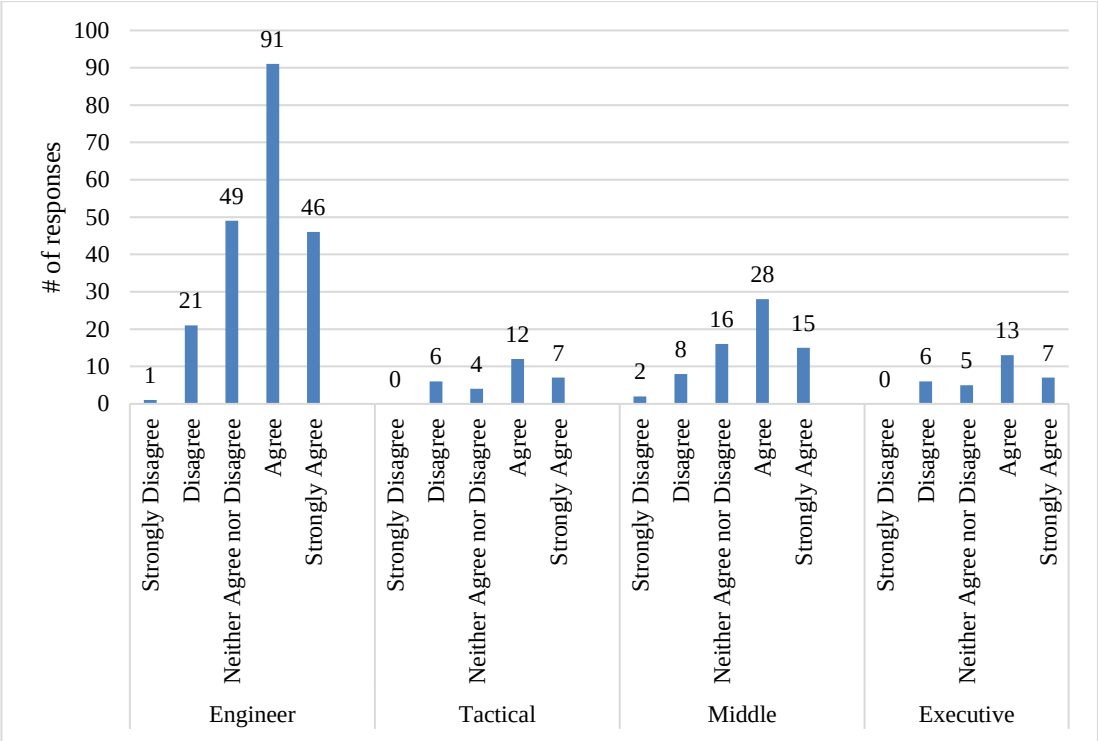

Figure 4.52: Organization level-wise analysis of the 'Lack of management support' in the online survey distribution.

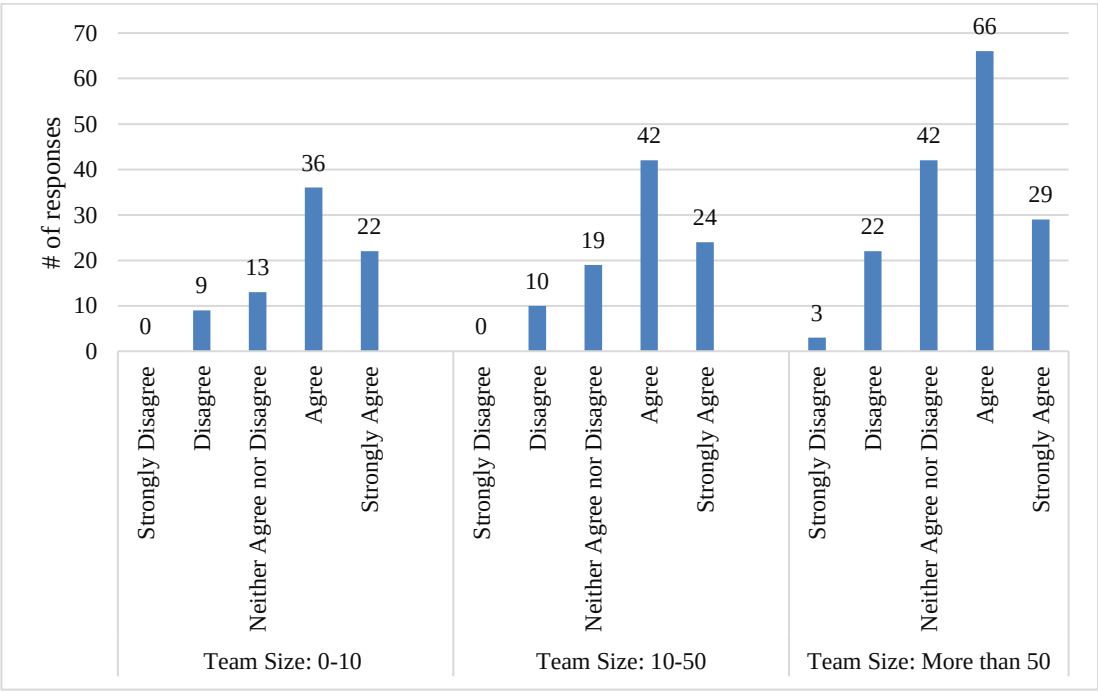

Figure 4.53: Size of the QA department wise analysis on the 'Lack of management support' in the online survey distribution.

“Provide strong management support” and “Keep the higher management informed by having weekly, monthly progress review or awareness meetings” were the researcher’s suggestions. Table 4.36 shows the distribution of the agreeableness towards the researcher’s suggestions to overcome the problem.

Table 4.36: Suggestions distribution made to overcome ‘Lack of management support’ problem in the online survey.

Solution Description	**Agree**	**Neutral**	**Disagree**
Provide strong management support.	87%	10%	3%
Keep the higher management informed by having weekly, monthly progress reviews or awareness meetings.	91%	8%	1%

The problem of “No project requirements” was standing tenth place of the significant problem list. As per the respondent’s comments, it is hard to work on tests that are not included in project requirements since there are no time allocations for those. Hence, 60% of respondents agreed that ‘No project requirements’ is a problem for SQA professionals in software security testing. However, 13% of respondents think that ‘No project requirements’ is not a problem for SQA professionals in software security testing. Figure 4.54 shows the distribution of the agreeableness towards ‘No project requirements.’ 90% of the respondents agreed to the suggestion made by the researcher. Figure 4.55 shows the gender-wise distribution of the agreeableness towards ‘No project requirements’ as a problem. Figure 4.56 shows the organization level-wise distribution of the agreeableness towards ‘No project requirements’ as a problem. Figure 4.57 shows the size of the QA department wise distribution of the agreeableness towards ‘No project requirements’ as a problem.

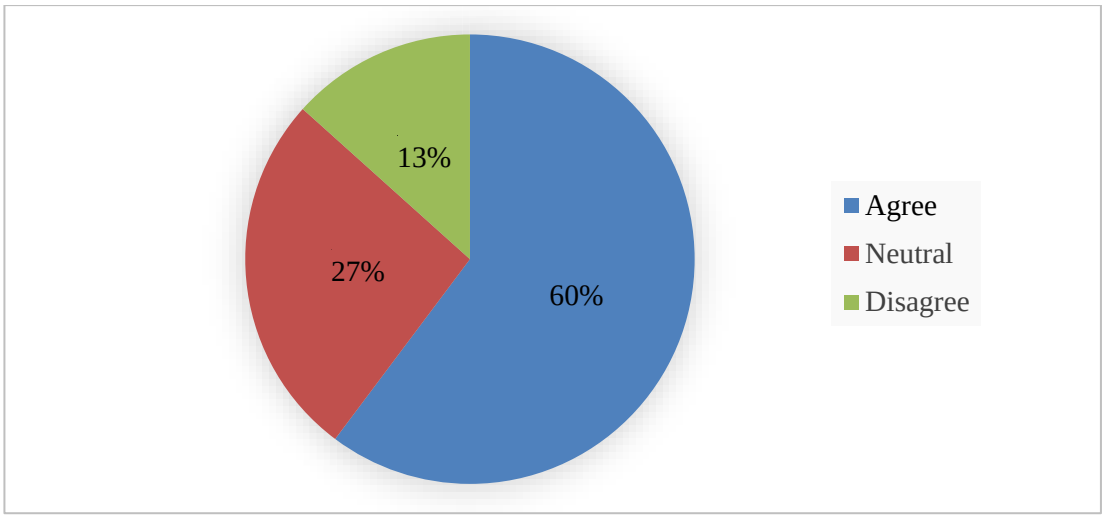

Figure 4.54: The agreeable extent of the participants for the ‘No project requirements’ as a problem.

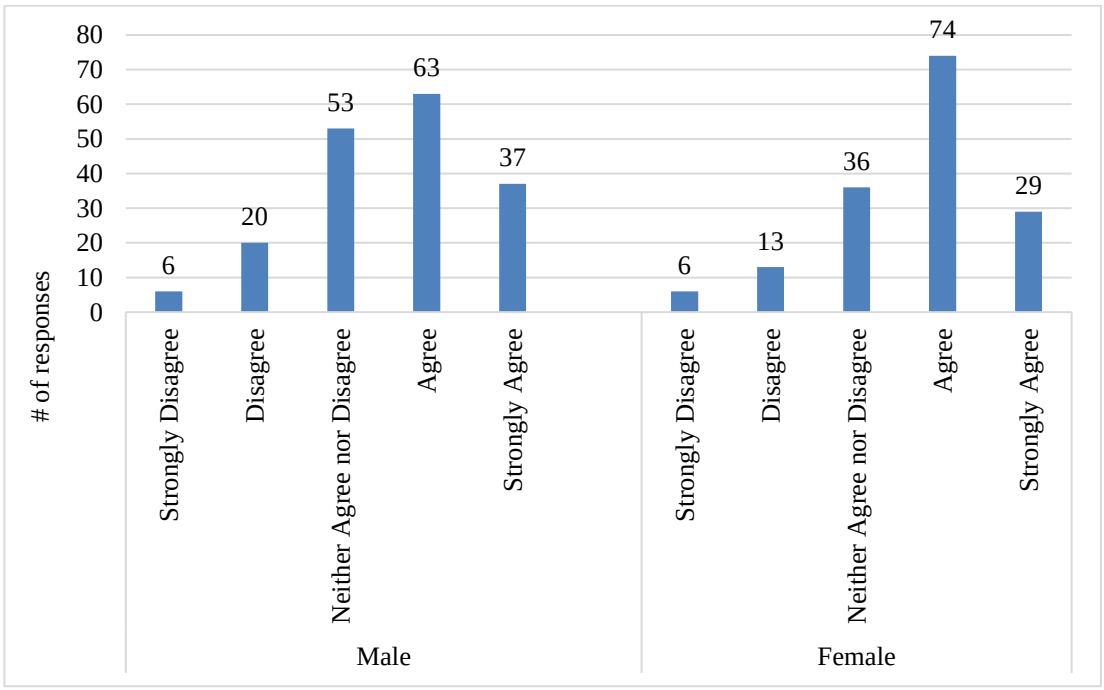

Figure 4.55: Gender-wise analysis of the ‘No project requirements’ in the online survey distribution.

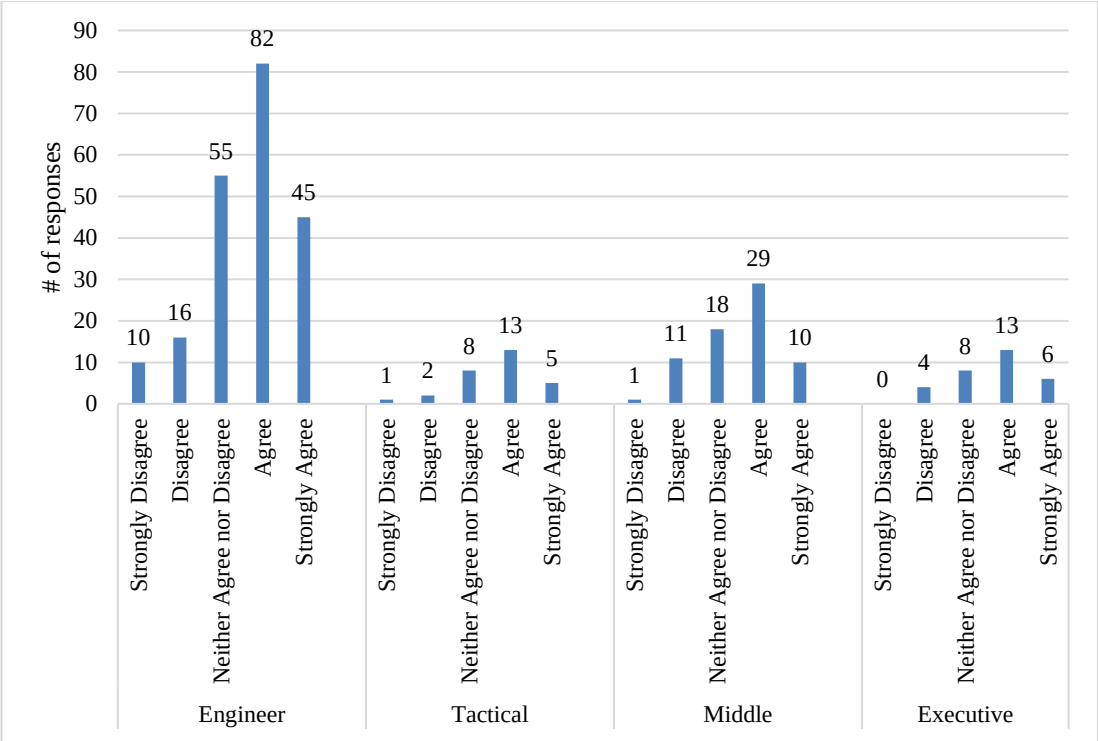

Figure 4.56: Organization level-wise analysis of the 'No project requirements' in the online survey distribution.

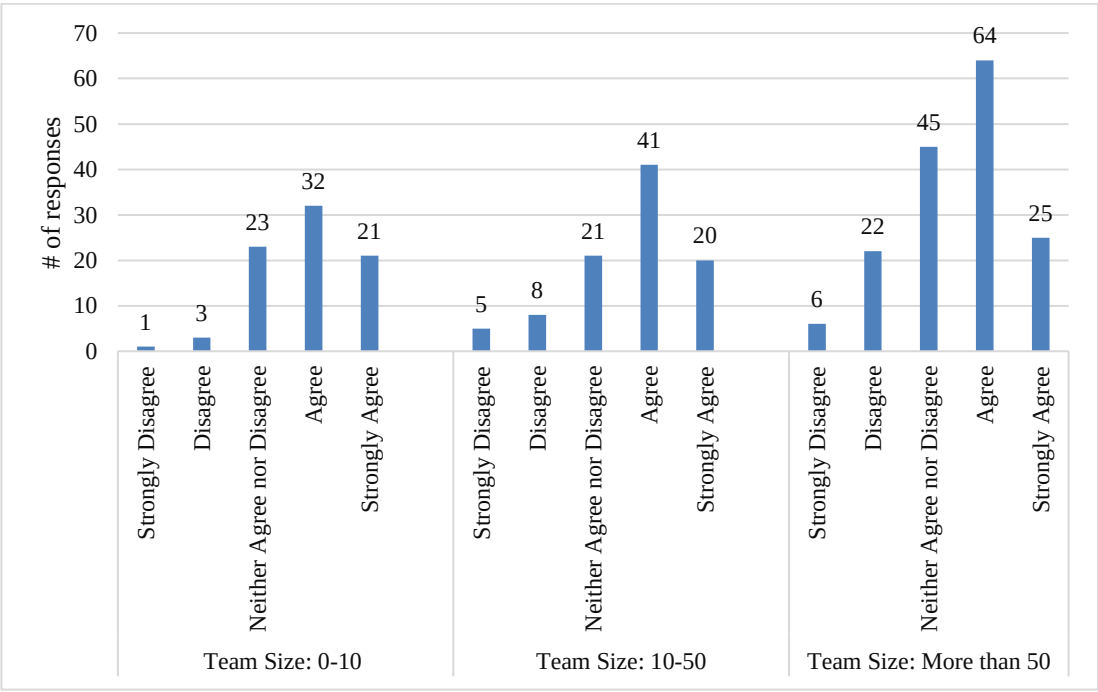

Figure 4.57: Size of the QA department wise analysis on the 'No project requirements' in the online survey distribution.

“Have SQA pool of people to service projects which are having security testing requirements” was the researcher’s suggestion. Table 4.37 shows the distribution of the agreeableness towards the researcher’s suggestion to overcome the problem.

Table 4.37: Suggestion distribution made to overcome ‘No project requirements’ problem in the online survey.

Solution Description	**Agree**	**Neutral**	**Disagree**
Have an SQA pool of people to service projects which are having security testing requirements.	90%	9%	1%

The problem of “Lack of motivation” was standing eleventh place of the significant problem list. As per the respondent’s comments, people who work with technical experts all the time are usually quite introverted if they do not hear a conversation that attracts attention. Hence, 62% of respondents agreed that ‘Lack of motivation’ is a problem for SQA professionals in software security testing. However, 15% of respondents think that ‘Lack of motivation’ is not a problem for SQA professionals in software security testing. Figure 4.58 shows the distribution of the agreeableness towards ‘Lack of motivation.’ 88% of the respondents agreed to the suggestion made by the researcher. Figure 4.59 shows the gender-wise distribution of the agreeableness towards ‘Lack of motivation’ as a problem. Figure 4.60 shows the organization level-wise distribution of the agreeableness towards ‘Lack of motivation’ as a problem. Figure 4.61 shows the size of the QA department wise distribution of the agreeableness towards ‘Lack of motivation’ as a problem.

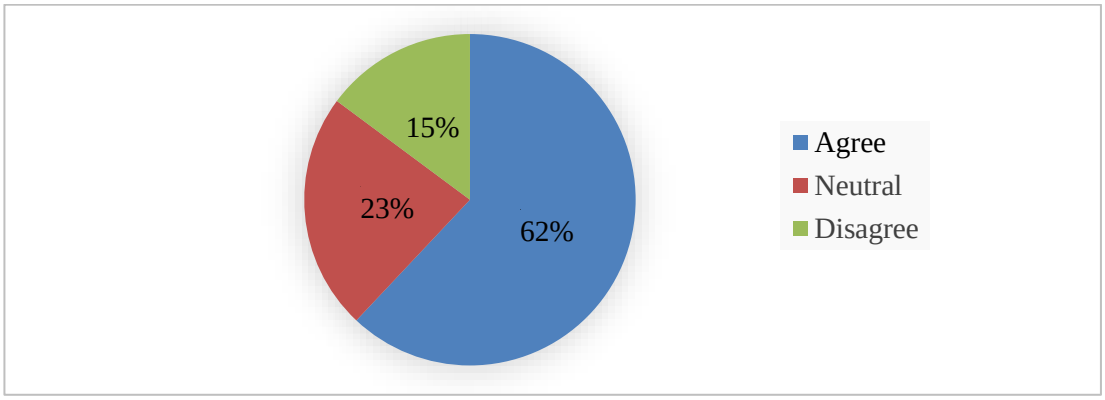

Figure 4.58: The agreeable extent of the participants for the ‘Lack of motivation t’ as a problem.

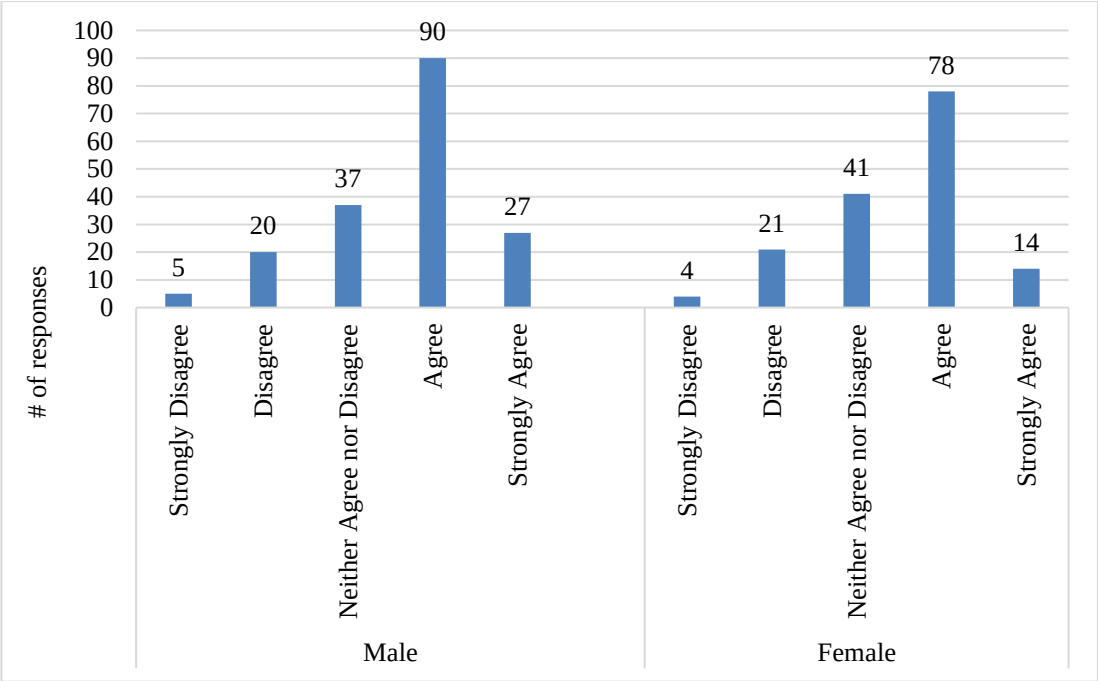

Figure 4.59: Gender-wise analysis of the 'Lack of motivation' in the online survey distribution.

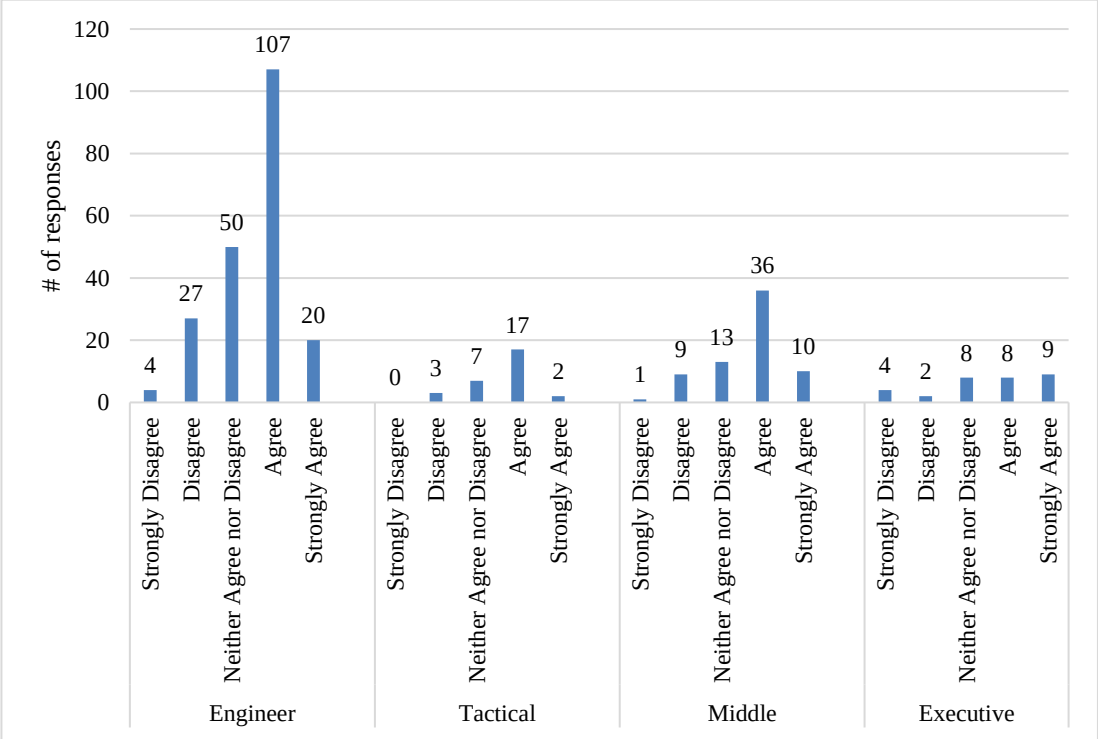

Figure 4.60: Organization level-wise analysis of the 'Lack of motivation' in the online survey distribution.

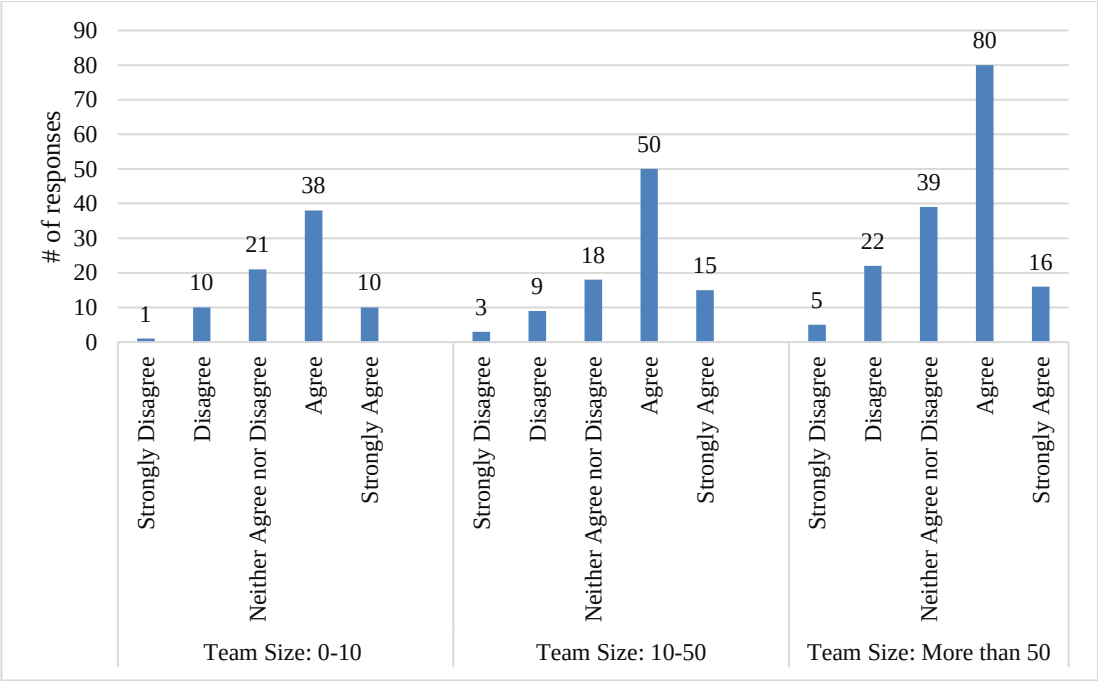

Figure 4.61: Size of the QA department wise analysis on the 'Lack of motivation' in the online survey distribution.

"Motivate SQA people to do security testing sessions during project idle times" was the researcher's suggestion. Table 4.38 shows the distribution of the agreeableness towards the researcher's suggestion to overcome the problem.

Table 4.38: Suggestion distribution made to overcome 'Lack of motivation' problem in the online survey.

Solution Description	**Agree**	**Neutral**	**Disagree**
Motivate SQA people to do security testing sessions during project idle times.	88%	11%	1%

The problem of "Lower salary scale compared to other IT professions" was standing twelfth place of the significant problem list. As per the respondent's comments, this is a debatable point when discussing the different levels of SQA professionals and managers. Hence, 43% of respondents agreed 'Lower salary scale compared to other IT professions' is a problem for SQA professionals in software security testing. However, 26% of respondents think that 'Lower salary scale compared to other IT professions' is not a problem for SQA professionals in software security testing. Figure 4.62 shows the distribution of the agreeableness towards 'Lower salary scale compared to other IT professions.' 85% of the respondents agreed to the suggestion made by the researcher. Figure 4.63 shows the gender-wise distribution of the agreeableness towards 'Lower salary scale compared to other IT professions' as a problem. Figure 4.64 shows the organization level-wise distribution of the agreeableness towards 'Lower salary scale compared to other IT professions' as a problem. Figure 4.65 shows the size of the QA department wise distribution of the agreeableness towards 'Lower salary scale compared to other IT professions' as a problem.

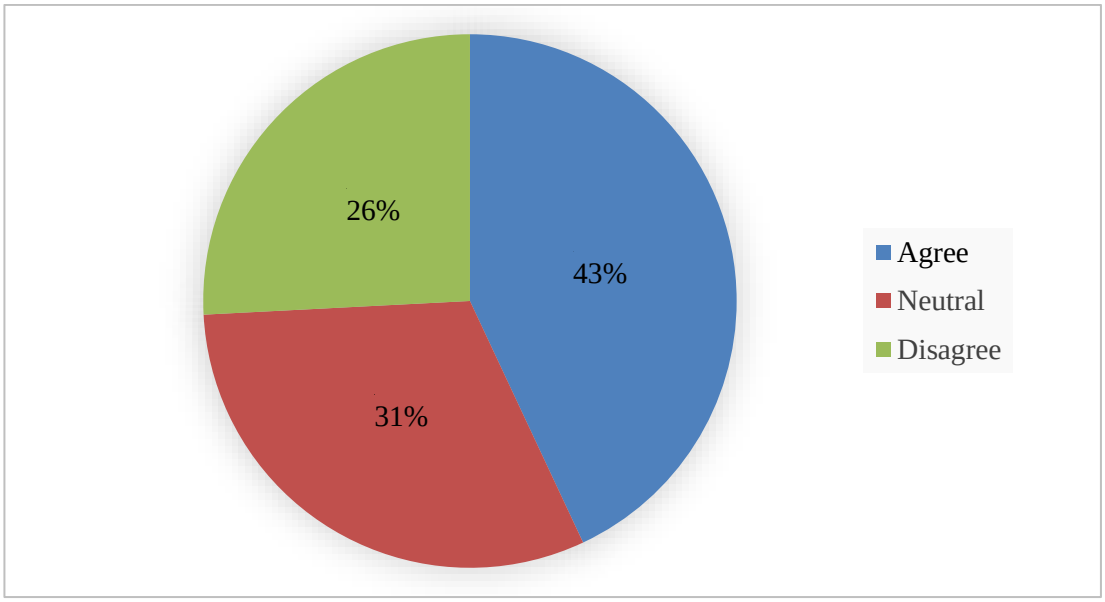

Figure 4.62: The agreeable extent of the 'Lower salary scale compared to other IT professions' as a problem.

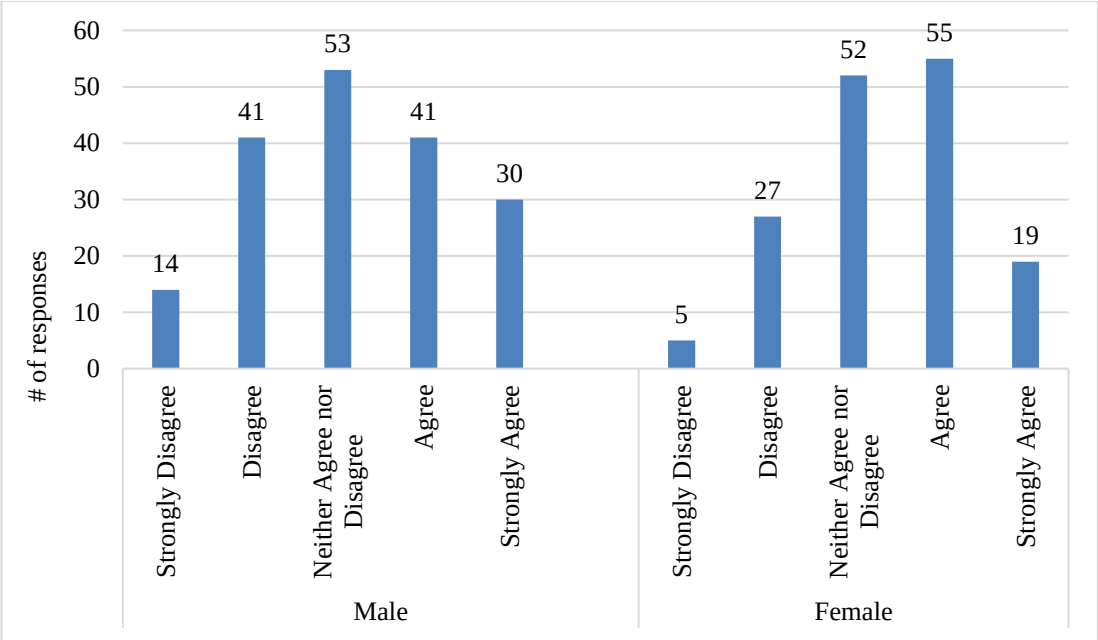

Figure 4.63: Gender-wise analysis of the 'Lower salary scale compared to other IT professions' in the online survey distribution.

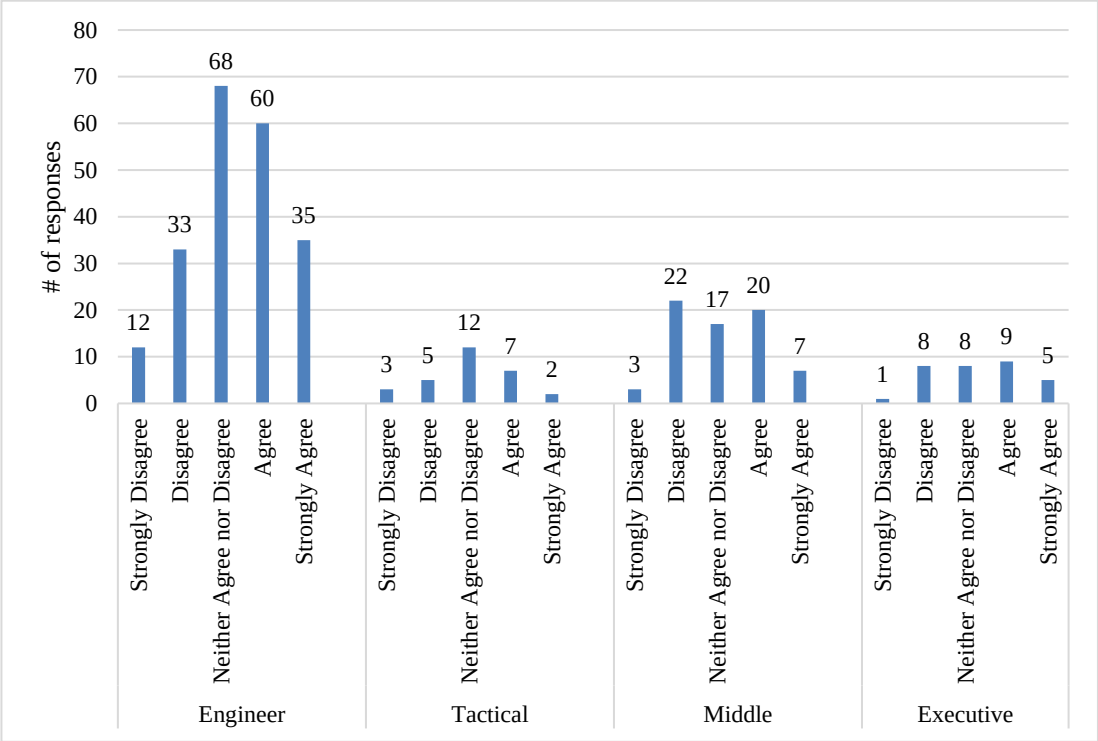

Figure 4.64: Organization level-wise analysis of the 'Lower salary scale compared to other IT professions' in the online survey distribution.

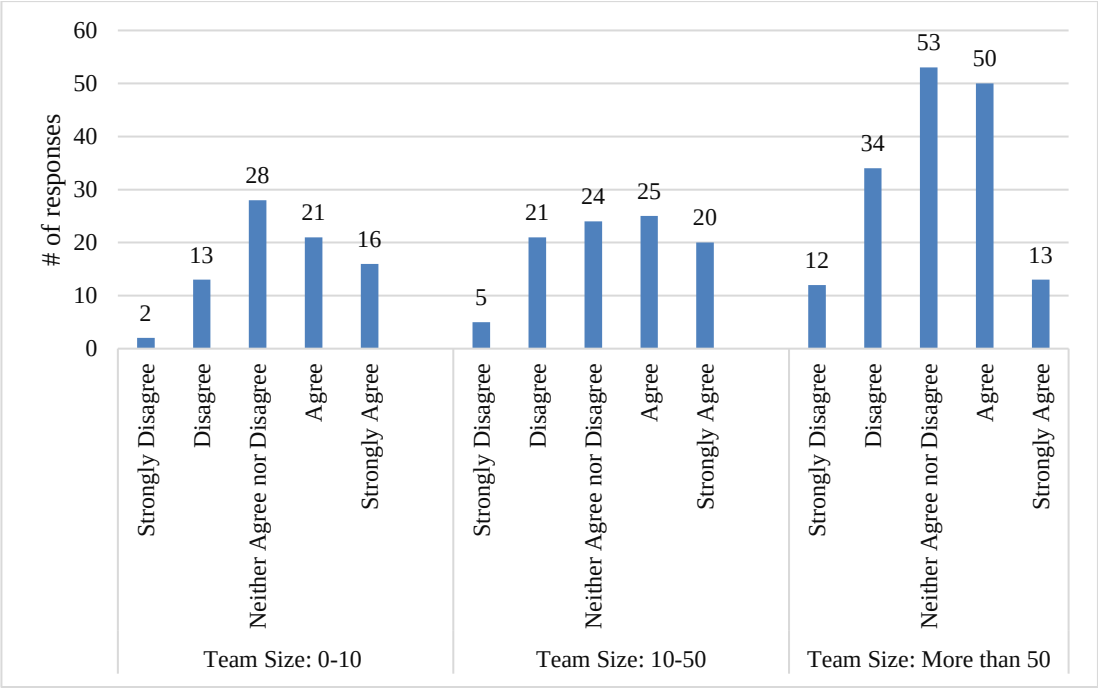

Figure 4.65: Size of the QA department wise analysis on the 'Lower salary scale compared to other IT professions' in the online survey distribution.

"Increased standard of living for the skilled SQA resources" was the researcher's suggestion. Table 4.39 shows the distribution of the agreeableness towards the researcher's suggestion to overcome the problem.

Table 4.39: Suggestion distribution made to overcome 'Lower salary scale compared to other IT professions' problem in the online survey.

Solution Description	**Agree**	**Neutral**	**Disagree**
Increased standard of living for the skilled SQA resources.	85%	12%	3%

4.3. Interview Results

A series of interviews were conducted with SQA senior management/experts to find suggestions to overcome the identified problems. It was essential to find out how management sees the problems identified since 62% of the respondents were from an engineering level. Besides, their suggestions are valuable because this study is related to the level of management, and only they can implement these suggestions in the organization. This section describes the agreeableness of management for significant problems and their suggestions for overcoming them.

All SQA experts faced for the interview agreed that "Lack of specialized SQA people in security testing" is a crucial problem to SQA professionals in software security testing. The following are the practical suggestions they mentioned to overcome the problem (Please refer to 'APPENDIX D' for more details).

- All about how management builds competencies within the organization: many initiatives need to do in these areas. Identify areas for the year (automation, productivity, security, mobility, and cloud testing) and organize internal training and invite external trainers to conduct training. Should not force people to become specialized QA. Individuals also need to spend some time to achieve this.
- Inside the workplace, training should do. Both technical and subject arrange dedicated QA's for security testing. So that they can develop and maintain knowledge. Collaborate support from developers and implementation engineers.
- Individuals should see the current threat to the industry right now. They need to know their level of competency to meet demand.
- Form a QA security taskforce. This group will have one main goal: to develop and maintain a security testing mindset among QA professionals. Members of the task force will be able to continually learn new tools and methods for security testing, apply their security testing abilities, and share their experience with other members of the group. Training people to perform POC to a certain extent and allows them to perform actions in real-time.

- The introduction of more QA-related modules at the degree level, such as security testing, and providing them with information about QA's involvement in security testing will help to some extent.
- The lack of qualified/specialized QA resources is a real problem faced by countries such as Sri Lanka. It would be a more laborious task if it had to address at the company level. Instead, this should address at the industry level. There should be initiatives to identify 3 or 4 core specialized skills. The high school curriculum at universities must be modified to cater for this. These courses should include more practical aspects and practical exercises in comparison with the theoretical part. Students should be given assignments for developing automation packages or security testing scripts. When these new graduates join the company, this hands-on knowledge will allow them to start working directly in security tests, automation projects, or productivity projects (initially with older resources). It will reduce the burden on companies that will spend more time and money training new graduates from scratch.
- Develop specialized SQA specialists in the field of security testing, providing the necessary training in the workplace, and they should be conducted both from a technical point of view and from the subject area.

Furthermore, experts agreed to the below suggestions made by the researcher in the online survey: -

- Form a dedicated QA security taskforce to develop and retain the security testing mindset among SQA people.
- Recruit detail-oriented and experienced SQA professionals.

Most of the SQA experts faced for the interview agreed that "Budget" is a crucial problem for SQA professionals in software security testing. The following are the practical suggestions they mentioned to overcome the problem (Please refer to 'APPENDIX D' for more details).

- When offering estimates, all security risk factors need to be identified, highlighted, quantified, and communicated to the client. For a particular module, this is the number of hours that we must spend, will determine the number of changes for which we need so much time. For a change, we determined the time. If it exceeds, contact the customer and tell us the extra time that we need. We place CR and get the time and budget approved.

- Determine the number of security tests for each module. Put them in 3 buckets (for example, 3-4 hours, 7-8 hours, 12 hours). Depending on how long it took. Conduct a series of Q&A sessions, and make assumptions on uncleared points. The number of defects is predicted depending on the competency level of the development team. For additional hours, hold a meeting with customers and get an approved budget for things we cannot control.

- Reveals the expected quality level after allocating this much amount of time and budget. When developers spend time testing, weiget together and develop a joint plan. During development, we do QA, invest in automation, add some additional resources to overcome.

- When we get the number of hours for quality assurance, check the available resources / get the expected quality from the client/area in which QA (the service can execute, align resources) also shows the client, if you give me this extra time, I can include these things, this is a risk that I can reduce, this is the level of quality that I can achieve.

- There is a way to do a good job, and there are always ways to do a better job. To do the best job (in this case, increase productivity and effectiveness in QA), various resources are required, such as human resources, tools, environment. It means that there is more cost to the project or company. Management must convince that QA is a vital component of the project, and it is essential to

identify security weaknesses. Encourage customers to reduce resources over time with innovative and creative approaches.

- People do not understand the value of QA. So, it is better to justify ROI. Talk to people who set aside a budget on more dollar terms. Justify QA resource requirements in the ROI equation (return on investment QA).
- Again, the problem is a lack of understanding of "what is required" concerning QA / QC. According to the expert experience, the planning stages of a project do not adequately evaluate these resources. At times, the quality assurance team must blame, as they cannot come to concrete plans to convince the PM of the risks of not adhering to proper quality control following a specific project. The solution is to have better project planning, which includes QA / QC planning.
- Make sure that the developer appreciates the participation of QA's to ensure the security of the project. They also need to demand quality. It is about productivity (how fast you can ship to market).
- Consider people to give the right estimates for project planning. When evaluating, use a re-prospective. It helps QA to evaluate themselves and determine the right speed and effectiveness. They can also define the same criter aifor a eam. Execute expert judgment for assessment — prioritization test scripts.
- Use breath first approach/methodology. It can run through important functions / basic flow (smoke test) to quickly identify critical/important security problems. Share knowledge with developers in the early stages to get quality releases in the early stages, expanding their capabilities. The problems will be fixed before QA receive the release. Prevention is always better than cure. Plan a poker approach.

Furthermore, experts agreed to the below suggestion made by the researcher in the online survey: -

- Provide proper SQA resources regardless of the profit margin.

Again, most of the SQA experts faced for the interview agreed that "Lack of knowledge about security testing fundamentals" is a crucial problem to SQA professionals in software security testing. The following are the practical suggestions they mentioned to overcome the problem (Please refer to ‘APPENDIX D’ for more details).

- It is essential to know the application QA is testing to assess the risks. Everything else will assume that QA possesses this knowledge - the technologies used by the application, the profile of different users, the capabilities that they should and should not have at different access levels, and the potential data that stored in the application. When it comes to understanding the security terms and definitions, OWASP is a great source. At first, the volume of terms and concepts can be overwhelming, so focus on understanding some terms, preferably those that are most likely to apply to the application. Examples are XSS, XSRF, SQL injection, and path traversal. CWE / SANS Top 25 lists the most common and critical errors that cause vulnerabilities.

- An excellent way to start learning is to start testing a software/application with known vulnerabilities, where QA can find instructions on how to detect them. Expert prefers Google Gruyere, which has separate lessons to cover every concept. QA can look at the tips which guide to find the vulnerability, and answers if necessary. Some other options are OWASP WebGoat and Damn Vulnerable Web App.

- It is likely that between the developers in a company, some know security topics. Ask them to team up with QA to study the behavior of the application. They should be able to show, for example, that the SQL injection string is not running on the database, and why it is not. If so, then it will be beneficial for QA. They can also teach QA people about the design of the application and how it designed to protect against attacks. If many people want to know about security, ask them to make a presentation.

- An excellent commercial option is the Burp Scanner; There are also free options like OWASP ZAP and Google RatProxy. They work by routing HTTP traffic to and from the application through a proxy server and then resending requests with various attack attempts that replace the original values. It may be an effective way to detect specific classes of vulnerability in a short period, but it is essential to understand (and make sure your stakeholders understand) that it is not a magic wand. The tool is naive and does not know the business logic of applications - it only reproduces requests and checks the answers.
- There are many types of vulnerabilities that cannot and will not be detected using tools, and the use of a scanning tool does not at all replace the need for manual security testing. Automated tools, even expensive ones, only detect relatively simple vulnerabilities, and they usually produce a lot of "noise" or false positives. QA should know enough about security vulnerabilities to be able to evaluate each discovery of an automated tool. Taking a scanner report and analyze it and learn from it is the best thing QA can do to understand security test fundamentals.

Furthermore, experts agreed to the below suggestions made by the researcher in the online survey: -

- Familiarize and adapt security testing fundamentals, protocols, tools, and methods to fit within existing processes.
- Maintain a security testing knowledge portal.

All SQA experts faced for the interview highly agreed that "Lack of detailed information and advice" is a crucial problem to SQA professionals in software security testing. The following are the practical suggestions they mentioned to overcome the problem (Please refer to 'APPENDIX D' for more details).

- Like any skill, QA will get higher with practice. Once they begin to find vulnerabilities in the application, they will begin to understand where those can be in the future, and they can raise them in advance.

- More focused training would help, such as various course providers such as SANS. There are security training courses specifically for those responsible for quality assurance, so look for security testing courses. Penetration testing courses tend to focus on hacking the network, but they often have parts on hacking web applications, so check the course content in advance.

- When testing a feature, QA's are likely to create test data. Instead of using "test1", "test2", or the names of cartoon characters, get into the habit of using attack lines. Thus, QA will find that they discover vulnerabilities almost by accident, only by using the function. If they have an automated tool or an import file that provides test data, they can share this data with other testers and developers, which means that they can run into problems without even knowing that they are conducting security tests.

- Some QA's may work with people who do not know about security issues. They may be new graduates or have previously worked in places where a firewall-protected the software. It is worth raising their awareness - to remind them of the adverse reaction of some well-known companies that have lost user data. When QA testing detects a vulnerability in the application, make sure that they demonstrate it, as well as possible potential exploits. An excellent demonstration tool is BeEF, which shows how powerful a simple XSS vulnerability another user and his browser can give.

- When QA begins to accumulate knowledge, make sure that others will also benefit from it. Give some basic security testing concepts. Give a lesson on how to use the automatic scanner. Testers and developers can learn from QA's, and QA's will consolidate their knowledge on these topics.

Furthermore, experts agreed to the below suggestions made by the researcher in the online survey: -

- Advice SQA professionals to approach security testing with a risk management mindset.

- Work in tandem with architects and IT security teams to map out security vulnerabilities.

SQA experts faced for the interview had mixed feelings about the "No security testing training" problem. Some were neutral, and some were disagreeing with this. The following are the practical suggestions they mentioned to overcome the problem (Please refer to ‘APPENDIX D’ for more details).

- Without sufficient experience, it is complicated to understand why we need all the tools, what are the benefits and drawbacks of each of them, and, most importantly, when we should use one instead of the other. This point of view is suitable both for testers who want to start their career with working with a security tool and for those who are going to pass certification before having at least a solid year of experience in manual testing. Training is an integral part of software testing. When the engineer begins to work, he will understand what his strengths and weaknesses are, and so he must decide what path he wants to take in his career.

- Blended learning is becoming increasingly popular, and as a company, we have seen an absolute increase in this learning method over the past year. Many QA and Dev professionals have improved their capabilities. Blended Learning is an effective combination of online and classroom learning. Many of the 20–20 clients prefer their employees to study locally rather than attend off-site training programs.

- The best way to find more and more web application vulnerabilities and security flaws are to continue to do what QA’s do. However, this is not only about getting “experience” - it is essential that you get an enjoyable experience that you learn from and continuously help you navigate your approaches. As with software development and traditional quality control, do not be afraid of hands-on training or even knowledge transfer from someone who has been testing web security for a while. Participating in RSA, Black Hat, and OWASP exhibitions on information security and web security can help QA’s to improve their web security testing skills.

- There is still much information - and plenty of online resources to help. QA may decide that more focused training will help, for example, various courses from providers such as SANS. There are many security training courses specifically for those responsible for quality assurance, so look for security courses for web developers instead. The so-called penetration testing courses tend to focus on hacking the network, but they often have parts on hacking web applications, so check the course content in advance.

- Among the developers in the company, some know security topics. Ask them to team up with QA's to study the behavior of the application. They should be able to show, for example, that the SQL injection string is not running on the database, and why it is not. If so, then it will be beneficial for both dev and the QA. They can also explain the design of the application and how it designed to protect against attacks. If many people want to know about security, ask them to make a presentation.

Furthermore, experts agreed to the below suggestion made by the researcher in the online survey: -

- Introduced more security testing meet-ups and training for SQA people.

SQA experts faced for the interview had mixed feelings about the "Lack of time" problem. Some were neutral, and some were disagreeing with this. The following are the practical suggestions they mentioned to overcome the problem (Please refer to 'APPENDIX D' for more details).

- The problem of not allocating sufficient time for security testing lies with project managers who do not take into account the quality assurance efforts that required due to either a lack of understanding of the role of quality assurance or a lack of information from the quality assurance team. Typically, a timing problem arises from additional testing cycles required due to poor development quality. The solution lies in the right methods of quality assurance (not quality control), which increase the quality of development.

- Very few negotiations can do with a client in IT. QA always needs to come up with creative ways to overcome this. When they have a deadline/end date, they have to work and plan things backward. All results, obligations that they give to the client must be justified. Security risks must highlight in advance. Quality specialists must learn to protect themselves and the team from this problem.
- When proposing estimates, it is essential to highlight all identified security risk factors for all project stakeholders. The results, obligations that give to the client must be justified — predicting the number of defects and testing cycles based on the level of competence of the development team at an early stage. QA can also add a factor for efforts based on the team's ability level (10% - 15%).
- Introduce Acceptance Test-Driven Development methodology. Similar to what the test team does, practicing development through testing using Acceptance Test Driven Development, the test team writes the tests before the code. Instead of writing a specification in the form of a static document, the test group creates an executable specification that will execute code that needs to write, and that can reorganize and improved.
- When requesting additional testing time from the client / senior management, always show them the value and the expected level of quality after allocating additional time. Follow practices, processes, and historical data.

Furthermore, experts agreed to the below suggestion made by the researcher in the online survey: -

- Form a dedicated QA security taskforce to develop and retain security testing.

SQA experts faced for the interview had mixed feelings about the "Complexity" problem. Some were neutral, and some were disagreeing with this. The following are the practical suggestions they mentioned to overcome the problem (Please refer to 'APPENDIX D' for more details).

- Security testing can be difficult if there is no dramatic and well-functioning structure or process. It is easy to say that something is too complicated, and just not doing it is one of the most typical excuses in the world. The way out is that security experts can establish well-defined testing protocols with easy-to-track processes that use template tools and cheat sheets, where suitable, and leveraging as much current technology and process as practicable.

- The cause many QA teams complain about security testing is that some security teams are trying to impose an entirely new set of testing tools and techniques on QA analysts - this is a sure way to meet opposition. To succeed, it is essential first to understand the existing processes, tools, and methodologies that QA teams use today, and then adapt the security testing protocols, tools, and methods to fit these existing processes.

- We do not expect the QA professionals to be able to put together a complex script to avoid the cross-site scripting library. However, we should reasonably expect the QA to either uses the tool efficiently or follow a well-documented process that has various tests and permutations that allow the QA to think independently and note dubious results for verification by security experts. Many QA professionals begin to take a keen interest in hacking and security testing when they are allowed to learn and test.

- QA engineers do not have to be security hacking experts if they have the right tools. QA engineers should not be experienced hackers; we would prefer that they were not. QA engineers need to be able to use the tools and effectively monitor the process to identify fundamental application security flaws.

- If security testing can drive through QA engineers, instead of taking up some precious cycles that have highly skilled application hackers, this is a huge victory, because security experts can spend their time checking and digging in

applications, and not just testing. Security teams need to make sure that the tools and processes that they set up for their QA people are generic and uncomplicated so that QA resources can effectively focus on what they do.

Furthermore, experts agreed to the below suggestion made by the researcher in the online survey: -

- Reduce SQA individual's lack of exposure to security testing by providing awareness.

SQA experts faced for the interview had mixed feelings about the "Less SQA involvement in system design, requirement gathering, and code review phases" problem. Some were neutral, and some were disagreeing with this. The following are the practical suggestions they mentioned to overcome the problem (Please refer to 'APPENDIX D' for more details).

- Top management must be aware of where to start QA. SDLC / QMS companies should review to entrust the participation of a QA team (at least a QA leader), from the very beginning of the project.
- Product Managers need to collect security requirements for the system in development, they know that they need to do this, they are not trained to collect such requirements, or the industry has not shown itself well examples of effective security requirements.
- QA and security teams should define the non-functional requirements that developers must adhere margin. These non-functional requirements underpin the creation of security-oriented development teams, and when QA teams work hand in hand with the security team from the start, it can be quite powerful. Automated security testing should consider as a critical component in this process, as development management recognizes that quality defects are the starting point for vulnerabilities.
- QA's goal is to help developers and business stakeholders to identify security requirements with sufficient accuracy so that they can test. Requirements should set out in broad, vague terms that they are subject to interpretation. For

example, when registering a system, a strong password must be selected. However, what is the definition of "strong"? How many characters; how many special characters? Moreover, is the password case sensitive? By nature, testers firmly insist on specifics because they understand that vague requirements cannot be verified.

- Use various metrics to understand the clarity level of requirements. The process should configure to highlight the gaps in requirements if they exist. Assign to register defects for requirements. It will encourage the professional to report security defects at the requirement phase.

- Security testing of software/applications should be included in the software development life cycle (SDLC) with routine QA testing. If a security vulnerability is discovered at a later stage either by the customer, this creates inconvenience for the business, and will also cost the business much more to fix it. Therefore, if developers will conduct unit testing when they write new code for a new function, the testing department should also test and confirm that the new function is safe and cannot use.

Furthermore, experts agreed to the below suggestion made by the researcher in the online survey: -

- Facilitate SQA participation in non-QA related phases of the development life cycle.

SQA experts faced for the interview had mixed feelings about the "Lack of management support" problem. Some were neutral, and some were disagreeing with this. The following are the practical suggestions they mentioned to overcome the problem (Please refer to ‘APPENDIX D’ for more details).

- Quality transparency must ensure for senior management. Provide management with innovative numbers related to security testing in a way that they understand.
- Trainimanagement. Management should focus on quality and its importance. Find out the easiest way to communicate with management. Use real-life examples to talk about the importance of QA's involvement as security testers.
- Train and demonstrate to senior management information about the risks, benefits, and costs based on past data, statistics, case studies related to other companies, how they use QA for security testing, and convince management. Based on the goal itself, convincing management was to happen at the very beginning. The requirement for management to communicate quality expectations to other developers/marketing/sales/accounting/ human resources departments. The presence of quality goals in setting goals for the organization at the corporate level. Integrated quality goals.
- The quality assurance manager has to be a sale person to selling the QA. He/She should not expect a formal installation (with plenty of infrastructures to support QA). It should strive to achieve this in the long run. Find out the easiest way to communicate with management. Use the real world and practical examples to talk about the value of QA as a security tester.
- The quality assurance manager should be involved in discussions related to the process and results and needs to introduce for process optimization areas. Try to become part of the decision-making process (should be able to enforce risk reduction measures).

Furthermore, experts agreed to the below suggestion made by the researcher in the online survey: -

- Provideistrong management support.
- Keep the higher management informed by having weekly, monthly progress reviews or awareness meetings.

SQA experts faced for the interview had mixed feelings about the "No project requirements" problem. Some were neutral, and some were disagreeing with this. The following are the practical suggestions they mentioned to overcome the problem (Please refer to 'APPENDIX D' for more details).

- Software security is an essential feature system and should be included in all life cycles. It would be useful to integrate it from the start of software development, i.e., from the requirements phase. To confirm this mitigation, security testing is one of the known methods that require further investigation.
- QA should be involved in requirement generation. So, if there are no project requirements, then QA should be able to implement a work breakdown structure is the most productive and practical way to devise requirements.
- It is hard to work on tests that are not included in project requirements since there are no time allocations for those. So that focuses on requirements as a process, not a result. Companies that focus both on the process and the results are much more successful than those that focus only on the quality of documentation. Focusing on the progress and methods used to develop documentation is essential to gaining economic benefits and success.
- Most companies need better QA staff than they do. The company will become much more effective if they have employees with sufficient competence to work on projects where their skills are needed. Commit to change. Most organizations know that requirements are essential; few people change their routine CIOs should pay attention to the improvement of all areas of people, processes, and tools used to support processes to achieve organizational improvement.

- In the same company, they may have projects which are having security testing requirements. So, it is better to have an SQA pool of people to service projects which are having security testing requirements. On the other hand, QA professional is responsible for software quality. Ensuring the security of the software is the central part of it.

Furthermore, experts agreed to the below suggestion made by the researcher in the online survey: -

- Have an SQA pool of people to service projects which are having security testing requirements.

SQA experts faced for the interview had mixed feelings about the "Lack of motivation" problem. Some were neutral, and some were disagreeing with this. The following are the practical suggestions they mentioned to overcome the problem (Please refer to 'APPENDIX D' for more details).

- The introduction of new technologies can help motivate the quality assurance professional, as acquiring new skills improves not only qualifications but also self-esteem. Besides, it is excellent to help QA in obtaining international certificates confirming their high level of knowledge. It is also lovely to include internal meetings and educational activities for personal development. As practice shows, short-term training is much more productive than annual continuing education courses. The fact is that even if they do not learn a lot during a particular event, they can still get a portion of inspiration to learn and improve their existing skillset.

- The company grows as the projects grow, and more severe customers come for services. Consequently, growth can also be one way to motivate the QA professional. When planning a large-scale project, this becomes a challenge for the company. It is also an impetus for personal development, even if the quality experts have not yet realized this. They may feel overwhelmed by greater responsibility, but the best solution that will help them get the most out of their complex mission is mentoring.

- People who work with technical experts all the time are usually quite introverted if they do not hear a conversation that attracts attention. If this smart guy has something to say, for example, about the security testing tools and concepts, it would be wise to listen carefully and show interest in the subject. At that moment, when QA notices that the development and tester groups collide in the group to talk - they know, this is a miracle and an absolute sign that a lot of new ideas will be offered to them soon.

- Communication with colleagues more than with friends throughout life, makes bosses around the world pay special attention to who correctly works in their companies. The smart choice of experts plays an essential role in motivating testers, as they must collaborate on a human and professional level. This deserves special attention because the tester, which is surrounded by people with the same life values, is always more inspired than the one who irritates.

- To motivate the QA professional, even more, a flexible work schedule is needed. More likely, testers will perform successfully in a project during their peak productive hours, and not during forced ones. However, QA needs to monitor the time zones of customers, because it is useful when the working hours of the selected teams coincide with them.

Furthermore, experts agreed to the below suggestion made by the researcher in the online survey: -

- Motivate SQA people to do security testing sessions during project idle times.

SQA experts faced for the interview had mixed feelings about the "Lower salary scale compared to other IT professions" problem. Some were neutral, and some were disagreeing with this. The following are the practical suggestions they mentioned to overcome the problem (Please refer to ‘APPENDIX D’ for more details).

- This is due to a lack of perception of the values that QA teams bring to projects from project managers. However, the poor quality of quality assurance specialists is also a reason, since many of them lack the appropriate skills to conduct adequate testing/quality control. On most projects, the QA team

seems to play a “second fiddle” for developers. This is not necessarily due to a lack of skills, but mainly due to a lack of confidence and certainty that they play a crucial role in the project team.

- We need to come up with new approaches to make QA life easier. Try to make themselves known by going beyond traditional manual testing and performing the gray box testing methods. Allow people to achieve the required level of competency. Show it as an example as a manager to prove that they did things better.

- Professionals in the Sri Lankan QA community can divide into different groups depending on skill level, experience level. Therefore, the QA salary cannot and should not be generalized. Various specializations must be defined (e.g., test automation, security testing), and remunerat onimust be determined accordingly. There should be a sufficient range of salaries (lower and upper) within each level/destination. This will allow management to provide the required wage growth for high-performing workers.

- Hire more technical specialist for testing. An interview may consist of more technical issues. This will prevent entering button pushers to the quality assurance industry. It will also be done by a professional for those who can do more than developers.

- Allow the QA member to climb the organization ladder as soon as possible. Also, teach them to perform more effective testing/quality control with appropriate skills.

Furthermore, experts agreed to the below suggestion made by the researcher in the online survey: -

- Increased standard of living for the skilled SQA resources.

4.4. Summary of Results

Results of the preliminary and online surveys were useful to identify the significant problems faced by SQA professionals in software security testing. Interview results helped to gather the expert's recommendations as well as for further verification of the identified problems and suggestions. A summary of the online survey distributions and ranks related to the significant problems are showing in Table 4.40. Even though the order of identified problems varied according to the demographic data, they received more than 70% agreeableness from the respondents. The research focus is to do a mapping of problems with suggestions as to the strategy that can use to develop the SQA mindset in software security testing. Hence the researcher considered that 'Neither Agree nor Disagree' as a positive response when ranking the significant problems.

Table 4.40: Summary of online survey problems.

Problem Description	Rank	Distribution	
		Agree	Disagree
Lack of specialized SQA people in security testing	1	96%	4%
Budget	2	94%	6%
Lack of knowledge about security testing fundamentals	3	93%	7%
Lack of detailed information’s and advice	4	93%	7%
No security testing training	5	93%	7%
Lack of time	6	91%	9%
Complexity	7	91%	9%
Less SQA involvement in system design, requirement gathering and code review phases	8	89%	11%
Lack of management support	9	87%	13%
No project requirements	10	87%	13%
Lack of motivation	11	85%	15%
Lower salary scale compared to other IT professions	12	74%	26%

A summary of the online survey distributions related to suggestions to overcome the above significant problems is showing in Table 4.41.

Table 4.41: Summary of online survey suggessions.

Suggestion Description	Distribution	
	Agree	**Disagree**
Form a dedicated QA security task force to develop and retain the security testing mindset among SQA professionals	99%	1%
Recruit detail-oriented and experienced SQA professionals	96%	4%
Provide proper SQA resources regardless of profit margin (e.g., people, tools, environments, etc.)	97%	3%
Familiarize and adapt security testing fundamentals, protocols, tools and methods to fit within existing processes	100%	0%
Maintain a security testing knowledge portal	99%	1%
Advice SQA professionals to approach security testing with a risk management mindset	99%	1%
Working in tandem with architects and IT security teams to map out security vulnerabilities	97%	3%
Introduced more security testing meet-ups and training for SQA people	99%	1%
Form a dedicated QA security taskforce to develop and retain security testing	99%	1%
Reduce SQA individual's lack of exposure to security testing by providing awareness	98%	2%
Facilitate SQA participation in non-QA related phases of the development life cycle (e.g., System design, Requirement gathering, Code review)	99%	1%
Keep the higher management informed by having weekly, monthly progress review or awareness meetings	99%	1%
Provide strong management support	97%	3%
Have SQA pool of people to service projects which are having security testing requirements	99%	1%
Motivate SQA people to do security testing sessions during project idle times	98%	2%
Increased standard of living for the skilled SQA resources	98%	2%

It is required to do a mapping of problems with suggestions as to the strategy that can use to develop the SQA mindset in software security testing. The mapping listed in Table 4.42.

Table 4.42: Mapping of problems with suggestions as to the strategy.

(This table continues to the next page.)

Problem Description	Suggestion Description	Distribution	
		Agree	Disagree
Lack of specialized SQA people in security testing	Form a dedicated QA security task force to develop and retain the security testing mindset among SQA professionals	99%	1%
	Recruit detail-oriented and experienced SQA professionals	96%	4%
Budget	Allocate sufficient funds in the budget to provide proper SQA resources. (e.g., people, tools, environments)	97%	3%
Lack of knowledge about security testing fundamentals	Familiarize and adapt security testing fundamentals, protocols, tools and methods to fit within existing processes	100%	0%
	Maintain a security testing knowledge portal	99%	1%
Lack of detailed information's and advice	Advice SQA professionals to approach security testing with a risk management mindset	99%	1%
	Working in tandem with architects and IT security teams to map out security vulnerabilities	97%	3%
No security testing training	Introduced more security testing meet-ups and training for SQA people	99%	1%
Lack of time	Form a dedicated QA security taskforce to develop and retain security testing	99%	1%
Complexity	Reduce SQA individual's lack of exposure to security testing by providing awareness	98%	2%
Less SQA involvement in system design, requirement gathering and code review phases	Facilitate SQA participation in non-QA related phases of the development life cycle (e.g.,	99%	1%

	System design, Requirement gathering, Code review)		
Lack of management support	Keep the higher management informed by having weekly, monthly progress review or awareness meetings	99%	1%
	Provide strong management support	97%	3%
No project requirements	Have SQA pool of people to service projects which are having security testing requirements	99%	1%
Lack of motivation	Motivate SQA people to do security testing sessions during project idle times	98%	2%
Lower salary scale compared to other IT professions	Increased standard of living for the skilled SQA resources	98%	2%

5. CONCLUSION AND RECOMMENDATIONS

This chapter examines the conclusion and recommendations based on research findings. Section 5.1 illustrates the evaluation of the research objectives. Section 5.2 provides a summary of the research findings and analysis. Sections 5.3 to 5.5 describe the limitations, recommendations, and future directions, respectively.

5.1. Evaluating the Objectives

This section illustrates how the researcher used the preliminary survey, online survey, and interviews to achieve the research objectives.

Objective 1: To identify the significant problems faced by SQA professionals in software security testing

To accomplish this objective, an online survey used to filter out significant problems. Problems were identified during the literature review phase and preliminary survey results. The online survey used to assess the identified problems using percentage and weighted scale approaches. To evaluate this objective, survey responses analyzed in different approaches and viewpoints. Most of the problems identified during the survey were because of the lack of security testing mindset, a lack of skilled/specialized QA resources to cater to the tasks of security testing and, not having advice and training. Another reason is that the manager's lack of understanding of the need for building security into the development life cycle and providing full support for their QA subordinates to achieve the high-quality bar. The research findings are listed in Section 4.4 and Table 4.40.

Objective 2: To identify and present recommendations and suggestions to overcome the identified problems

To accomplish this objective, an online survey and interview sessions used to filter out the key suggestions identified through the literature review phase and preliminary survey findings. Interviews conducted with the SQA experts who are having broad experience in implementing strategies in the IT industry. No single approach suitable for all organizations. Each one must consider the effort’s size, scope, and expectations before jumping in and getting all the QA professionals up to a standard level of

security testing. Creating a security testing strategy for QA is differs widely in small, nimble organizations versus large, heavily spread ones. Most of the problems related to SQA professionals in software security testing can solve with strong management support and continuous education through the use of a dedicated QA security task force and a security testing knowledge portal. The research findings are listed in Section 4.4 and Table 4.41.

5.2. Summary of Contributions

SQA is an essential department for any IT organization because it has an on-going process within the SDLC that routinely checks the developed software to ensure it meets desired quality measures. Ensuring software security is vital when meeting quality measures. So, it is crucial to start leveraging the power of SQA to bring better software security to our industries. According to the research findings, SQA professionals in the IT industry face various problems in software security testing. Hence, it is essential to find out a strategy to overcome those problems.

It is needless to say; no single strategy fits all organizations. Each one must consider the effort's size, scope, and expectations before jumping in and getting all the QA professionals up to a standard level of security testing. Creating a security testing strategy for QA is differs broadly in small, agile organizations versus large, heavily spread ones. Management support is required for any QA teams in various organizations to adopt a security testing mindset. Utilizing any security activity in the software development lifecycle requires it. Because it ensures that it receives the proper attention and funding, it needs to see it through.

Developing a security testing mindset requires more than just going through one or a couple of security training sessions. It is a continuous, iterative process. Someone who takes a few courses in security testing will not necessarily become a successful software security tester. Such testers would only apply those tools and methods taught to them. It is unfair to expect QA testers to task themselves with continuously reading related literature, monitoring related web sites, and correctly applying the new knowledge learned during the regular project cycle. So, it is useful to form a QA security taskforce. This task force goal is to develop and retain the security testing

mindset among QA professionals. To make this work, QA professionals must be motivated to participate. Managers should encourage contribution to the taskforce by listing specific goals. An essential piece of bonding security testing knowledge is the use of a knowledge portal.

Adopting a proper strategy and getting management support are critical factors for success. Table 4.42 shows the mapping of problems with suggestions as to the strategy. This will solve most of the problems faced by the SQA professionals in software security testing.

5.3. Limitations

The main limitation of this research study was the time limit for collecting enough data. Because of this, there may be several limitations to the study. Representativeness of the sample is a limitation of this study since not all organizations included in the sample. The honesty of the respondents will have a significant impact on the results of the study. This will limit observing 100% accurate results.

5.4. Recommendations

As declared in the research objective, it is vital to present the identified recommendations and suggestions to the potential SQA management. By using content analysis, the following vital recommendations identified.

- SQA managers should take actions to form a dedicated QA security taskforce to develop and retain the security testing mindset among QA professionals and build a strong relationship between staff and the taskforce;
    - The indicated recommendation identified during the literature review and filtered out from the online survey and further verified in the interviews. (Please refer to Section 2.6, 4.2.6, and 4.3 for more details.)
- SQA managers should consider recruiting detail-oriented and experienced SQA professionals;
    - The present recommendation identified during the literature review and filtered out from the online survey and further verified in the interviews. (Please refer to Section 2.6, 4.2.6, and 4.3 for more details.)

- SQA managers should set specific goals for their team member to familiarize and adapt security testing fundamentals, protocols, tools and methods to fit within existing processes;
    - This recommendation identified during the literature review and filtered out from the online survey and further verified in the interviews. (Please refer to Section 2.6, 4.2.6, and 4.3 for more details.)
- Product managers should collect actionable security requirements;
    - The indicated recommendation identified during the interviews. (Please refer to Section 4.3 for more details.)
- Managers should motivate QA professionals to participate in security testing discussions;
    - This recommendation filtered out from the online survey and further verified in the interviews. (Please refer to Section 4.2.6 and 4.3 for more details.)
- To bring much additional value and to make discussions lively, managers need to take necessary actions to hire external speakers, especially if they are experts in the security field;
    - The present recommendation carried out during the interviews. (Please refer to Section 4.3 for more details.)
- Management should make attendance mandatory to ensure active participation in security testing discussions;
    - The indicated recommendation carried out during the interviews. (Please refer to Section 4.3 for more details.)

- Managers should encourage supplying to the QA security taskforce by listing specific goals in the selected participants' annual reviews or giving some awards;
    - The present recommendation carried out during the interviews. (Please refer to Section 4.3 for more details.)
- Managers should ensure whether their SQA team members are keeping up to date with the latest training delivered by the taskforce team by having a knowledge portal that contains all of the necessary contact information for security employees who can help answer security-related questions;
    - This recommendation identified during the literature review and filtered out from the online survey and further verified in the interviews. (Please refer to Section 2.6, 4.2.6, and 4.3 for more details.)
- SQA managers should provide tasks to notify appropriate security testing resources on the web, including information about security tools, attacks, vulnerabilities, and tutorials;
    - The indicated recommendation carried out during the interviews. (Please refer to Section 4.3 for more details.)
- SQA managers should provide a set of security testing standards & advice for QA professionals to geared towards the project or organization's best interests;
    - This recommendation carried out during the interviews. (Please refer to Section 4.3 for more details.)
- Introduce security testing fundamentals to the undergraduate level as a subject where the courses should include more practical aspects and hands-on sessions compared to the theory portion;
    - The present recommendation carried out during the interviews. (Please refer to Section 4.3 for more details.)

- Managers should provide remuneration and appreciation for potential/competent SQA professionals;
    - The present recommendation identified during the literature review and filtered out from the online survey and further verified in the interviews. (Please refer to Section 2.6, 4.2.6, and 4.3 for more details.)
- Managers should advice SQA professionals to approach security testing with a risk management mindset;
    - The present recommendation filtered out from the online survey and further verified in the interviews. (Please refer to Section 4.2.6 and 4.3 for more details.)
- Managers should facilitate SQA participation in non-QA related phases of the development life cycle;
    - This recommendation filtered out from the online survey and further verified in the interviews. (Please refer to Section 4.2.6 and 4.3 for more details.)
- Managers should organize team hierarchy in a such a way to have SQA pool of people to service projects which are having security testing requirements;
    - This recommendation filtered out from the online survey and further verified in the interviews. (Please refer to Section 4.2.6 and 4.3 for more details.)
- Managers should allow their team members to familiarize and adapt security testing fundamentals, protocols, tools and methods to fit within existing organizational processes since security testing can be quite tricky if there is no dramatic and well-functioning structure or process;
    - This recommendation filtered out from the literature review and further verified in the online survey and the interviews. Please refer to Section 2.6, 4.2.6, and 4.3 for more details.)

5.5. Future Work

Problems faced by SQA professionals in software security testing is a broad area. It is worthwhile to leverage the power of SQA to enhance software security as it is a growing profession. The following aspects are some suggestions for future research that are related to the SQA profession in software security testing.

- Consider client-side support for the security requirement and other professional categories like Product Manager, Software Engineer, Business Analysis, Project Managers, Support Engineers, DB Administrators, Network Administrators. For example, Product managers should collect security requirements for the system under development. They should be trained to collect such actionable security requirements. Software engineers should know how to design and implement software systems that behave defensively. (It is essential to identify the problems and suggestions to overcome those problems for other professional categories in the IT industry. By improving the other professions, the SQA profession can obtain more benefits. Hence those suggestions can be taken as indirect overcoming methods for SQA problems in software security testing.)
- Organize and carry out focus group interviews to discuss factors in an open forum. (This is helpful to identify more effective suggestions as an industry.)
- Assess how the problems affect the individual's productivity. (This will help employers to improve the employee's productivity by implementing the suitable/required suggestions for the optimal/ selected problems. Further, this will bring many benefits to the employer to improve the organization as a whole.)

REFERENCES

Abela, R. (2017). Web Application Security Testing should be part of QA Testing. [online] Netsparker.com. Available at: https://www.netsparker.com/blog/web-security/web-application-security-tests-included-in-functionality-qa-tests/ [Accessed 21 Feb. 2019].

Basu, E. (2013). What Is A Penetration Test And Why Would I Need One For My Company?. [online] Forbes.com. Available at: https://www.forbes.com/sites/ericbasu/2013/10/13/what-is-a-penetration-test-and-why-would-i-need-one-for-my-company/#68250df018a0 [Accessed 21 Feb. 2019].

Bonver, E. and Cohen, M. (2008). Developing and Retaining a Security Testing Mindset. IEEE Security & Privacy Magazine, 6(5), pp.82-85.

Borodina, K. (2019, March 15). How to Learn Penetration Testing. Retrieved from https://www.codemopolitan.com/learn-penetration-testing/.Borodina, K. (2019, March 15). How to Learn Penetration Testing. Retrieved from https://www.codemopolitan.com/learn-penetration-testing/.

Dowd, M., McDonald, J. and Schuh, J. (2007). The art of software security assessment. Indianapolis, Ind.: Addison-Wesley.

English, R. (2014). Incorporating Web Application Security Testing Into Your Quality Assurance Process. [online] StickyMinds. Available at: https://www.stickyminds.com/article/incorporating-web-application-security-testing-your-quality-assurance-process [Accessed 21 Feb. 2019].

Eriksson, O. (2018, April 18). Do you wish to learn to hack? 5 steps to learning penetration testing. Retrieved from https://medium.com/@oscar.eriks/you-wish-to-know-how-to-hack-5-steps-to-learn-penetration-testing-b9f29699878e.

Frankk,D. 2014. Importance of Software Quality Assurance. [online] Available at: http://www.evancarmichael.com/Technology/6726/Importance-of-Software-Quality-Assurance.html. [Accessed 04 April 2018].

Hrynczak, M. (2016). 13 Steps to Learn and Perfect Security Testing in Your Org. Retrieved from https://www.atlassian.com/blog/archives/13-steps-to-learn-perfect-security-testing-in-your-org

Iqbal, N., & Qureshi, R. J. (2012). Improvement of Key Problems of Software Testing in Quality Assurance. Science International-Lahore (p. 1). Lahore: eprint arXiv: 1202.2506.

JÄNTTI, M. (2008). Difficulties in Managing Software Problems and Defects. (pp. 1-62).Finland: University of Kuopio.

Javed, A., AshfaqQazi, K., Maqsood, M., Shah, K.A. (2012). How to Improve Software Quality Assurance in Developing Countries. Advanced Computing: An International Journal. 3 (2), p1-12.

Kevitt, M. (2008). Best Software Test & Quality Assurance Practices in the project Lifecycle. Dublin: School Computer Applications.

Laskowski, J. (2011). Why software quality assurance and IT security need to work together. [online] Ibm.com. Available at: https://www.ibm.com/developerworks/rational/library/software-quality-assurance-IT-security/index.html [Accessed 21 Feb. 2019].

Lent, J. (2013). Security testing basics: QA professionals take the lead. [online] SearchSoftwareQuality. Available at: http://searchsoftwarequality.techtarget.com/feature/Security-testing-basics-QA-professionals-take-the-lead [Accessed 21 Feb. 2019].

Los, R. (2011). Why QA Doesn't Do Security Testing. [online] Infosecisland.com. Available at: http://www.infosecisland.com/blogview/10736-Why-QA-Doesnt-Do-Security-Testing.html [Accessed 21 Feb. 2019].

Mcgraw, G. (2006). Software Security: Building Security In. 2006 17th International Symposium on Software Reliability Engineering. doi:10.1109/issre.2006.43.

Nasib, S.G. (2005). Factors Affecting Effective Software Quality Management Revisited.ACM SIGSOFT Software Engineering Notes. 30 (2), p1-4.

Potter, B. and McGraw, G. (2004). Software security testing. IEEE Security & Privacy Magazine, 2(5), pp.81-85.

Rosenberg, L. (2002). Software Quality Assurance Engineering at NASA. Aerospace Conference Proceedings, (pp. 5-2569 – 5-2575). Greenbelt.

Sanksoft, C. (2017). Importance of Software Security Testing - The Official 360logica Blog. [online] The Official 360logica Blog. Available at: https://www.360logica.com/blog/importance-of-software-security-testing [Accessed 21 Feb. 2019].

Sigrid, E. (2006). How to save on quality assurance challenges in software testing. Jornadas sobre Testeo de Software (pp. 103-121). Valencia: ITI, Universidad Politecnica de Valencia.

Smith, S. W., & Marchesini, J. (2008). The craft of system security. Upper Saddle River, NJ: Addison-Wesley.

Sri Lanka Information and Communication Technology Agency.2013.National ICT

Takenen, A., & Miller, C. (2008). Fuzzing for software security testing and quality assurance. s.1.: Artec house. doi:https://www.google.com/search?tbm=bks&q=Fuzzing for Software Security Testing and Quality Assurance.

ThinkSys. (2017, June 29). Strategies for Security Testing. Retrieved from https://www.thinksys.com/qa-testing/strategies-for-security-testing/.

Tian-yang, G., Yin-sheng, S. and You-yuan, F. (2010). Research on Software Security Testing, Vol.4 No.9.

Tuteja, M., Dubey. G. (2012). A Research Study on the importance of Testing and Quality Assurance in Software Development Life Cycle (SDLC) Models. Soft Computing and Engineering, (pp. 1-7). Noida.

Whittaker, J. A., & Thompson, H. H. (2006). How to break software security: effective techniques for security testing. Boston: Pearson/Addison Wesley.

Wisseman, S. (2018). Application security and QA: Vulnerabilities are just another defect. [online] TechBeacon. Available at: https://techbeacon.com/app-dev-testing/application-security-qa-why-they-are-better-together [Accessed 21 Feb. 2019].

Wickramasinghe, V., & Jayabandu, S. (2007), Towards workplace flexibility: flexitime arrangements in Sri Lanka, Employee Relations, Vol. 29 No. 6.

Workforce Survey 2013.[pdf] Sri Lanka Information and Communication Technology Agency. Available at: http://www.icta.lk/attachments/article/1247/Final%20Report-WFS.pdf [Accessed 01 March 2018].

Woody, C., Ellison, R., & Nichols, W. (2014, December). Predicting Software Assurance Using Quality and Reliability Measures (Rep.). doi:CMU/SEI-2014-TN-026.

Wysopal, C., & Wesley, A. (2006). The Art of Software Security Testing: Identifying Software Security Flaws.

APPENDIX A: PRELIMINARY SURVEY QUESTIONNAIRE

This is a preliminary survey for my MBA research on 'Leveraging the Power of SQA to Enhance Software Security.' Please fill out this survey as genuinely and in as many details as possible. Thanks, and I appreciate your valuable inputs.

-Section A-

1. **Level in the organization?** (Required)
    - Executive Management
    - Middle Management
    - Tactical Management
    - Engineer/Executive
2. **Type of your company?** (Required)
    - Product Development
    - IT Services
    - Both
3. **Target Market?** (Required)
    - Local Market
    - Overseas Market
    - Both
4. **Size of the SQA Department?** (Required)
    - Less than 10
    - 11-50
    - 51-100
    - More than 100

-Section B-

5. **As per your experience, what are the problems faced by SQA professionals in software security testing?** (Required)

6. **What are your suggestions for overcoming those problems?** (Required)

7. **Do you have any experience in implementing those suggestions? If so, please briefly explain.**

8. **Anything else likes to share?**

It is a great help if you can provide future feedback by participating in the follow-up survey. Please provide your email if you prefer to attempt in the follow-up survey.

APPENDIX B: ONLINE SURVEY QUESTIONNAIRE

Dear Friends,

As a part of my MBA in IT research, I am conducting this survey on 'Leveraging the Power of SQA to Enhance Software Security.' I invite you to participate in this study by completing the following questionnaire. It will take about ~15 min to complete the survey.

This survey is stipulated confidential and anonymous. Your responses will not be personally identified with you, and all findings will appear in the aggregated form. You and your organization will not link in any manner.

Survey Link:
https://docs.google.com/forms/d/e/1FAIpQLSdj2oSAeXMBvgnIvktpv13RIJBNZqpjU2Wgw7DmIB5VVEjsBA/viewform

Your participation in the research survey would be much appreciated. If you have any queries or wish to know more, please feel free to contact me using the details provided below.

Thank you for your time and help in making this study possible.

Sincerely,
Hashantha Jayasekara
hashan.udara90@gmail.com
+94714997282
Dept. of Computer Science and Engineering
University of Moratuwa

1. **To what extent do you agree with the following problems faced by SQA professionals in software security testing? (Required)**

   Complexity (e.g., hard to understand)

   - Strongly Agree
   - Agree
   - Neither Agree nor Disagree
   - Disagree
   - Strongly Disagree

   Lack of motivation

   - Strongly Agree
   - Agree
   - Neither Agree nor Disagree
   - Disagree
   - Strongly Disagree

   Lack of knowledge about security testing fundamentals (e.g., testing tools, common attacks like SQL injections)

   - Strongly Agree
   - Agree
   - Neither Agree nor Disagree
   - Disagree
   - Strongly Disagree

   Lack of detailed information's and advice

   - Strongly Agree
   - Agree
   - Neither Agree nor Disagree
   - Disagree
   - Strongly Disagree

No security testing training

- Strongly Agree
- Agree
- Neither Agree nor Disagree
- Disagree
- Strongly Disagree

Lack of specialized SQA people in security testing

- Strongly Agree
- Agree
- Neither Agree nor Disagree
- Disagree
- Strongly Disagree

Lack of management support

- Strongly Agree
- Agree
- Neither Agree nor Disagree
- Disagree
- Strongly Disagree

Less SQA involvement in system design, requirement gathering and code review phases

- Strongly Agree
- Agree
- Neither Agree nor Disagree
- Disagree
- Strongly Disagree

Lack of time (e.g., due to regular project cycle)

- Strongly Agree
- Agree
- Neither Agree nor Disagree
- Disagree
- Strongly Disagree

No project requirements

- Strongly Agree
- Agree
- Neither Agree nor Disagree
- Disagree
- Strongly Disagree

Lower salary scale compared to other IT professions

- Strongly Agree
- Agree
- Neither Agree nor Disagree
- Disagree
- Strongly Disagree

Budget (e.g., less allocation of SQA people, tools, environments)

- Strongly Agree
- Agree
- Neither Agree nor Disagree
- Disagree
- Strongly Disagree

2. **Any other problems that you may have experienced/observed?** (Required)

3. **To what extent do you agree with the following suggestions to overcome the problems faced by SQA professionals in software security testing?** (Required)

   Form a dedicated QA security taskforce to develop the security testing mindset among SQA people

   - o Strongly Agree
   - o Agree
   - o Neither Agree nor Disagree
   - o Disagree
   - o Strongly Disagree

   Maintain a security testing knowledge portal

   - o Strongly Agree
   - o Agree
   - o Neither Agree nor Disagree
   - o Disagree
   - o Strongly Disagree

   Motivate SQA people to do security testing sessions during project idle times

   - o Strongly Agree
   - o Agree
   - o Neither Agree nor Disagree
   - o Disagree
   - o Strongly Disagree

   Familiarize and adapt security testing fundamentals, protocols, tools and methods to fit within existing processes

   - o Strongly Agree
   - o Agree
   - o Neither Agree nor Disagree
   - o Disagree
   - o Strongly Disagree

Introduced more security testing meet-ups and training for SQA people

- Strongly Agree
- Agree
- Neither Agree nor Disagree
- Disagree
- Strongly Disagree

Reduce SQA individual's lack of exposure to security testing by providing awareness

- Strongly Agree
- Agree
- Neither Agree nor Disagree
- Disagree
- Strongly Disagree

Advice SQA professionals to approach security testing with a risk management mindset

- Strongly Agree
- Agree
- Neither Agree nor Disagree
- Disagree
- Strongly Disagree

Recruit detail-oriented and experienced SQA professionals

- Strongly Agree
- Agree
- Neither Agree nor Disagree
- Disagree
- Strongly Disagree

Provide strong management support

- Strongly Agree
- Agree
- Neither Agree nor Disagree
- Disagree
- Strongly Disagree

Keep the higher management informed by having weekly, monthly progress review or awareness meetings

- Strongly Agree
- Agree
- Neither Agree nor Disagree
- Disagree
- Strongly Disagree

Facilitate SQA participation in non-QA related phases of the development life cycle (e.g., system design, requirement gathering, code review)

- Strongly Agree
- Agree
- Neither Agree nor Disagree
- Disagree
- Strongly Disagree

Working in tandem with architects and IT security teams to map out security vulnerabilities

- Strongly Agree
- Agree
- Neither Agree nor Disagree
- Disagree
- Strongly Disagree

Have SQA pool of people to service projects which are having security testing requirements

- Strongly Agree
- Agree
- Neither Agree nor Disagree
- Disagree
- Strongly Disagree

Increased standard of living for the skilled SQA resources

- Strongly Agree
- Agree
- Neither Agree nor Disagree
- Disagree
- Strongly Disagree

Allocate sufficient funds in the budget to provide proper SQA resources. (e.g., people, tools, environments)

- Strongly Agree
- Agree
- Neither Agree nor Disagree
- Disagree
- Strongly Disagree

4. **Any other suggestions to overcome the problems faced by SQA professionals in software security testing? Please also mention the problem(s) that can be overcome by applying your suggestions.** (Required)

5. **Do you have any experience in implementing the above any suggestion(s)? If so, please briefly explain.** (Required)

6. **Level in the organization?** (Required)
    - Executive Management
    - Middle Management
    - Tactical Management
    - Engineer/Executive

7. **Gender?** (Required)
    - Male
    - Female

8. **Type of your company?** (Required)
    - Product Development
    - IT Services
    - Both

9. **Target Market?** (Required)
    - Local Market
    - Overseas Market
    - Both

10. Size of the SQA Department? (Required)

- Less than 10
- 11-50
- 51-100
- More than 100

It is a great help if you can provide further feedback by participating in a follow-up interview. Please provide your email if you wish to participate in the follow-up interview.

APPENDIX C: INTERVIEW QUESTIONNAIRE

Dear Sir/ Madam,

I am Hashantha Jayasekara - A student at the University of Moratuwa MBA in IT 2017 batch. Moreover, I am currently working as a Senior Software Automation Engineer at Xinfinit (PVT) LTD.

I have conducted a survey for my MBA in IT research on "**Leveraging the Power of SQA to Enhance Software Security**." Through the survey, I have identified significant problems faced by SQA professionals in software security testing, and I am in progress in identifying the solutions to overcome those problems. Appreciate if you can provide me your valuable inputs to complete this study by providing an appointment to have an interview with you.

Please let me know the possibility.

Many thanks in advance

Hashantha Jayasekara
+94714997282/ +94713986487

1. Details of the contacted person

Name:
Designation:
Company Name:
Email Address:
Contact Number:

2. Details of the company

Type: (Product Development/ IT Services / Both)
Target Market Segment: (Local Only/ Overseas Only/ Both)
Size of the QA Department:

3. What is your agreeableness (agree/ neutral/ disagree) and suggestions(s) to overcome the following problems faced by SQA professionals in software security testing?

   Complexity (e.g., hard to understand)

   Lack of motivation

   Lack of knowledge about security testing fundamentals (e.g., testing tools, common attacks like SQL injections)

   Lack of detailed information's and advice

   No security testing training

   Lack of specialized SQA people in security testing

   Less SQA involvement in system design, requirement gathering and code review phases

   Lack of management support

   Lack of time (e.g., due to regular project cycle)

No project requirements

Lower salary scale compared to other IT professions

Budget (e.g., less allocation of SQA people, tools, environments)

4. Any other suggestions to improve this study?

APPENDIX D: INTERVIEW RESULTS

Problem	Interviewer Profile	Suggestions/Recommendations
Lack of specialized SQA people in security testing	Expert 01	All about how management builds competencies within the organization. Many initiatives need to be done in these areas. Identify areas for the year (automation, productivity, security, mobility, and cloud testing) and organize internal training and invite external trainers to conduct training. Should not force people to become specialized QA. Individuals also need to spend some time to achieve this.
	Expert 02	Inside the workplace, training should be done. Both technical and subject. Arrange dedicated QA's for security testing. So that they can develop and maintain knowledge. Collaborate support from developers and implementation engineers.
	Expert 03	Individuals should see the current threat to the industry right now. They need to know their level of competency to meet demand. Training programs will not help build people. It is all about the training that they receive at work in the organization. Training people to perform POC to a certain extent and allows them to perform actions in real-time. The introduction of more QA-related modules at the degree level, such as security testing, and providing them with information about QA's involvement in security testing will help to some extent.
	Expert 04	The lack of qualified/specialized QA resources is a real problem faced by countries such as Sri Lanka. This would be a more laborious task if it had to be addressed at the company level. Instead, this should be addressed at the industry level. There should be initiatives to identify 3 or 4 core specialized skills. The high school curriculum at universities must be modified to cater for this. These courses should include more practical aspects and practical exercises in comparison with the theoretical part. Students should be given assignments for developing automation packages or security testing scripts. When these new graduates join the company, this hands-on knowledge will allow them to start working directly in security tests, automation projects, or productivity projects (initially with

		older resources). This will reduce the burden on companies that will spend more time and money training new graduates from scratch.
	Expert 05	Develop specialized SQA specialists in the field of security testing, providing the necessary training in the workplace, and they should be conducted both from a technical point of view and from the subject area.
Budget	Expert 01	-When offering estimates, all security risk factors need to be identified, highlighted, quantified, and communicated to the client. For a particular module, this is the number of hours that we must spend, will determine the number of changes for which we need so much time. For a change, we determined the time. If it exceeds, contact the customer and tell us the extra time that we need. We place CR and get the time and budget approved. - Determining the number of security tests for each module. Put them in 3 buckets (for example, 3-4 hours, 7-8 hours, 12 hours). Depending on how long it took. Conduct a series of Q&A sessions, and make assumptions on points which are not cleared. The number of defects is predicted depending on the competency level of the development team. - For additional hours, hold a meeting with customers and get an approved budget for things we cannot control. - Reveals the expected quality level after allocating this much amount of time and budget. - When developers spend time testing, we get together and develop a joint plan. During development, we do QA, invest in automation, add some additional resources to overcome. - When we get the number of hours for quality assurance, check the available resources / get the expected quality from the client/area in which QA (the service can execute, align resources) also shows the client, if you give me this extra time, I can include these things, this is a risk that I can reduce, this is the level of quality that I can achieve.
	Expert 02	- There is a way to do a good job, and there are always ways to do a better job. To do the best job (in this case, increase productivity and effectiveness in QA), various resources are required, such as human resources, tools, environment. This means

		that there is more cost to the project or company. Management must be convinced that QA is a vital component of the project, and it is essential to identify security weaknesses. Encourage customers to reduce resources over time with innovative and creative approaches. - People do not understand the value of QA. So, it is better to justify ROI. Talk to people who set aside a budget on more dollar terms. Justify your resource requirements in the ROI equation (return on investment QA).
	Expert 03	Again, the problem is a lack of understanding of "what is required" concerning QA / QC. In my experience, the planning stages of a project do not adequately evaluate these resources. At times, the quality assurance team must be blamed, as they cannot come to concrete plans to convince the PM of the risks of not adhering to proper quality control following a specific project. The solution is to have better project planning, which includes QA / QC planning.
	Expert 04	Make sure that the developer appreciates the participation of QA's to ensure the security of the project. They also need to demand quality. It is about productivity (how fast you can ship to market).
	Expert 05	-Considering people to give the right estimates for project planning. When evaluating, use a re-prospective. It helps you evaluate yourself and determine the right speed and effectiveness. You can also define the same criteria for a team. Execute expert judgment for assessment. Prioritization test scripts. - Use breath first approach/methodology. It can run through important functions / basic flow (smoke test) to quickly identify critical/important security problems. -Share knowledge with developers in the early stages to get quality releases in the early stages, expanding their capabilities. The problems will be fixed before you receive the release. Prevention is always better than cure. Plan a poker approach.
Lack of knowledge about security	Expert 01	It is essential to know the application you are testing to assess the risks. Everything else will assume that you possess this knowledge - the technologies used by the application, the profile of different users, the capabilities that you should and should not have at

testing fundamentals		different access levels, and the potential data that is stored in the application. When it comes to understanding the security terms and definitions, OWASP is a great source. At first, the volume of terms and concepts can be overwhelming, so focus on understanding some terms, preferably those that are most likely to apply to your application. Examples are XSS, XSRF, SQL injection, and path traversal. CWE / SANS Top 25 lists the most common and critical errors that cause vulnerabilities.
	Expert 02	An excellent way to start learning is to start testing a software/application with known vulnerabilities, where you can find instructions on how to detect them. I prefer Google Gruyere, which has separate lessons to cover every concept. You can look at the tips to help you find the vulnerability, and answers if necessary. Some other options are OWASP WebGoat and Damn Vulnerable Web App.
	Expert 03	It is likely that between the developers in your company, some know security topics. Ask them to team up with you to study the behavior of the application. They should be able to show, for example, that the SQL injection string is not running on the database, and why it is not. If so, then it will be beneficial for both QA. They can also teach you the design of the application and how it designed to protect against attacks. If many people want to know about security, ask them to make a presentation.
	Expert 04	An excellent commercial option is the Burp Scanner; There are also free options like OWASP ZAP and Google RatProxy. They work by routing HTTP traffic to and from the application through a proxy server and then resending requests with various attack attempts that replace the original values. This may be an effective way to detect specific classes of vulnerability in a short period, but it is essential to understand (and make sure your stakeholders understand) that it is not a magic wand. The tool is naive and does not know the business logic of applications - it only reproduces requests and checks the answers.
	Expert 05	-There are many types of vulnerabilities that cannot and will not be detected using tools, and the use of a scanning tool does not at all replace the need for manual security testing.

		-Automated tools, even expensive ones, only detect relatively simple vulnerabilities, and they usually produce a lot of “noise” or false positives. You should know enough about security vulnerabilities to be able to evaluate each discovery of an automated tool. Taking a scanner report and analyze it and learn from it is the best thing you can do to understand security test fundamentals.
Lack of detailed information’s and advice	Expert 01	Like any skill, you will get higher with practice. Once you begin to find vulnerabilities in the application, you will begin to understand where they can be in the future, and you can raise them in advance. Performing regular code checks will increase the efficiency of your scanner.
	Expert 02	More focused training would help, such as various course providers such as SANS. There are security training courses specifically for those responsible for quality assurance, so look for security courses for web developers instead. So-called "penetration testing" courses tend to focus on hacking the network, but they often have parts on hacking web applications, so check the course content in advance.
	Expert 03	When testing a feature, you are likely to create test data. Instead of using "test1", "test2" or the names of cartoon characters, get into the habit of using attack lines. Thus, you will find that you discover vulnerabilities almost by accident, only by using the function. If you have an automated tool or an import file that provides test data, do the same. You can share this data with other testers and developers, which means that they can run into problems without even knowing that they are conducting security tests.
	Expert 04	You may work with people who do not know about security issues. They may be new graduates or have previously worked in places where a firewall-protected the software. It is worth raising their awareness - to remind them of the negative reaction of some well-known companies that have lost user data. When your testing detects a vulnerability in the application, make sure that you demonstrate it, as well as possible potential exploits. An excellent demonstration tool is BeEF, which shows how powerful a simple XSS vulnerability another user and his browser can give you.
	Expert 05	When you begin to accumulate knowledge, make sure that others will also benefit from it. Give some

		basic security testing concepts. Give a lesson on how to use the automatic scanner. Testers and developers can learn from you, and you will consolidate your knowledge on these topics.
No security testing training	Expert 01	-Without sufficient experience, it is complicated to understand why we need all the tools, what are the benefits and drawbacks of each of them, and, most importantly, when we should use one instead of the other. -This point of view is suitable both for testers who want to start their career with working with a security tool and for those who are going to pass certification before having at least a solid year of experience in manual testing. -Training is an integral part of software testing. When the engineer begins to work, he will understand what his strengths and weaknesses are, and so he must decide what path he wants to take in his career.
	Expert 02	Blended learning is becoming increasingly popular, and as a company, we have seen an absolute increase in this learning method over the past year. Many QA and Dev professionals have improved their capabilities. Blended Learning is an effective combination of online and classroom learning. Many of the 20–20 clients prefer their employees to study locally rather than attend off-site training programs.
	Expert 03	The best way to find more and more web application vulnerabilities and security flaws are to continue to do what you do. However, this is not only about getting “experience” - it is essential that you get an enjoyable experience that you learn from and continuously help you navigate your approaches. As with software development and traditional quality control, do not be afraid of hands-on training or even knowledge transfer from someone who has been testing web security for a while. Participating in RSA, Black Hat, and OWASP exhibitions on information security and web security can help you improve your web security testing skills.
	Expert 04	There is still much information - and plenty of online resources to help. You may decide that more focused training will help, for example, various courses from providers such as SANS. There are many security training courses specifically for those

		responsible for quality assurance, so look for security courses for web developers instead. The so-called penetration testing courses tend to focus on hacking the network, but they often have parts on hacking web applications, so check the course content in advance.
	Expert 05	It is likely that between the developers in your company, some know security topics. Ask them to team up with you to study the behavior of the application. They should be able to show, for example, that the SQL injection string is not running on the database, and why it is not. If so, then it will be beneficial for both QA. They can also teach you the design of the application and how it designed to protect against attacks. If many people want to know about security, ask them to make a presentation.
Lack of time	Expert 01	The problem of not allocating sufficient time for security testing lies with project managers who do not take into account the quality assurance efforts that are required due to either a lack of understanding of the role of quality assurance or a lack of information from the quality assurance team. Typically, a timing problem arises from additional testing cycles required due to poor development quality. The solution lies in the right methods of quality assurance (not quality control), which increase the quality of development.
	Expert 02	Very few negotiations can be done with a client in IT. You always need to come up with creative ways to overcome this. When you have a deadline/end date, you have to work and plan things backward. All results, obligations that you give to the client must be justified. Security risks must be highlighted in advance. Quality specialists must learn to protect themselves and the team from this problem.
	Expert 03	When proposing estimates, it is essential to highlight all identified security risk factors for all project stakeholders. The results, obligations that are given to the client must be justified. Predicting the number of defects and testing cycles based on the level of competence of the development team at an early stage. You can also add a factor for efforts based on the team's ability level (10% -15%).
	Expert 04	Introducing the Acceptance Test-Driven Development methodology. Similar to what the test

		team does, practicing development through testing using Acceptance Test Driven Development, the test team writes the tests before the code. Instead of writing a specification in the form of a static document, the test group creates an executable specification that will execute code that needs to be written, and that can be reorganized and improved.
	Expert 05	When requesting additional testing time from the client / senior management, always show them the value and the expected level of quality after allocating additional time. Follow practices, processes, and historical data.
Complexity	Expert 01	Security testing can be difficult if there is no dramatic and well-functioning structure or process. It is easy to say that something is too complicated, and just not doing it is one of the most typical excuses in the world. The way out is that security experts can establish well-defined testing protocols with easy-to-track processes that use template tools and cheat sheets, where suitable, and leveraging as much current technology and process as practicable.
	Expert 02	Many SQA teams complain about security testing is that security teams try to force a new set of testing tools and techniques on QA analysts - this is a sure way to meet opposition. To succeed, it is essential first to understand the existing processes, tools, and methodologies that QA teams use today, and then adapt the security testing protocols, tools, and methods to fit these existing processes and be minimally invasive or disruptive.
	Expert 03	We do not expect the QA professionals to be able to put together a complex script to avoid the cross-site scripting library. Nevertheless, we should reasonably expect the QA to either uses the tool efficiently or follow a well-documented process that has various tests and permutations that allow the QA to think independently and note dubious results for verification by security experts. Many QA professionals begin to take a keen interest in hacking and security testing when they are allowed to learn and test.
	Expert 04	QA engineers do not have to be security hacking experts if they have the right tools. QA engineers should not be experienced hackers; we would prefer that they were not. QA engineers need to be able to use the tools and effectively monitor the process to identify fundamental application security flaws.

	Expert 05	If security testing can be driven through QA engineers, instead of taking up some precious cycles that have highly skilled application hackers, this is a huge victory, because security experts can spend their time checking and digging in applications, and not just testing. Security teams need to make sure that the tools and processes that they set up for their QA people are generic and uncomplicated so that QA resources can effectively focus on what they do.
Less SQA involvement in system design, requirement gathering and code review phases	Expert 01	Top management must be aware of where to start QA. SDLC / QMS companies should be reviewed to entrust the participation of a QA team (at least a QA leader), from the very beginning of the project.
	Expert 02	QA and security teams should define the non-functional requirements that developers must adhere to. These non-functional requirements underpin the creation of security-oriented development teams, and when QA teams work hand in hand with the security team from the start, it can be quite powerful. Automated security testing should be considered a key component in this process, as development management recognizes that quality defects are the starting point for vulnerabilities.
	Expert 03	QA's goal is to help developers and business stakeholders identify security requirements with sufficient accuracy so that they can be tested. Requirements are set out in broad, vague terms that they are subject to interpretation. For example, when registering a system, a strong password must be selected. However, what is the definition of "strong"? How many characters; how many special characters? Moreover, is the password case sensitive? By nature, testers firmly insist on specifics because they understand that vague requirements cannot be verified.
	Expert 04	Use various metrics to understand the clarity level of requirements. The process should be configured to highlight the gaps in requirements if they exist. Assign to register defects for requirements. This will encourage the professional to report security defects at the requirement phase.
	Expert 05	Security testing of software/applications should be included in the software development life cycle (SDLC) with routine QA testing. If a security vulnerability is discovered at a later stage either by the customer, this creates inconvenience for the business, and will also cost the business much more

		to fix it. Therefore, if developers will conduct unit testing when they write new code for a new function, the testing department should also test and confirm that the new function is safe and cannot be used.
Lack of management support	Expert 01	Quality transparency must be ensured for senior management. Providing management with innovative numbers related to security testing in a way that they understand.
	Expert 02	Train management. Management should focus on quality and its importance. Find out the easiest way to communicate with management. Use real-life examples to talk about the importance of QA's involvement as security testers.
	Expert 03	Train and demonstrate to senior management information about the risks, benefits, and costs based on past data, statistics, case studies related to other companies, how they use QA for security testing, and convince management. Based on the goal itself, convincing management was to happen at the very beginning. The requirement for management to communicate quality expectations to other developers/marketing/sales/accounting/ human resources departments. The presence of quality goals in setting goals for the organization at the corporate level. Integrated quality goals.
	Expert 04	The quality assurance manager has to be a sale person to selling the QA. You should not expect a formal installation (with plenty of infrastructures to support QA). It should strive to achieve this in the long run. Find out the easiest way to communicate with management. Use the real world and practical examples to talk about the value of QA as a security tester.
	Expert 05	The quality assurance manager should be involved in discussions related to the process and results and needs to introduced for process optimization areas. Try to become part of the decision-making process (should be able to enforce risk reduction measures).
No project requirements	Expert 01	Software security is an essential feature system and should include in all life cycles. It would be useful to integrate it from the start of software development, i.e., from the requirements phase. To confirm this mitigation, security testing is one of the known methods that require further investigation.

	Expert 02	QA should be involved in requirement generation. So, if there are no project requirements, then QA should be able to implement a work breakdown structure is the most productive and practical way to devise requirements.
	Expert 03	It is hard to work on tests that not included in project requirements since there are no time allocations for those. So that focuses on requirements as a process, not a result. Companies that focus both on the process and the results are much more successful than those that focus only on the quality of documentation. Focusing on the progress and methods used to develop documentation is essential to gaining economic benefits and success.
	Expert 04	Most companies need better IT staff than they do. Your company will become much more effective if you have employees with sufficient competence to work on projects where their skills are needed. Commit to change. Most organizations know that requirements are essential; few people change their routine CIOs should pay attention to the improvement of all areas of people, processes, and tools used to support processes to achieve organizational improvement.
	Expert 05	In the same company, you may have projects which are having security testing requirements. So, it is better to have an SQA pool of people to service projects which are having security testing requirements. On the other hand, QA professional is responsible for software quality. Ensuring the security of the software is the main part of it.
Lack of motivation	Expert 01	The introduction of new technologies can help motivate the quality assurance professional, as acquiring new skills improves not only qualifications but also self-esteem. Besides, it is excellent to help your QA in obtaining international certificates confirming their high level of knowledge. It is also lovely to include internal meetings and educational activities in your personal development. As practice shows, short-term training is much more productive than annual continuing education courses. The fact is that even if they do not learn a lot during a particular event, they can still get a portion of inspiration to learn and improve their existing skillset.

	Expert 02	The company grows as the projects grow, and more serious customers come for services. Consequently, growth can also be one way to motivate the QA professional. When you are planning a large-scale project, this becomes a challenge for the company. It is also an impetus for personal development, even if your quality experts have not yet realized this. They may feel overwhelmed by greater responsibility, but the best solution that will help them get the most out of their complex mission is mentoring.
	Expert 03	People who work with technical experts all the time are usually quite introverted if they do not hear a conversation that attracts attention. If this smart guy has something to say, for example, about the security testing tools and concepts, it would be wise to listen carefully and show interest in the subject. At that moment, when you notice that the development and tester groups collide in the group to talk - you know, this is a miracle and an absolute sign that a lot of new ideas will be offered to you soon.
	Expert 04	Communication with colleagues more than with friends throughout life, makes bosses around the world pay special attention to who correctly works in their companies. The smart choice of experts plays an essential role in motivating testers, as they must collaborate on a human and professional level. This deserves special attention because the tester, which is surrounded by people with the same life values, is always more inspired than the one who irritates.
	Expert 05	To motivate the QA professional, even more, a flexible work schedule is needed. More likely, testers will perform successfully in your project during their peak productive hours, and not during forced ones. However, you need to monitor the time zones of customers, because it is useful when the working hours of the selected teams coincide with them.
Lower salary scale compared to other IT professions	Expert 01	This is due to a lack of perception of the values that QA teams bring to projects from project managers. However, the poor quality of quality assurance specialists is also a reason, since many of them lack the appropriate skills to conduct adequate testing/quality control. On most projects, the QA team seems to play a “second fiddle” for

		developers. This is not necessarily due to a lack of skills, but mainly due to a lack of confidence and certainty that they play a vital role in the project team.
	Expert 02	We need to come up with new approaches to make QA life easier. Try to make yourself known by going beyond traditional manual testing and performing the gray box testing methods. Allow people to achieve the required level of competency. Show it as an example as a manager to prove that they did things better.
	Expert 03	Professionals in the Sri Lankan QA community can divide into different groups depending on skill level, experience level. Therefore, the QA salary cannot and should not be generalized. Various specializations must be defined (e.g., test automation, security testing), and remuneration must determine accordingly. There should be a sufficient range of salaries (lower and upper) within each level/destination. This will allow management to provide the required wage growth for high-performing workers.
	Expert 04	Hiring a more technical specialist for testing. An interview may consist of more technical issues. This will prevent entering button pushers to the quality assurance industry. It will also be done by a professional for those who can do more than developers.
	Expert 05	Allow the QA member to climb the organization ladder as soon as possible. Also, teach them to perform more effective testing/quality control with appropriate skills.