JCBD
JOURNAL OF COMPUTERS AND DIGITAL BUSINESS
Volume 4, Issue 1, January 2025, pp. 24-36
Homepage: https://jurnal.delitekno.co.id/index.php/jcbd

Investigating the Level of Experience in Using More Effective Software Design Tools for Enhancing Security among Federal College of Education, Gidan Madi Staff and Students

Hajara Abdulkadir[1*], Bello Alhaji Buhari[2], Abdulrahman Umar[3]

[1] Department of Computer Science, Federal College of Education Gidan Madi, Sokoto, Nigeria
[2] Department of Computer Science, Usmanu Danfodiyo University, Sokoto, Nigeria
[3] Department of Chemistry, Federal College of Education Gidan Madi, Sokoto, Nigeria

*buhari.bello@udusok.edu.ng

https://doi.org/10.56427/jcbd.v4i1.629

ARTICLE INFO

Article History
Received: 9 December 2024
Revised: 15 January 2025
Accepted: 17 January 2025

Keywords
Software Design Tools
Security Enhancement
Data Privacy
Security
Gidan Madi

ABSTRACT

Higher education institutions operate under strict regulations and standards to ensure compliance with data protection rules, safeguard sensitive information, and preserve the privacy of employees and students. This study evaluates the expertise of staff and students at the Federal College of Education, Gidan Madi, in employing software design tools to enhance security. Using a mixed-methods approach, the research involves a wide range of stakeholders, including academics, non-academics, and students. The findings reveal that software design tools significantly improve the institution's ability to detect and respond to security events. Participants highlighted the importance of data encryption, expressed confidence in their knowledge of the latest advancements in security tools, and rated institutional support for security measures as excellent. The study also identified gaps in network monitoring capabilities, which are critical for proactive security management. Based on these findings, the study recommends the implementation of advanced network monitoring tools to enhance security measures at the institution. These results are expected to benefit higher education stakeholders, IT administrators, and decision-makers responsible for designing and maintaining robust security systems. By addressing current challenges and improving expertise in using security tools, this research contributes to the broader goal of strengthening data protection in higher education institutions.

1. Introduction

Vyas in [1] presented a summary of the main security issues that Java developers encounter and possible remedies to reduce the risks. Security Issues When Developing Java Applications Vulnerabilities in Code: Code vulnerabilities such as SQL injection, cross-site scripting (XSS), and remote code execution can affect Java programs. These flaws may result in application crashes, illegal access, and data leaks. Java programs are an example of insecure dependencies since they frequently use third-party frameworks and libraries. To improve security, make use of well-established identity management systems. Strong encryption techniques are used in data encryption to safeguard data both in transit and at rest.Use methods such as HTTPS when communicating online and encryption libraries for storing data. Observation and Reaction to Incidents: Use reliable monitoring tools to find irregularities and security breaches. Create an incident response strategy so that you can respond quickly to security breaches.

An empirical analysis of 1,396 vulnerability reports impacting 698 Python packages in the Python ecosystem (PyPi) was given by Alfadel et al. in [2]. They specifically research the spread and lifespan of security flaws, taking into consideration the time it takes for a vulnerability to be found and patched.

Furthermore, software projects that rely on packages (dependent projects) may also be vulnerable due to vulnerabilities in those packages. To gain more insight into the frequency of vulnerabilities in their dependencies and the speed at which they can be updated, they examine a collection of 2,224 Python projects on GitHub. According to their research, vulnerabilities in Python packages are becoming more common over time and take longer than three years to identify. Forty-six percent of these vulnerabilities are only addressed after being made public, which allows attackers plenty of time to exploit them.

Pearce et al. in [3] investigated the application of large language models (LLMs) for code for zero-shot vulnerability repair, such as AI21's Jurassic J-1 and OpenAI's Codex. They look at issues with prompt design that induce LLMs to produce patched versions of vulnerable code. This is challenging since there are a lot of ways to frame important information in natural languages in both semantic and syntactic ways. On a variety of artificial, manually created, and real-world security bug scenarios, they conduct a thorough analysis of five commercially available, blackbox, "off-the-shelf" LLMs in addition to an open-source model and our own locally trained model. Although their experiments show promise (all together, the LLMs could fix 100% of our manually created and artificially generated scenarios), a qualitative assessment of the model's performance across a corpus of historical real-world examples reveals difficulties in producing functionally correct code.

Votipka et al. in [4] looked into how and why programmers make security-relevant mistakes even when they have a foundational understanding of security. They achieved this by thoroughly examining 94 entries to a safe programming competition that simulated three real-world constraints: security, performance, and accuracy. Participants were requested to write secure code and also look for weaknesses in the programs of other teams; in all, teams presented 866 exploits against the submissions that were taken into consideration. We employed iterative open coding over a rigorous six-month period to manually but methodically define each submitted project and vulnerability. They categorized vulnerabilities based on the nature, simplicity of exploitation, and attacker control that was permitted, as well as projects based on security implementation strategies. A few trends showed up. For instance, basic errors were the least frequent, occurring in just 21% of projects. On the other hand, vulnerabilities resulting from a misinterpretation of security concepts were much more prevalent, showing up in 78% of projects. Their findings have implications for bettering vulnerability-finding tools, secure-programming APIs, documentation, and security awareness.

A revolutionary Security-by-Design methodology based on Security Service Level Agreements (SLAs) was proposed by Casola et al. in [5]. It can be integrated into contemporary development processes and supports the risk management life-cycle in an almost entirely automated manner. It specifically depends on a fully automated security assessment phase and a guided risk analysis procedure, which allow it to evaluate the security attributes provided by a cloud service and report on them in a Security SLA. They verified the suggested methodology using an actual case study, which demonstrated its efficacy in shortening the secure design process time and raising the level of security awareness within the development and design teams.

In 2020, Silic and Lowry [6] carried out a design-science research study with the aim of enhancing an organization's internal security training program and staff phishing prevention efforts, which were shown to be ineffective. In order to do this, they developed a gamified security training system with a dual focus on gamification's ability to increase intrinsic motivation and security learning's effectiveness. Their most important theoretical contribution is the recontextualized kernel theory they offer from the adoption model of the hedonic-motivation system, which can be utilized to evaluate employee security constructs in addition to their intrinsic incentives and coping mechanisms for compliance and learning. A six-month field study involving four hundred participants demonstrates that gamified security training can lead to statistically significant positive behavioral changes by satisfying users' coping needs and motivations. In this context, they also offer a fresh empirical evidence of the conceptual significance of "appropriate challenge." Our work is scrutinized based on the concepts of proof-of-concept and proof-of-value, and we culminate with a research plan that guides us towards the ultimate proof-in-use.

Fannoun and Kerins in [7] provided an assessment of practice within a software development unit that is in the process of developing. The purpose of the assessment was to determine what lessons were learned and how those lessons were implemented to improve organizational development. Their research provides the basis for developing and implementing a recommended support tool that would enhance professional practice.

Another study was conducted by [8 - 9]. These tools, which concentrate on assessing static code analysis tools, assist developers in identifying risky coding habits and enforcing secure coding practices as preventative measures. Potential hazards and flaws in software systems can be found by applying a threat modeling technique. These tools let developers identify potential attack vectors, select security countermeasures, and examine the system architecture

2. Methodology

The research methodology serves as the study's compass by offering an organized framework for methodically addressing the research objectives. With the aim of determining the degree of expertise in sing-specific software design tools that are more successful for boosting security among Federal College of Education Gidan Madi personnel and students, this technique is meant to provide a nuanced and thorough analysis. This approach aims to ensure that the investigation is carried out honestly and precisely, that it follows the study's goals, and that it produces well-informed findings. More information on the public's experience with particular software products that effectively improve security will be available to us. We explore the complex domain of this experience in academic contexts using a variety of methodologies in an effort to understand it more deeply than is evident.

a. Research Design

A mixed-methods approach was selected as the research technique for this study because to its ability to produce a thorough and multifaceted grasp of the topic themes. To investigate the various facets of software design tools' contribution to improving security measures in higher education, both quantitative and qualitative research approaches must be used.

Structured surveys form the basis of the study design's quantitative section. A representative and varied sample of Federal College of Education Gidan Madi participants will receive these surveys. The study's quantitative component is to determine how users feel about using particular software design tools that improve security more successfully. We can find statistical links, patterns, and trends that are relevant to the study's objectives by gathering data through surveys.

Conversely, the qualitative component explores user experiences and institutional activities in great detail. This format consists of semi-structured interviews and case studies carried out at the chosen institution. These qualitative approaches aim to offer compelling stories, deep insights, and rich contextual data. Participants will be able to discuss their thoughts, difficulties, and experiences using software design tools on an interview platform.

b. Population of the Study

Staff and students of Federal College of Education, Gidan Madi, the main case study institution, serve as a model for the research's population. This group is made up of a range of stakeholders who engage with certain software design tools and are essential to the study's emphasis, including students, instructors, administrative staff, and IT people. Surrounded by various cities and villages, the Federal College of Education, GidanMadi, is situated in an essentially rural area. Still, the internal community of the institution was the main focus of this study during its initial scope.

c. Sample and Sampling Technique

One hundred people were selected from the Federal College of Education, GidanMadi neighborhood to make up the study's sample. This group of people consists of academic staff, IT staff, administrative workers, and students. Since the institution serves as the key case study, the study will concentrate on a representative and diverse sample of this community to find out how people view software design tools and how they could be applied to strengthen security measures.

Using stratified random sampling, the quantitative portion of the study—which will primarily consist of surveys—will be conducted. According to the various stakeholder groups (administrative staff, instructors, students, and IT personnel), this strategy stratifies the population. A random sample will be chosen from each stratum. In addition to ensuring that survey responses are representative of the various demographic groupings, this makes it possible to draw insightful comparisons between various subgroups.

Purposeful sampling will be used in the qualitative section of the study, which consists of case studies and interviews. Intentional sampling requires careful consideration of the study's goals when choosing participants. Here, the participants will be selected based on their specialized roles and backgrounds in information security and software design tools. This strategy makes sure that the viewpoints of important stakeholders are thoroughly taken into account in the qualitative section.

In the setting of Federal College of Education, GidanMadi, this study seeks to obtain a thorough understanding of the population's experiences, views, and practices regarding information security and particular software design tools. To do this, it combines purposive sampling for the qualitative component and stratified random sample for the quantitative component. These sampling techniques have been selected to maintain the representativeness of the sample and allow for a detailed analysis of the experiences of those with specific expertise and responsibilities in this area.

d. Instrument for Data Collection

A structured questionnaire is the main tool used in this study to gather quantitative data. The purpose of the questionnaire is to gather user opinions, behaviors, and first hand information about the experience with

particular software design tools and their efficacy on security protocols within the chosen institution. The questionnaire's blend of closed-ended and Likert-scale questions allows respondents to express their ideas and experiences in a consistent way. The chosen sample of participants will receive the questionnaire, and various parts of the research objectives will be quantified based on their responses.

There will be semi-structured interviews with important parties. These stakeholders include academic, non academic and students with expertise in information security and software design tools. Open-ended questions are included in the interview guide to enable participants to express their ideas, opinions, and experiences. The rich qualitative information gathered from these interviews will enable a more thorough examination of the study questions.

Case study approaches will be used in the case study component. The methods and steps for carrying out in-depth case studies at the chosen institution are described in these protocols. The institutional setting, the usage of particular software design tools, and information security measures are all extensively covered by the standards that control data collecting. Case studies provide a thorough perspective on the particular situations in which these approaches are applied as well as an in-depth analysis of the study objectives.

e. Procedure for Data Collection

A systematic questionnaire has been designed to gather user perspectives and experiences about software design tools and their influence on security measures. Using stratified random selection techniques, the Federal College of Education, GidanMadi, provided the participants for the sample. Informed permission is obtained from participants after they have been informed about the aim of the study and their rights as participants.

Semi-structured interview guides and case study methods are developed to make data collection easier. Key stakeholders—faculty, administrative staff, and IT specialists knowledgeable about software design tools and information security—are selected for interviews and case studies using purposive sampling techniques.

f. Data Analysis

To allow for flexible and simple result interpretation, each questionnaire question was expressed as a table. The table contains five useful sections: Cumulative percent, valid percent, Frequency, Percent, and a bar chart to help visualize the data that was gathered.

3. Results

All questionnaire questions were expressed as tables to facilitate easy and flexible interpretation of the results. Five helpful elements make up the table: Cumulative percent, valid percent, Frequency, Percent, and a bar chart to aid in the visualization of the collected data.Q1: Sex Categories

Table 1: Sex Categories

	Frequency	Percent	Valid Percent	Cumulative Percent
Male	60	60.0	60.0	60.0
Female	40	40.0	40.0	100.0
Total	100	100.0	100.0	

Source: Questionnaire Administered (2023)

Figure 1 of the preceding table shows that 40% of respondents were women and 60% of respondents were men. Figure 1 of the preceding table shows that 40% of respondents were women and 60% of respondents were men.

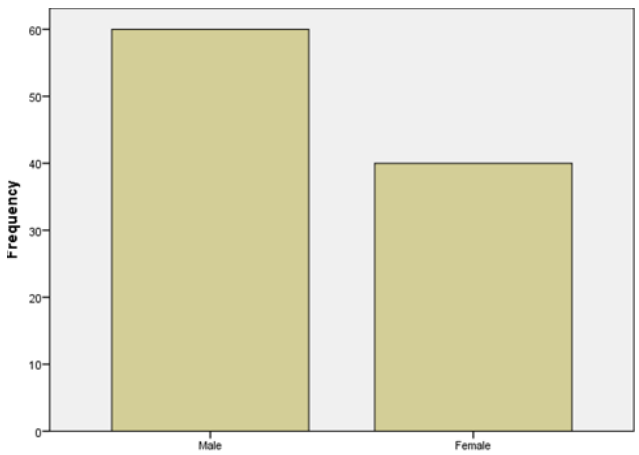

Figure 1: Age

Q2: Years

Table 2: Years

	Frequency	Percent	Valid Percent	Cumulative Percent
18-30 Years	10	10.0	10.0	10.0
31-40 Years	40	40.0	40.0	50.0
41-50 Years	35	35.0	35.0	85.0
51 Years and Above	15	15.0	15.0	100.0
Total	100	100.0	100.0	

Source: Questionnaire Administered (2023)

As shown in figure 2, 9.8% of the respondents were in the 18–30 age range, 39.2% were in the 31–40 age range, 8% were in the 41–50 age range, and 4% were above 50.

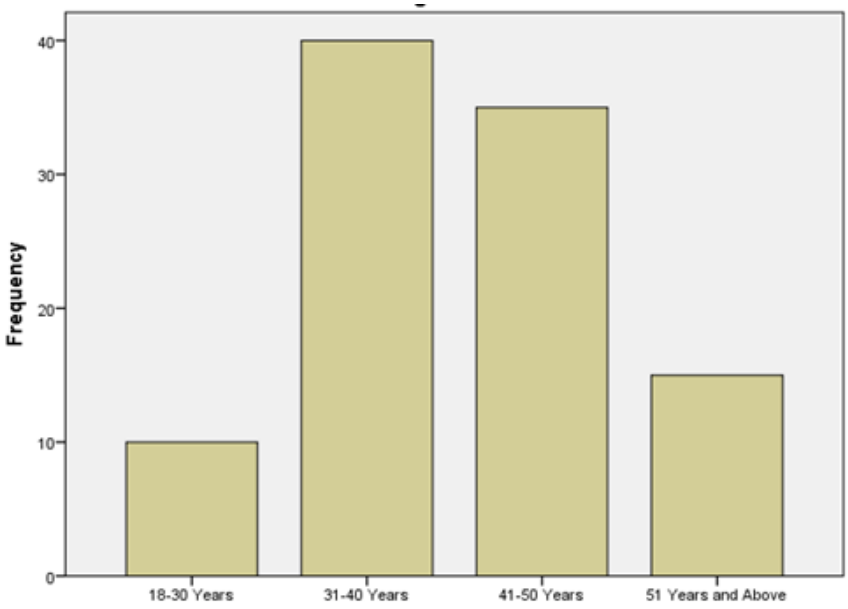

Figure 2: Year

Table 3: Marital Status

	Frequency	Percent	Valid Percent	Cumulative Percent
	20	20.0	20.0	20.0
	75	75.0	75.0	95.0
	5	5.0	5.0	100.0
	100	100.0	100.0	

Source: Questionnaire Administered (2023)

The accompanying table makes it clear that 25% of respondents were married, 20% were single, and 5% were divorced. Figure 3 provides an illustration of this.

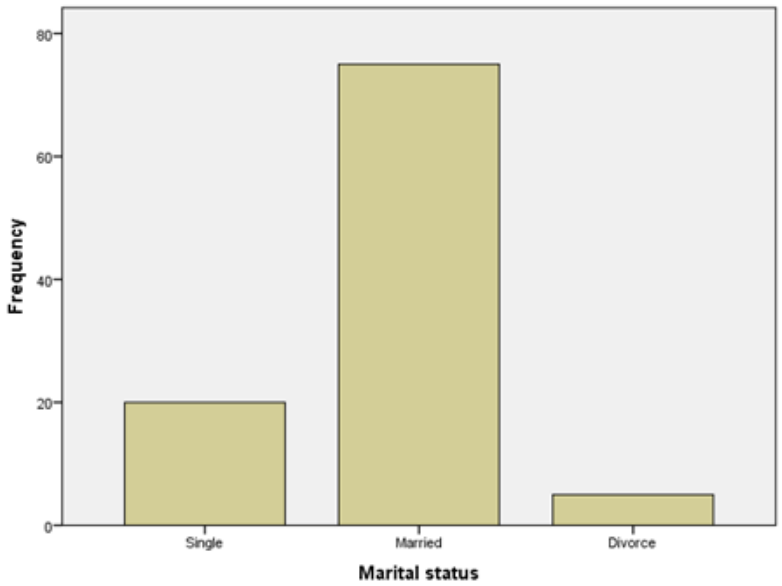

Figure 3: Marital Status

Q4: What is your primary role within Federal College of Education, Gidan Madi?

Table 4: What is your primary role within Federal College of Education, Gidan Madi?

	Frequency	Percent	Valid Percent	Cumulative Percent
Student	**20**	**20.0**	**20.0**	**20.0**
Faculty Member	**29**	**29.0**	**29.0**	**49.0**
Administrative Staff	**20**	**20.0**	**20.0**	**69.0**
IT Personnel	**31**	**31.0**	**31.0**	**100.0**
Total	**100**	**100.0**	**100.0**	

Source: Questionnaire Administered (2023)

At Federal College of Education, Gidan Madi, the following roles are assigned in accordance with the data presented: Twenty percent of the sample consists of students, faculty members, twenty nine percent, administrative staff twenty percent, and IT professionals thirty one percent. This can be shown in Figure 4.

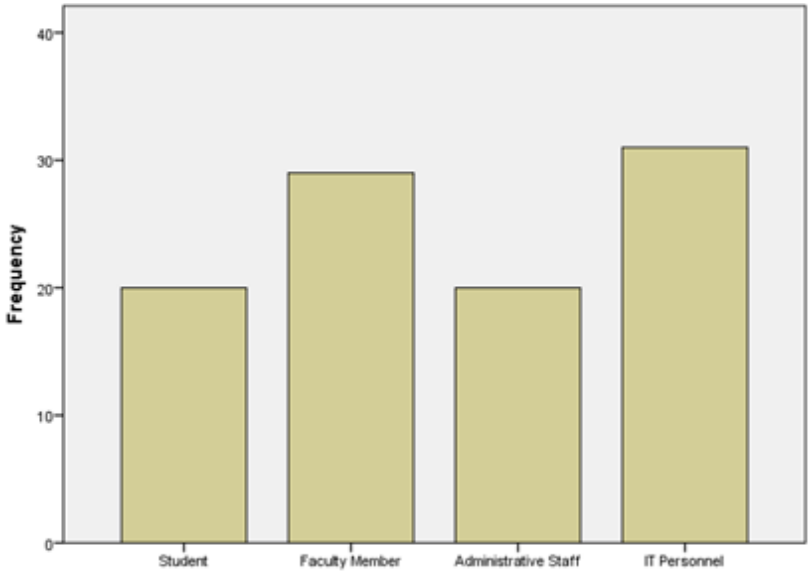

Figure 4: What is your primary role within Federal College of Education, Gidan Madi?

Table 5: In your experience, which specific software design tools do you believe is more effective in enhancing security?

	Frequency	Percent	Valid Percent	Cumulative Percent
UML (Unified Modeling Language)	**40**	**40.0**	**40.0**	**40.0**
Balsamiq	**12**	**12.0**	**12.0**	**52.0**
Sketch	**30**	**30.0**	**30.0**	**82.0**
Axure RP	**18**	**18.0**	**18.0**	**100.0**
Total	**100**	**100.0**	**100.0**	

Source: Questionnaire Administered (2023)

The participants had differing opinions about how well some software design tools work to improve security based on their own experiences. Of particular note, 40% of respondents preferred UML (Unified Modelling Language), 12% highlighted Balsamiq, 30% said Sketch was useful, and 18% mentioned Axure RP as being worthwhile. The fifth figure illustrates this.

Figure 5: In your experience, which specific software design tools do you believe is more effective in enhancing security?

Table 6 How do you perceive the impact of using software design tools on the institution's ability to detect and respond to security incidents?

	Frequency	Percent	Valid Percent	Cumulative Percent
Significantly improves detection and response	**65**	**65.0**	**65.0**	**65.0**
Moderately improves detection and response	**20**	**20.0**	**20.0**	**85.0**
Has a minimal impact on detection and response	**15**	**15.0**	**15.0**	**100.0**
Total	**100**	**100.0**	**100.0**	

Source: Questionnaire Administered (2023)

In evaluating the influence of software design tools on the institution's capability to detect and respond to security incidents, respondents indicated varied perceptions. A majority (65%) believe it significantly improves detection and response, while 20% see a moderate improvement, and 15% observe a minimal impact. This can be shown in figure 6.

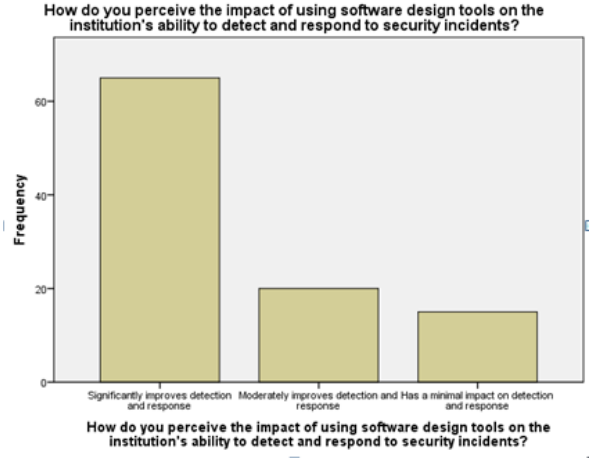

Figure 6: How do you perceive the impact of using software design tools on the institution's ability to detect and respond to security incidents?

Table 7: Are there specific security features or practices that you believe should be prioritized when using software design tools in educational institutions?

	Frequency	Percent	Valid Percent	Cumulative Percent
Data encryption	**41**	**41.0**	**41.0**	**41.0**
Access control and authentication	**19**	**19.0**	**19.0**	**60.0**
Regular security audits	**22**	**22.0**	**22.0**	**82.0**
Security training for users	**18**	**18.0**	**18.0**	**100.0**
Total	**100**	**100.0**	**100.0**	

Source: Questionnaire Administered (2023)

When considering the utilization of software design tools in educational institutions, respondents highlight specific security features or practices that merit prioritization. Notably, 41% emphasize the significance of data encryption, while 19% stress the importance of access control and authentication. Additionally, 22% recommend regular security audits, and 18% underscore the need for security training for users. This is shown in figure 7.

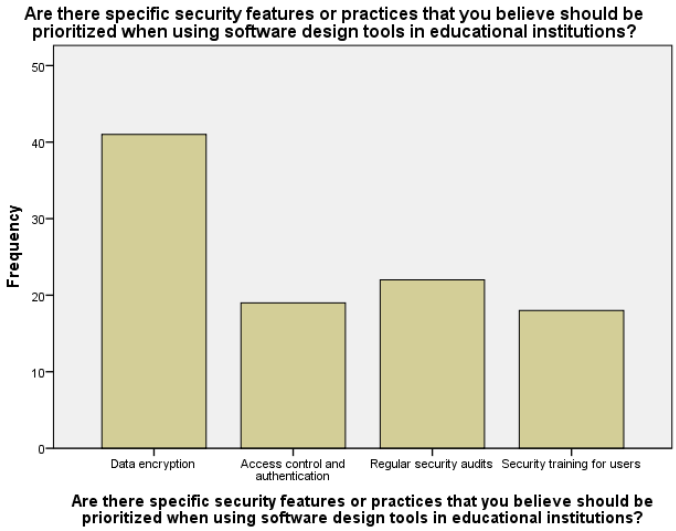

Figure 7: Are there specific security features or practices that you believe should be prioritized when using software design tools in educational institutions?

Table 8: How well-informed do you feel about the latest developments in software design tools for security in higher institutions?

	Frequency	Percent	Valid Percent	Cumulative Percent
Very well-informed	**38**	**38.0**	**38.0**	**38.0**
Well-informed	**22**	**22.0**	**22.0**	**60.0**
Moderately informed	**10**	**10.0**	**10.0**	**70.0**
Slightly informed	**5**	**5.0**	**5.0**	**75.0**
Not informed	**25**	**25.0**	**25.0**	**100.0**
Total	**100**	**100.0**	**100.0**	

Source: Questionnaire Administered (2023)

Respondents' levels of awareness about the latest developments in software design tools for security in higher institutions vary. A notable 38% feel very well-informed, 22% consider themselves well-informed, 10% express moderate awareness, 5% feel slightly informed, and 25% admit to not being informed. This is shown in figure 8.

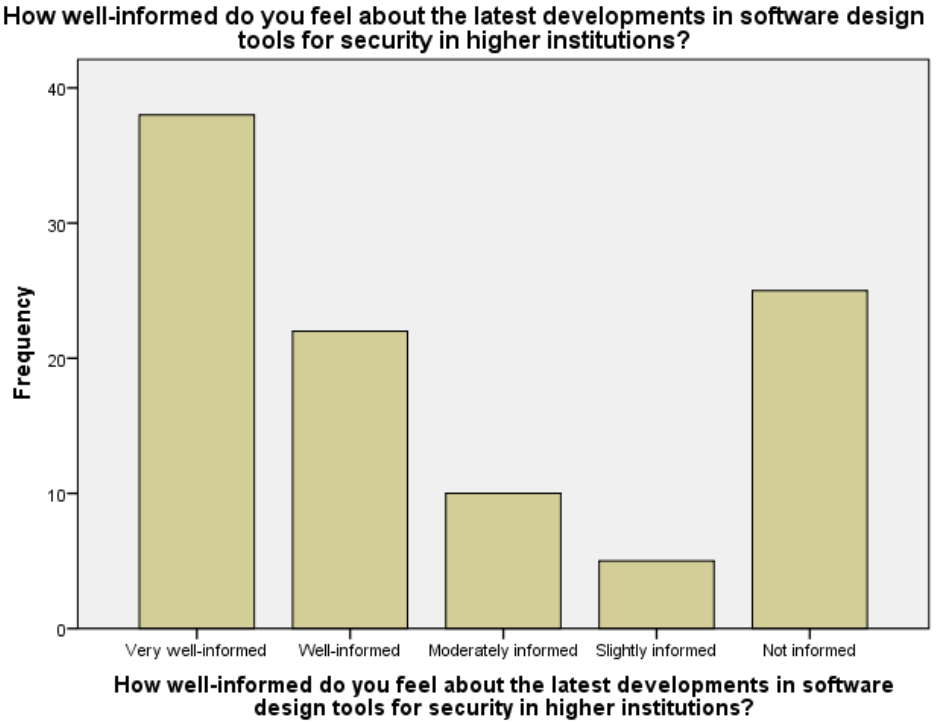

Figure 8: How well-informed do you feel about the latest developments in software design tools for security in higher institutions?

Table 9: How do you perceive the level of support and resources provided by your institution to enhance security measures?

	Frequency	Percent	Valid Percent	Cumulative Percent
Excellent	**65**	**65.0**	**65.0**	**65.0**
Good	**22**	**22.0**	**22.0**	**87.0**
Fair	**10**	**10.0**	**10.0**	**97.0**
Poor	**3**	**3.0**	**3.0**	**100.0**
Total	**100**	**100.0**	**100.0**	

Source: Questionnaire Administered (2023)

Respondents' perceptions of the level of support and resources provided by their institution to enhance security measures vary. The majority, 65%, view the support as excellent, 22% rate it as good, 10% find it fair, and a small percentage, 3%, consider it poor. This can be shown in figure 19.

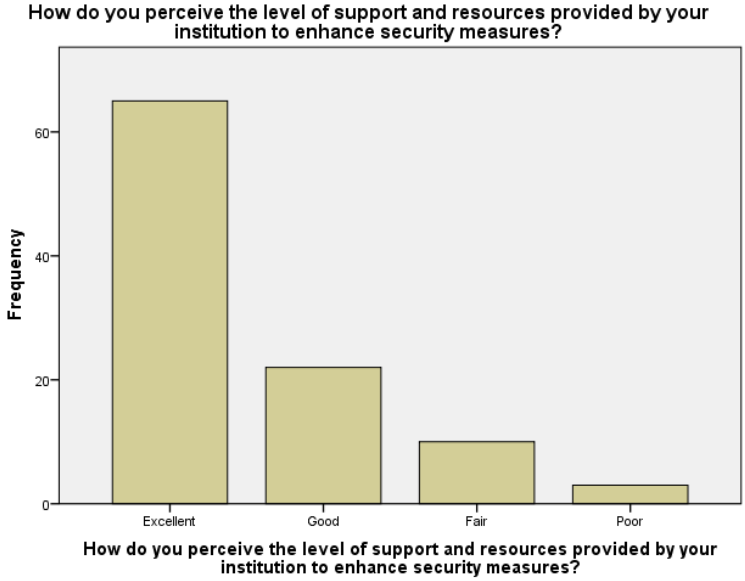

Figure 9: How do you perceive the level of support and resources provided by your institution to enhance security measures?

Table 10: Do you believe that your institution adequately addresses the security needs of all stakeholders, including students, faculty, and administrative staff?

	Frequency	Percent	Valid Percent	Cumulative Percent
Yes	**96**	**96.0**	**96.0**	**96.0**
No	**4**	**4.0**	**4.0**	**100.0**
Total	**100**	**100.0**	**100.0**	

Source: Questionnaire Administered (2023)

Respondents overwhelmingly express the belief that their institution adequately addresses the security needs of all stakeholders, encompassing students, faculty, and administrative staff, with 96% affirming this perspective. Only a small minority of 4% indicates a contrary opinion. This can be shown in figure 10.

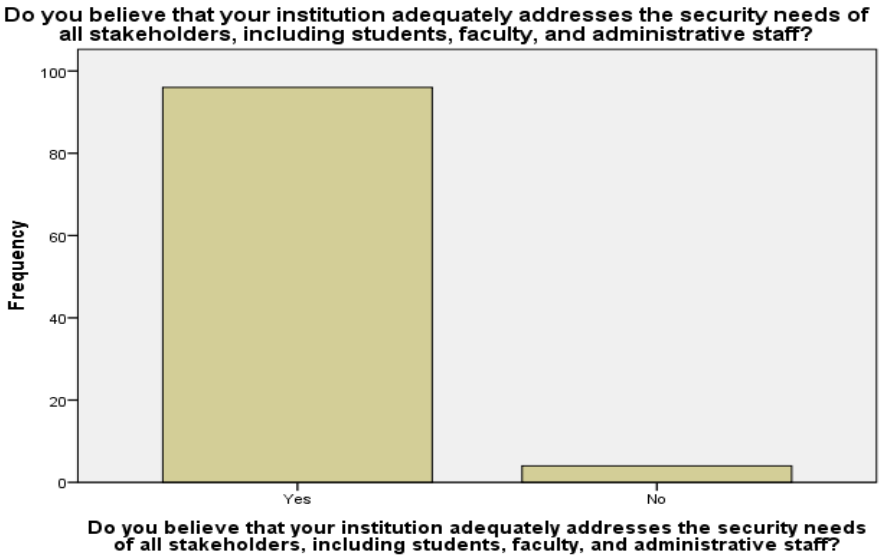

Figure 10: Do you believe that your institution adequately addresses the security needs of all stakeholders, including students, faculty, and administrative staff?

Table 11: Are there any specific software design tools or practices that you believe should be implemented at Federal College of Education, Gidan Madi, to enhance security? (Select all that apply)

	Frequency	Percent	Valid Percent	Cumulative Percent
Improved network monitoring tools	35	35.0	35.0	35.0
Enhanced access control measures	29	29.0	29.0	64.0
Regular security awareness training	10	10.0	10.0	74.0
More robust data encryption	26	26.0	26.0	100.0
Total	100	100.0	100.0	

Source: Questionnaire Administered (2023)

Respondents suggest various measures for enhancing security at Federal College of Education, Gidan Madi. Notably, 35% recommend the implementation of improved network monitoring tools, 29% advocate for enhanced access control measures, 10% propose regular security awareness training, and 26% emphasize the importance of more robust data encryption. This can be shown in figure 11.

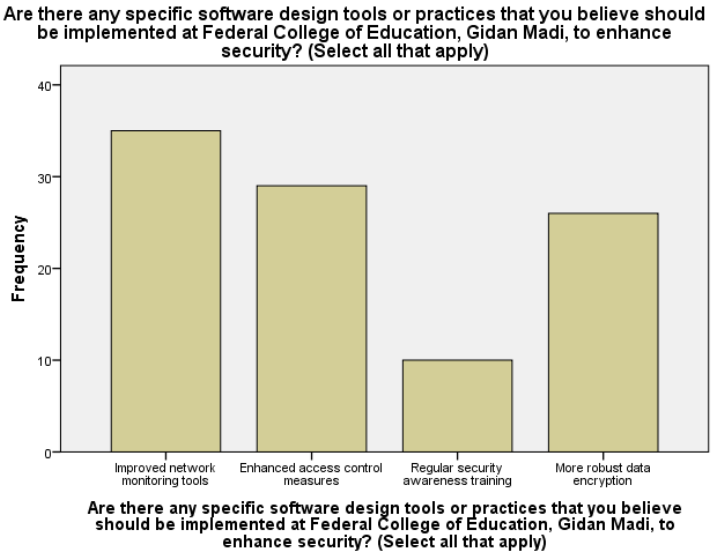

Figure 11: Are there any specific software design tools or practices that you believe should be implemented at Federal College of Education, Gidan Madi, to enhance security?

4. Discussion

We surveyed the Federal College of Education Gidan Madi staff and students to see out how much they understood about using software design tools. Table 1 showed that men made up about 60% of the participants, indicating a significant participation rate from women as well. Table 2 shows that the majority of participants are between the ages of 31 and 40. The majority of participants are married, according to the statistics in Table 3. Table 4 shows that the majority of participants are IT professionals.

Additionally, Table 5 demonstrates that most participants (40%) preferred UML (Unified Modelling Language) to express their differing opinions about how well-defined certain software design tools are at improving security; Table 6 indicates that most participants (65%) think it greatly enhances detection and response when assessing the impact of software design tools on the institution's capacity to identify and address security incidents; Table 7 indicates that most participants (41%) highlight the importance of data encryption; and Table 8 indicates that most participants (38%) feel extremely well-informed about the most recent advancements in software design tools for security in higher education.

Lastly, table 9 shown that majority of participants (65%) view the level of support and resources provided by their institution to enhance security measures as excellent, table 10 shown that majority of the participants (96%) express the belief that their institution adequately addresses the security needs of all stakeholders,

encompassing students, faculty, and administrative staff and table 11 shown that majority of participants (35%) recommend the implementation of improved network monitoring tools for enhancing security at Federal College of Education, Gidan Madi.

5. Conclusion

All things considered, a thorough evaluation of software design tools for enhancing security measures at Federal College of Education, Gidan Madi, indicates a complicated environment where technological solutions are critical to bolstering the school's overall security posture. The extraordinarily positive response from respondents (65% expressing confidence in the major breakthroughs brought forth by these technologies) indicates a good step towards a solid cybersecurity system.

Importantly, the IT Department's central role and the high degree of departmental cooperation and communication highlight the collaborative and cooperative culture inside the organization, which highlights a shared commitment to a strong security infrastructure. This cooperative culture is essential for developing a comprehensive grasp of security requirements and guarantees that all parties involved—faculty, administrative personnel, and students—feel sufficiently supported.

However, most participants preferred UML (Unified Modelling Language) to express their differing opinions about how well-defined certain software design tools are at improving security, most participants think it greatly enhances detection and response when assessing the impact of software design tools on the institution's capacity to identify and address security incidents, most participants highlight the importance of data encryption, most participants feel extremely well-informed about the most recent advancements in software design tools for security in higher education, majority of participants view the level of support and resources provided by their institution to enhance security measures as excellent, majority of the participants express the belief that their institution adequately addresses the security needs of all stakeholders, encompassing students, faculty, and administrative staff and majority of participants recommend the implementation of improved network monitoring toolsfor enhancing security at Federal College of Education, Gidan Madi.

Taking these results into consideration, we suggested efforts for continuous education. The implementation of consistent and customized training programs is necessary to resolve challenges related to technological compatibility and guarantee that all users are proficient in utilizing software design tools. Maintain the organization's commitment to very frequent security policy updates (80%) to ensure that security processes are in line with changing threats and industry best practices. Establish a framework for the continual evaluation of security measures and solicit feedback from pertinent parties on a regular basis so that strategies can be adjusted to take advantage of new technological developments and evolving requirements.

Acknowledgement

This research was funded by TETFUND under the Institution Based Research (IBR) Annual Intervention, we therefore acknowledge their immense support. Also, we appreciate Federal College of Education, Gidan Madi, Sokoto for their infrastructure support

References

[1] Vyas, B. (2023). Security Challenges and Solutions in Java Application Development. *Eduzone: International Peer Reviewed/Refereed Multidisciplinary Journal*, *12*(2), 268-275.

[2] Alfadel, M., Costa, D. E., & Shihab, E. (2023). Empirical analysis of security vulnerabilities in python packages. *Empirical Software Engineering*, *28*(3), 59.

[3] Pearce, H., Tan, B., Ahmad, B., Karri, R., & Dolan-Gavitt, B. (2023, May). Examining zero-shot vulnerability repair with large language models. In *2023 IEEE Symposium on Security and Privacy (SP)* (pp. 2339-2356). IEEE.

[4] Votipka, D., Fulton, K. R., Parker, J., Hou, M., Mazurek, M. L., & Hicks, M. (2020). Understanding security mistakes developers make: Qualitative analysis from build it, break it, fix it. In *29th USENIX Security Symposium (USENIX Security 20)* (pp. 109-126).

[5] Casola, V., De Benedictis, A., Rak, M., & Villano, U. (2020). A novel Security-by-Design methodology: Modeling and assessing security by SLAs with a quantitative approach. *Journal of Systems and Software*, *163*, 110537.

[6] Silic, M., & Lowry, P. B. (2020). Using design-science based gamification to improve organizational security training and compliance. *Journal of management information systems*, *37*(1), 129-161.

[7] Fannoun, S., & Kerins, J. (2019). Towards organisational learning enhancement: assessing software engineering practice. *The learning organization*, *26*(1), 44-59.
[8] Smith, A., Johnson, B., Thompson, C., & Davis, D. (2019). Security Challenges in Higher Education. *Journal of Cybersecurity Research*, 10(3), 112-128.
[9] Johnson, E., & Brown, G. (2019). Enhancing Web Application Security in Tertiary Institutions: A Review of Software Design Tools. *Journal of Web Security*, 12(3), 112-126