Birlesik Dünya Arastırma BD CENTER Innovasyon ve Yayıncılık Merkezi

Global Journal of Computer Sciences: Theory and Research

Global Journal of computer sciences Theory and Research

Volume 12, Issue 1, (2022) 12-22 www.gjcs.eu

The development of a web-based application security testing framework in Addis Ababa, Ethiopia

Samrawit Yimer [a *], Addis Ababa Science and Technology University Addis Ababa, Ethiopia.
Beakal Gizachew [b], Addis Ababa Science and Technology University Addis Ababa, Ethiopia.

Suggested Citation:
Yimer, S. & Gizachew, B. (2022). The development of a web-based application security testing framework in Addis Ababa, Ethiopia. *Global Journal of Computer Sciences: Theory and Research*. *12*(1), 12-22. https://doi.org/10.18844/gjcs.v12i1.6762

Received from December16, 2021; revised from February 25, 2022; accepted from April 11, 2022.
Selection and peer-review under responsibility of Assist. Prof. Dr. Ezgi Pelin Yıldız, Kafkas University,

Abstract

The web-based application is the most significant platform to promote business. Nowadays the number of customers of web-based applications increases rapidly. This study aims to use both qualitative and quantitative research methodology to study the challenge the software companies in Addis Ababa, Ethiopia are facing, in the process of software security testing while developing a web-based application. Most software company and organization in Ethiopia test the web-based application at the end of the SDLC which make it less secure. Data for this study was collected using questionnaires and an interview. This research was applied in eight software companies and in two well-known organizations that develop their software. The contribution of this study is to mitigate the challenges the developer has when they perform security testing in a WBA. This study also contributes to having visible data about security testing performance and processes in Addis Ababa, Ethiopia.

Keywords: Software securıty testıng, Web-based application, Case of addıs Ababa.

* ADDRESS FOR CORRESPONDENCE: **Samrawit Yimer,** Addis Ababa Science and Technology University Addis Ababa, Ethiopia,
Email address: yimersam@gmail.com

1. Introduction

The rapid growth of the internet, mainly on the World Wide Web has made security one of the most important issues, especially for web-based applications [1]. People from different places and backgrounds use the services made by web-based applications. For example, e-learning, banking system, different online payments, e-shopping, so and so on [2]. Due to the increase in customers and different services that provide by web-based applications became our day-to-day activity. It seems like most companies understood the need for software for their daily activity. In a web application, software vulnerability can be reduced by applying a security test in all phases of SDLC.

The main purpose of software security testing is to find vulnerabilities and loopholes in the software before it causes loss of information [3]. In security testing finding vulnerabilities and threats helps the software not be exploited. If the software is successfully exploited, both functionality and non-functionality of the software will be manipulated or stopped. There are different kinds of software security testing such as penetration testing, vulnerability assessment, audit testing, code review, security Auditing, Ethical hacking, Risk Assessment, security scanner so, and so on [4-6]. The most commonly applied security testing methods today are expensive and are every so often too complicated with their activities and different phases. Because of the complexity issue, some developers often tend to neglect the software security testing process.

1.1. Related Literature Review

The security framework should be suitable for both small and large software projects. There are some popular methodologies in the software security community that are used in different scenarios [7-8]. A typical framework usually consists of a series of steps that guide the team from the planning to the implementation phase in WBA development. The most widely known frameworks are the Open Web Application Security Project (OWASP) and the National Institution of Standard Technology (NIST) Secure Software Development Framework (SSDF) [9-12].

OWASP focus on the security task in each phase but in most developing country, the main problem is the lack of management. OWASP did not include the project management responsibility and rules that should be followed in each phase of the SDLC [13]. In Ethiopia, there were two researchers Shimelis Tamitu Duressa[14] and Habtamu Girma Debebe [15] that focus on software testing and software security testing using penetration testing [16]. In their research, most companies in Ethiopia do not implement proper testing in a WBA. In their research from eleven E-governmental websites, all the site was vulnerable to SQL injection ad cross-site script attack. The researcher suggests implementing security testing in each phase without implementing an integrated framework.

1.2. Purpose of study

The web-based application is the most significant platform to promote business. Nowadays the number of customers of web-based applications increases rapidly. People use it for different reasons whether to pay for school or to reserve a hotel. There are many vulnerabilities of a web-based application are reported every day. Despite these vulnerabilities, there is a huge responsibility and burden on the web-based application developer team. Most of the vulnerability comes from the lack of security testing in the process of development. The programmer has the luck of adopting security testing in every phase of the software development life cycle. Most companies in Ethiopia test their software at the last phase of the software development life cycle. There is a different framework, tool, and technique that help the developer team to adopt security testing in every phase but there is a poor adoption and usability in the company. This study aims to use both qualitative and quantitative research methodology to study the challenge the software companies in Addis Ababa, Ethiopia are facing, in the process of software security testing while developing a web-based application.

2. Materials and Method

2.1. Research Design

To address the major issue in this study both qualitative and quantitative research approach methods have been applied. The qualitative research method enables us to gain a real-life contextual understanding and the challenges the software company in Ethiopia are facing. The quantitative research method purposes of measuring the magnitude and frequency of using software security testing methods, tools, and techniques in the software company of Ethiopia.

2.2. Participants and Sampling

This research was applied in eight software companies and in two well-known organizations that develop their software. The samples were chosen across different application domains like banking/finance, retail/eCommerce, etc in Ethiopia. Of the ten companies, there were fifty employees and the project manager were participating in this process and there was also a one-to-one interview with each software company.

2.3. Data Collection instrument

Both qualitative and quantitative data were collected to address the research question. The literature review ware conducted to gather the current status of software security testing. By conducting the literature review, a set of questionnaires and interviews was set to insight into the existing practice of software security testing and the challenge they are facing. The questionnaires were used to gather data to achieve the quantitative objective of the study so that it can achieve a mixed research methodology technique. Questionnaires are a research instrument with a list of questions that helps to get a formal response from the sample. The questionnaire contains 22 questions that were developed after the literature is done.

*2.3.1. **Data Cleansing***

Data cleansing is the process of removing irrelevant or unrelated attributes collected from the literature review. There were eliminated data that does not add and that does not cause harm to the research. Those removed data will not cause any harm.

2.4. Data Analysis

The collected data from the literature review and the questionnaire and interview were summarized by using statistical methods like percentages and charts, and frequency distribution. The findings from qualitative data gatherer through interviews were coded and summarized to find the survey data. While the quantitative data analysis was achieved using SPSS (Statically package for social sciences).

*2.4.1. **Validation the Framework***

Finally, the study as shown in the research questions and objectives is to propose a security testing framework that helps the software development team and the company to will use integer security testing in all phases of SDLC while developing a WBA.

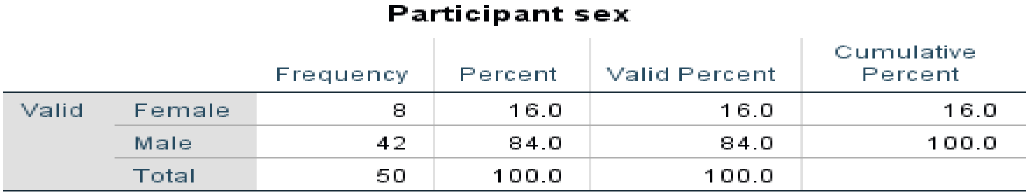
3. Results

3.1. Respondents' Demographic Data

Gender: - The data collected indicate that the frequency distribution of participant's gender variable disclosed that the majority 84% of the participant were male and the rest 16% only are female.

Table 1

Frequency distribution gender of the participants



Participant sex

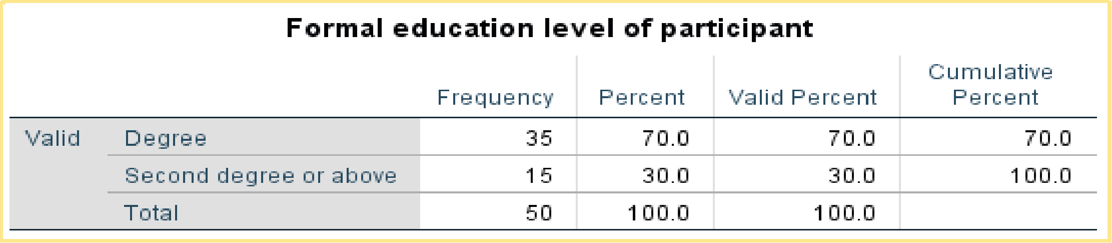
		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Female	8	16.0	16.0	16.0
	Male	42	84.0	84.0	100.0
	Total	50	100.0	100.0	

3.1.1. *Education Status of participants*

In this study, most of the 50 participants e thirty-five of them have Bachler Degree (70%)and the rest fifteen of the participants (30%) have a degree of masters. In this study, no participant has a diploma degree or below.

Table 2

Formal education level of participant



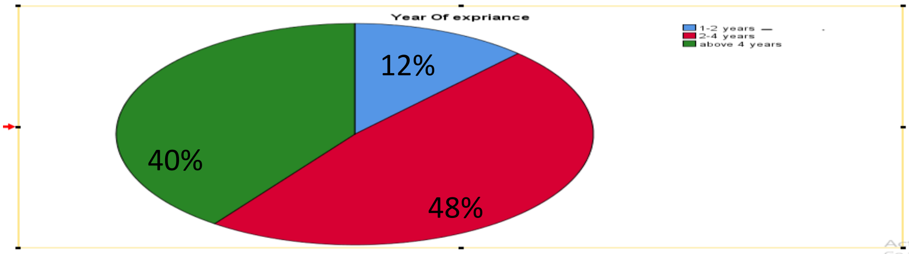
Formal education level of participant

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Degree	35	70.0	70.0	70.0
	Second degree or above	15	30.0	30.0	100.0
	Total	50	100.0	100.0	

3.1.2. *Experience of the participant*

The majority of the participant have experience between2-4 years which is 48% when it is set in percentage and twenty (20) participants have more than 4years experience (40%) and the rest 6 have 1-2 years' experience.

Fig 1: Year of experience



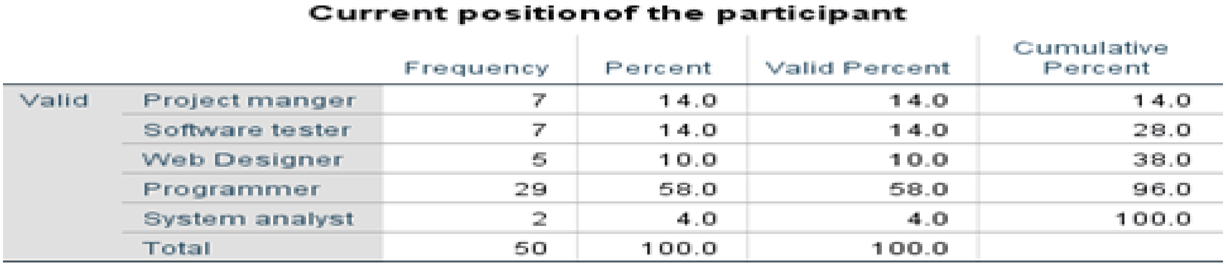
3.1.3. *Current position of the participant*

The majority of the participant worked as software programmers (58%) followed by 14% as software testers and project managers. That shows that the amount of testing is less. One ofthe respondents in an interview said that

"As a small company, it's hard to afford a tester, especially in security. Mostly we are focusedon the development part then we review the security in the process of the development andafter the code is finished".

Table 3

Positon of participant



Current positionof the participant

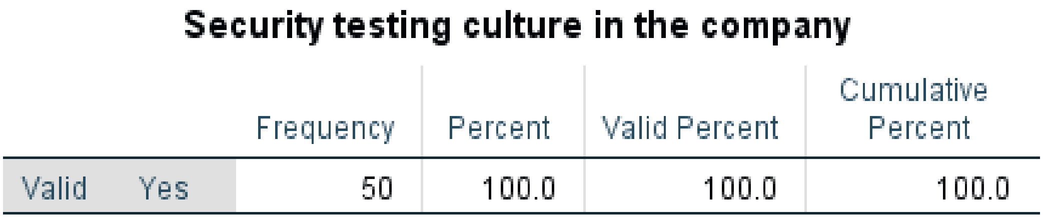
		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Project manger	7	14.0	14.0	14.0
	Software tester	7	14.0	14.0	28.0
	Web Designer	5	10.0	10.0	38.0
	Programmer	29	58.0	58.0	96.0
	System analyst	2	4.0	4.0	100.0
	Total	50	100.0	100.0	

3.2. Software Security Testing Effort in SDLCSoftware Security Testing culture

Almost all the software companies said they applied security testing in the process ofdeveloping a WBA.

Table 4

Software Security testing culture in the company



Security testing culture in the company

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Yes	50	100.0	100.0	100.0

3.3. Team structure in the company

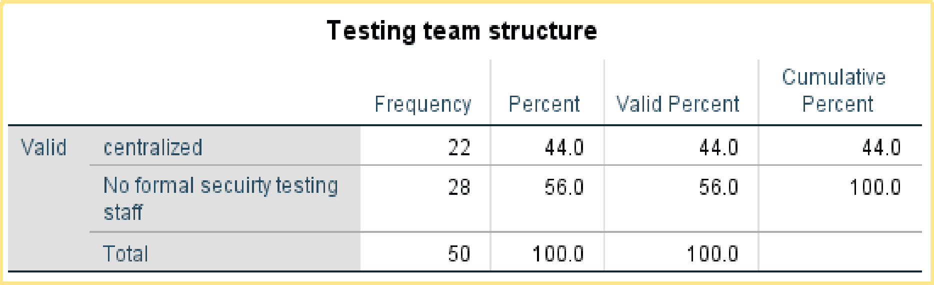
In the study the team structure in the software security testing process is divided into three: -

i. Centralized

ii. It is divided into the SDLC phase and the final is not formal.

Table 5

Software Security Testing Team structure



Testing team structure

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	centralized	22	44.0	44.0	44.0
	No formal secuirty testing staff	28	56.0	56.0	100.0
	Total	50	100.0	100.0	

One respondent said that "*Due to the amount of staff and the experience they have, it is not possible to divide staff to test in each SDLC phase, we mostly do not use any formal structure we just try to test after the code is finished and we use tools*"

3.4. Training in Software Security Testing

The participant was asked whether they get formal training in software security testing. The result indicates that 76% of the participant did not get any former training and 24% get formal training.

"Lack *of training in security testing is a major challenge in our company. there is no training that is found in the local area. Most of the trading is online and it costs in dollarcurrency due to that the company owner is not invested only in security testing but alsoin other training".*

3.5. Software security testing method

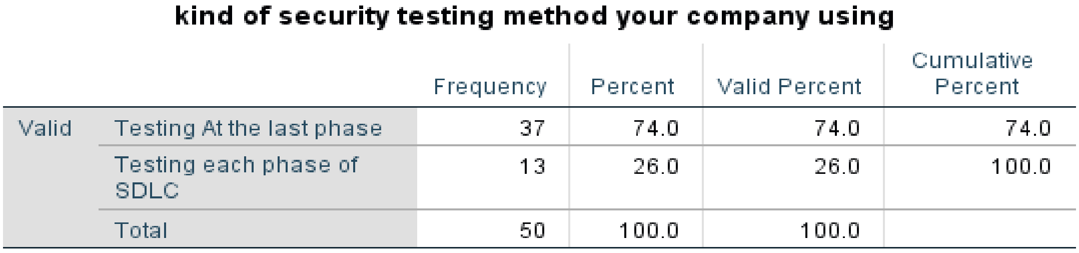
They are two types of security testing methods to develop secured software. In the study, most respondent tests their software at the last phase of the SDLC which is 37(74) %, and the other 13 percent test their software at each phase of SDLC.

"Most of the time we check the functionality of the software, and we try to develop a WBAwith the latest framework, so we don't face issues other than we try to secure the WBA using external security method".

"As a small company, it's hard to afford a tester, especially in security. Mostly we are focusedon the development part then we review the security in the process of the development andafter the code is finished".

Table 6

Software security testing method



kind of security testing method your company using

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Testing At the last phase	37	74.0	74.0	74.0
	Testing each phase of SDLC	13	26.0	26.0	100.0
	Total	50	100.0	100.0	

3.6. Software security tool

"As a team, we usually use open-source tool to test out the application due to the financial issue but we try to use different tool so we can get different view and result".

Table 7

Security testing tool

SST Tool	Always		Sometimes		Never	
	Freq.	%	Freq.	%	Freq.	%
Open source	46	92%	4	8%	0	0
Commercial	6	12%	17	34%	27	54%
Developed by the company	0	0	0	0	50	100%

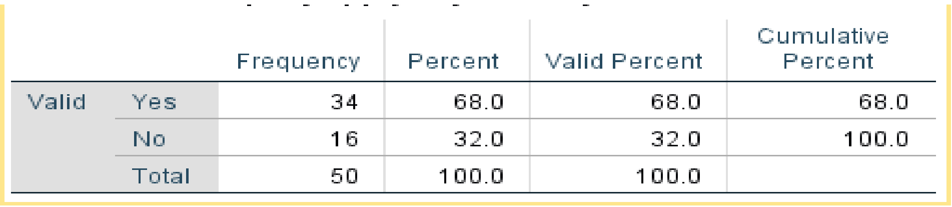
3.7. Security testing framework

Respondents were asked whether they the use security standard framework to follow the security protocol while developing a WBA.

"Most companies in Ethiopia either don't use any framework or they will not follow it strictly. Most of the framework is difficult to follow due to the complex documentation and there is no framework in Ethiopia that fit the software industry in Ethiopia."

Table 8

Participants that are used Framework



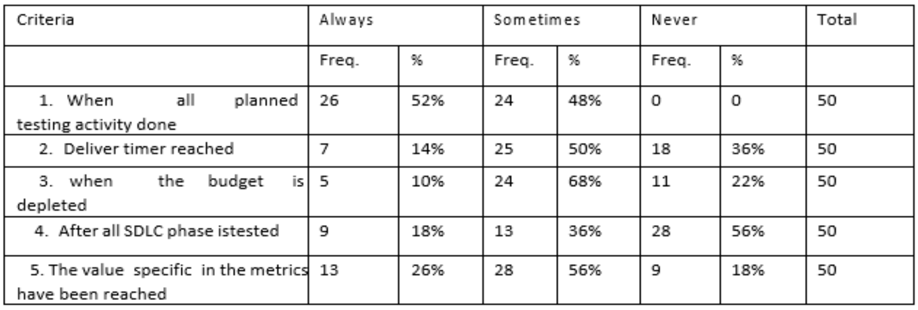
		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Yes	34	68.0	68.0	68.0
	No	16	32.0	32.0	100.0
	Total	50	100.0	100.0	

3.8. Criteria to Pass The WBA After Security Testing

Respondents were asked what kind of criteria should be full filled to pass the WBA from the security testing process and most of the participants pass the application when all the test plan activity is done which is 26(52%)

Table 9

criteria to pass the WBA after security testing



Criteria	Always		Sometimes		Never		Total
	Freq.	%	Freq.	%	Freq.	%	
1. When all planned testing activity done	26	52%	24	48%	0	0	50
2. Deliver timer reached	7	14%	25	50%	18	36%	50
3. when the budget is depleted	5	10%	24	68%	11	22%	50
4. After all SDLC phase istested	9	18%	13	36%	28	56%	50
5. The value specific in the metrics have been reached	13	26%	28	56%	9	18%	50

3.9. *Challenges in the Use of Methods*

The respondents were asked to rate the challenges that affect the adoption of softwaresecurity testing especially when it comes to testing each phase of the SDLC.

Table 10

Challenges in the Use of Methods

Challenges	S.agree		Agree		Neutral		Disagree		S.Disagree	
	Fre q.	%	Freq .	%	Freq.	%	Freq .	%	Freq.	%
1. Lack of experience	13	26 %	33	66%	4	85	0	0	0	0
2. Lack of budget	9	18 %	21	42%	12	24%	3	6%	5	10%
3. Time consuming	11	22%	19	38%	11	22%	4	8%	5	10%
4. Difficult to use	3	6%	12	24%	18	36%	17	34%	0	0
5. Complexity of the framework	2	4%	32	64%	12	24%	4	8%	0	0
6. Resource efficiency	14	28 %	22	44%	12	24%	2	4%	0	o
7. Lack of attention	6	12%	17	34%	18	36%	9	18%	0	0
8. Lack of training	17	34%	33	66%	0	0	0	0	0	0

3.10. *Summary of the Interview*

As it is stated in the research methodology to rich the goal of the qualitative research method there was an interview. The interview was designed to address the research objective. The resultof the interview will summarize as the following: -

1. Choosing SST method, tool, and framework: - One of the respondents said, “am not sure we choose the tools as our preference and interest”. Some of the companies did not even think about the necessity of SST. There is a lack of understanding of how to use the framework and how to choose a suitable framework for a WBA.

2. Challenges for not implementing security testing in each: - the majority of challenges the respondent included: -

- Lack of skilled/ experienced manpower
- Sufficient stuff
- Lack of time and budget
- Lack of testing infrastructures like a computer, internet connection, and workspace
- Lack of security testing guidelines
- Lack of training and awareness on testing methods and techniques and tool
- Information gap between staff members
- Poor documentation is the security testing process

- Software developer also works as a software tester
- Lack of motivation
- Lack of collaboration between staff members
- The staff members are unable to adapt and used the security framework

3. Solution: - the respondent proposes a solution to solve the challenges

- Give awareness about SST to the staff member
- Provide formal training for the staff member regarding software security testing methods, tools, and technique
- Structuring software security testing in the SDLC phase as an independent function
- Experience sharing in SST practice among companies among senior experts and company
- Practices software security framework and standards in the company
- The Government should prepare policies and standards to improve the quality of software companies.
- Proper documentation of security testing processes should include test cases and report
- Prepare Guidelines to apply security testing in every phase of SDLC
- Encouraging customers to participate in software security testing.

4. Discussion

4.1. Software Security testing Improvement Framework (SSTIF)

ISTF building first starts by identifying the major challenges and practices of software security testing in the Ethiopian software industry [17,18]. So, to mitigate the challenges and problems associated with software security testing to make the security testing process efficient and effective developing a framework/ guideline for a resource-constrained environment like the Ethiopian software industry.

4.2. Structure of the framework.

This framework is specifically developed based on the current status of the software security testing process in the Ethiopian software industry. This framework helps the developer team with the challenges they are facing while developing a WBA. This framework mainly focused on addressing the existing challenges that software company faces in software security testing [19-22]. Therefore, based on the existing situation of the software industry in Ethiopia and the requirements that are identified through mixed research methods (both quantitative and qualitative study) we proposed the Software Security Testing Improvement Framework (SSTIF).

The framework is structured in the software development life cycle (SDLC) phase and each phase has three major areas like management tasks, Security testing practices, and rules [23-26]. This major area is also categorized in sub-sections.

i. Management Task: - The major challenge we observe in this study is there is the lack of management. In some companies, the management is also involved in both the development and testing process and there is a lack of communication and coordination regards to security testing policy, strategy, plan, budgeting, and staff management.

ii. Security testing practice: - this category has three sub-categories

- Security task: - the challenges associated with proper security adaption in each phase of SDLC. This security task helps the company what kind of task it should be done in each phase.
- Technique: - the challenges associated with choosing the proper security testing technique in the SDLC phase (white box, black box, gray box)
- Tools: - challenges associated with different software security tools (open source or commercial).

iii. Rules: - challenges associated with software security testing rules that should be done/cooperate in SDLC and between the team members.

iv. Phases: - There are five phases in the SDLC:

- Phase 1 Initiation / Requirement
- Phase 2 Design
- Phase 3 Implementation
- Phase 4 Tasting
- Phase 5 Deployment.

Table 11

Software Security testing improvement Framework (SSTIF) workflow

Phase 1	Assign Role and responsibility Create Awareness Set internationalstandard	Stakeholder identification Prepare a securityrequirement checklist Security requirementanalysis Security risk assessment	securitystandard Threat identification	Manually	Remove ambiguity Prevention
Phase 2	Prepare security principle guideline	Prepare an architectural risk assessment checklist Architecture riskanalysis Design review	Externalreview	Manuallyreview	Documenteviewed Avoid Vulnerability.
Phase 3	Providetraining Pair programmer Set development standard	Static code analysis Prepare code Standard checklist development Assess the developmentstandards	Code review White box technique	Automated scanner	Flexibility Language
Phase 4	Testing plan	Security testingmetrics Testing review Analysis tools	White box Blackbox	Vulnerability assessment Static analysistools	Acceptance Report
Phase 5	Infrastructure Test monitoring andcontrol	Pre-implementation risk assessment Dynamic analysis ATM(applicationinfrastructure management)	Blackbox Gray box	Penetration testing Vulnerability assessment	Security assessment Strong securitytesting
SDLC phases	Managerial task	Task	Technique	Tools	Rules
		Security Testing Practice			

5. Conclusion

The study was done only in Addis Ababa, Ethiopia, within limited companies and participant further research should be conducted to investigate the practices and challengesof Software security testing in the software companies in Ethiopia. And software security testing always needs improvement and updates so that future researchers can improve the functionality of the framework by categorizing different tasks and techniques.

The contribution of this study is to mitigate the challenges the developer has when they perform security testing in a WBA. This study also contributes to visible data about security testing performance and processes in Addis Ababa, Ethiopia. Then finally an integrated software security testing improvement framework was introduced to the company to guide the developer and test to perform security testing in each phase of the software development life cycle.

REFERENCES

[1] N. Teodoro and C. Serrao, “Web application security: Improving critical web-based applications quality through in-depth security analysis,” in 2011 International Conference on Information Society (i-Society), 2011, pp. 457-462.

[2] Nosheen Nazira*, Muhammad Kashif Nazirb “Security Issues in SDLC” 2018.

[3] N. Haridas, "Software Engineering-Security as a Process in the SDLC," SANS Institute, p. 29, 2007.

[4] G. McGraw, "Security Penetration Testing" PUBLISHED BY THE IEEE COMPUTER SOCIETY" IEEE SECURITY & PRIVACY ■ JANUARY/FEBRUARY 2005.

[5] E. İ. Tatlı "Developer-oriented Web Security by Integrating Secure SDLC into IDEs" April 2018

[6] A. Fathi, A. Sawehli "Improving Software Security Testing of Software Development Life Cycle (SDLC) For Web-Based Applications by Providing a Quality System (Web-Vs) for Vulnerability Assessment," 2019.

[7] A. Hudaib, M. AlShraideh, "A Survey on Design Methods for Secure Software Development," 2017.

[8] T. Yuan-Hsin "An integrated security testing framework for Secure Software Development Life Cycle," 2016.

[9] M. Juan de Vicente "The Application of a New Secure Software Development Life Cycle (S-SDLC) with Agile Methodologies" Security Testing: A Survey, 2019.

[10] M. Deylami, H., Ardekani, "Effects of software security on software development life cycle and related security issues," 2015.

[11] M. Felderer, B. Matthias, J. Martin, D. B. Achim, R. Breu, A. Pretschner, "Security testing a survey" 2016.

[12] T. Yuan-Hsin "An integrated security testing framework for Secure Software Development Life Cycle," 2016.

[13] P. Zech, M. Felderer, R. Breu "Knowledge-based security testing of web applications by logic programming" The Author(s), 2017.

[14] S. Tamiru. (2017). An Integrated Software Test Process Framework: The Case of Selected Ethiopian Software Companies (Doctoral dissertation, St. Mary's University).

[15] H. G. Debebe, "Security Testing of Ethiopian E-Governmental Websites Using Penetration Testing Tools. Master's Thesis". Near East University. Turkish Republic of Northern Cyprus. 2019.

[16] www.insa.gov.et

[17] A. Mamdouh, and S. Almuairfi. "Security risks in the software development lifecycle." International Journal of Recent Technology and Engineering 8.3: 7048- 7055, 2019.

[18] A. M. Ibukunoluwa, "Security testing challenges of web developers in the Lagos, Nigeria IT industry" Diss. 2020.

[19] D. R. Revo, G. M. A. Sasmita, and I. Putu Agus Eka Pratama. "The Testing for Information Gathering Using OWASP Testing Guide V4 (Case Study: Udayana University SIMAK-NG Application)." Jurnal Ilmiah Teknologi dan Komputer 1.1: 23-33.

[20] S.Horton, Are Software Security Issues a Result of Flaws in Software Development Methodologies? Diss. Utica College, 2020.

[21] L. A. Davaindran, et al. "Implementation of Security Features in Software Development Phases." arXiv preprint arXiv:2012.13108, 2020.

[22] R. Maxwell, H. Tzu-Tang, and L. A. Md. "Security Considerations for the Development of Secure Software Systems." 2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON). IEEE, 2019.

[23] G. Gurung, R. Shah, D. P. Jaiswal "Software Development Life Cycle Models-A Comparative Study" 2020 International Journal of Scientific Research in Computer Science, Engineering and Information Technology.

[24] V. C. Wilfred, "The Waterfall Model and the Agile Methodologies: A comparison by project characteristics." Research Gate 2 (2017): 1-6.

[25] C. D. Soares, et al. "How is security testing done in agile teams? a cross-case analysis of four software teams." International Conference on Agile Software Development. Springer, Cham, 2017.

[26] R. Marco, and M. Prandini. "An integrated application of security testing methodologies to e-voting systems." International Conference on Electronic Participation. Springer, Berlin, Heidelberg, 2010.