Received: 14 October 2022 | Revised: 30 May 2023 | Accepted: 31 May 2023
DOI: 10.1002/isd2.12285

RESEARCH ARTICLE

WILEY

Implications of regulatory policy for building secure agile software in Nigeria: A grounded theory

Abdulhamid A. Ardo | Julian M. Bass | Tarek Gaber

Department of Computer Science & Software Engineering, University of Salford, Manchester, UK

Correspondence
Abdulhamid A. Ardo, Department of Computer Science & Software Engineering, University of Salford, Manchester, UK.
Email: a.a.ardo@edu.salford.ac.uk

Funding information
Petroleum Technology Development Fund (PTDF) Nigeria-Overseas Scholarship Scheme (OSS)

Abstract

Nigeria is ranked second worldwide, after India, in reported incidences of cyberattacks. Attackers usually exploit vulnerabilities in software which may not have adequately considered security features during the development process. Agile methods have the potential to increase productivity and ensure faster delivery of software, although they tend to neglect non-functional requirements such as security. The implementation of government policies, such as the Nigeria Data Protection Regulation (NDPR) Act 2019, impacts the security activities carried out by agile teams. Despite its significance, there is a paucity of research on security issues especially in the Agile Software Development (ASD) domain. To address this gap, a grounded theory study was conducted with 15 agile software practitioners in Nigeria. Based on our analysis of the interview transcripts, we developed a grounded theory of the security challenges confronting agile practitioners. The four challenges identified were (a) a lack of collaboration between security and agile teams; (b) the tendency to use foreign software hosting companies; (c) a poor cybersecurity culture; and (d) the high cost of building secure agile software. We used these challenges to identify gaps within the existing secure ASD and found a lack of indigenous software hosting companies in Nigeria. Our study also revealed tensions between the Nigerian regulatory environment and agile software developers' compliance. While practitioners acknowledged the government's efforts, there were concerns about the practicality of implementing such legislation. We recommend government action to increase awareness of local software hosting companies' capabilities, and closer collaboration between agile and security teams. Thus, the novel contribution of this article is the development of the policy adherence challenges (PAC) model.

KEYWORDS
agile methodology, grounded theory, Nigeria, secure software development, team collaboration

1 | INTRODUCTION

Governments around the world are introducing data protection policies with the intention of improving their security landscapes (Canedo et al., 2021). Some of these policies include NDPR, General Data Protection Regulation in Europe and the Brazilian General Data Protection Law

(LGPD) (Olukoya, 2022). Other information security and data protection policies include VAHTI (Finnish word meaning "guard") and the National Institute of Standards and Technology 800-53 (NIST 800-53) in the United States. In Nigeria, like the above countries, it is a legal requirement for software systems to be NDPR compliant. With software applications becoming an integral part of human life, especially in the aftermath of the COVID-19 pandemic, the need to adhere to such regulations cannot be overemphasized. While the ubiquitous nature of software systems has enormous benefits, it also raises concerns about software security in the face of increased misuse and cyberattacks. According to the Sophos Group's State of Cloud 2020 Survey, 86% of Nigerian companies were susceptible to cyberattacks (Sophos Group plc, 2021). The survey was based on data from companies hosting their applications on offshore cloud service providers such as Amazon Web Services (AWS), Azure, and Alibaba. In an earlier report, Sophos Group revealed that hackers now have access to the data of some Universities in Nigeria despite the enactment of the NDPR, which was meant to securely guide organizational data handling. It has also been reported that the average cost of data and software breaches jumped from $4.24million in 2021 to $4.35million in 2022 (Tunggal, 2022). This further positions security as the highest priority to consider where traditional and agile methodologies are concerned (Tøndel et al., 2022).

Agile methodologies remain central to software development and engineering practice. They are based on the principles and values of the Agile Manifesto (Beck et al., 2001). These methods were primarily introduced to solve some of the inherent problems of traditional development methods, especially with regard to the frequency of project failures (Dybå & Dingsøyr, 2008). Therefore, the main goal of agile methods remains the improvement of software products' quality and productivity. While agile methods have succeeded in providing organizations with the right processes and tools to proactively handle continuously changing requirements, there are still growing concerns with the integration of non-functional requirements (i.e., security, safety, and usability) within the development process (Riisom et al., 2018). Different security practices, techniques and methods have been provided to help minimize the risk of cybersecurity attack through a secure-by-design process. The study by Tøndel et al. (2019) proposed the adoption of protection poker security practice to estimate the risks in ASD projects. The security-enhanced agile software development process (SEAP) proposed by Baca et al. (2015) provided a method to conduct an incremental risk analysis, and perform security practices such as penetration testing and secure code reviewing sessions. The study reported an improved method to identify and handle agile project risks. Integrating the above-mentioned secure practices/activities has proven to address the security needs of such companies. However, according to Venson et al. (2019), a lack of integrated security practices within the agile development process has led to increased software attacks globally.

Literature is lacking on the perception of agile teams regarding data protection and information security policies in the Nigerian context. The study conducted by Agbali et al. (2020) investigated the implications of the NDPR policy for practitioners, specifically data controllers and processors, in different sectors of the Nigerian economy. A study by Chika and Tochukwu (2020) explored compliance with data protection in Nigeria. Although conducted in a Portuguese context, research by Leite et al. (2021) investigated the impact of the General Data Protection Regulation (GDPR) on software development practices. The study identified that the requirements and modeling phases experienced significant changes. Furthermore, Rygge and Jøsang (2018) used a threat poker to estimate security and privacy risks for agile teams. This aimed to satisfy the requirements for secure-by-design, which is a prerequisite for the privacy-by-design referenced in the GDPR policy. Although Agbali et al. (2020) and Chika and Tochukwu (2020) explored the implications of compliance with NDPR, their studies did not focus on software development. While the study by Leite et al. (2021) related to software development, it was not conducted in the Nigerian context. The study focused on traditional software development methodology and its findings were only applicable to the waterfall model. In comparison, the study by Rygge and Jøsang (2018) investigated an agile context in relation to GDPR. However, none of the above papers studied secure ASD in the context of the NDPR. These papers only considered the NDPR as a legal requirement to which Nigerian companies are mandated to adhere. However, there is a lack of studies within the existing literature that examine the NDPR from the perspective of agile practitioners developing secure software. Thus, this study contributes to the body of empirical evidence on regulatory policy compliance and adoption in secure ASD. Despite the importance of secure regulatory policies in agile environments, current empirical research is limited and so this gap needs to be filled (Moyón et al., 2020). The two research questions (RQs) formulated for this study are:

RQ1. What are practitioners' perceptions of the NDPR for secure agile software development in Nigeria?

RQ2. What challenges do practitioners observe when developing secure software using agile methods in Nigeria?

To answer these RQs, we employed a qualitative research approach using 15 semi-structured interviews with agile practitioners from Nigeria. We adopted a data analysis method informed by grounded theory (Glaser, 1978; Glaser & Strauss, 1967). Respondents were asked about the security practices they use during each phase of the SEAP and how they adhere to the NDPR Act. Practitioners were also asked about the challenges faced during the development of secure agile software.

This paper contributes to the existing body of knowledge in two ways: (i) it identifies the tension between government attempts to regulate cybersecurity space through the NDPR vis-à-vis the difficulties encountered by agile practitioners adjusting to the new regulatory landscape; (ii) it develops a grounded theory describing the challenges faced by practitioners when building secure agile software in Nigeria. The study's emergent theory was called the policy adherence challenges (PAC) model and this paper provides part of the analysis that resulted in the model presented in Ardo, Bass, and Gaber (2022).

The remaining parts of this article are organized as follows: Section 2 presents related work in the field of ASD. Section 3 describes the methodology adopted for the study. The emergent theory, study findings and discussion of the findings in relation to the literature and research questions are presented in Sections 4, 5 and 6 respectively. Finally, Section 7 concludes the study by providing directions for future research.

2 | BACKGROUND

Literature reviews are controversial in the grounded theory method. Some proponents advise against the conduct of an extensive literature review at an early stage to avoid the imposition of any preconceived ideas (Glaser, 1978). However, McGhee et al.'s (2007) notes some arguments that support the need for a literature review before identifying research categories. These arguments include justifying a need for the research, obtaining approval from research ethics committees, and exploring existing knowledge to assess if grounded theory will be an appropriate method.

In this study, a minor review of literature was conducted at an early stage to acquire a basic grasp of agile methods and effectively converse with interviewees. Once the study findings had been substantially developed, a major review of existing literature related to the phenomenon under investigation was undertaken. This approach reflects those of similar studies in the agile software domain which also followed a two-phase literature review process (Bass, 2016a; Hoda et al., 2012a, 2012b).

Consequently, this section reviews ASD literature in general and focuses on developing countries. The challenges of secure agile software are discussed with reference to the introduced regulatory framework in Nigeria, namely, the NDPR.

2.1 | Agile software development

The emergence of the ASD Manifesto in 2001 brought unprecedented change to the software engineering research domain. ASD is an iterative and incremental approach where self-organizing teams adjust to meet changing customer requirements. Several methods have been introduced which adhere to the manifesto and they include scrum, eXtreme programming (XP), Lean and feature-driven development (FDD) (Dybå & Dingsøyr, 2008). All methods were introduced to address the core principles of the agile manifesto. Hence, early research endeavors focused on the adoption of agile methods (Nerur et al., 2005) while the tailoring of agile methods was investigated by Bass (2016b). Despite copious research on ASD in the global north, few exploratory studies exist in developing countries (Mohallel & Bass, 2019; Rahy et al., 2020; Regassa et al., 2017).

2.2 | Agile methods in developing countries

The adoption of agile methods is not a particularly new phenomenon in developing countries where there is widespread use of agile practices by software companies. Prior studies in some countries such as Ethiopia have provided empirical evidence on how the adoption of agile methods can be influenced by user involvement and the nature of the clients and contracts involved (Regassa et al., 2017). The research conducted by Mohallel and Bass (2019) discovered the positive impact of adopting agile methods on customer satisfaction and the development process in Egypt. The study, however, highlighted several problem areas for Egyptian software development companies including the use of inaccurate estimation efforts, a lack of sprint planning, and the relationship between constant pressure on the development teams and software quality.

Akinnuwesi et al. (2013) reported a perception gap between software developers and end-users which in-turn hinders the SEAP. This has led to the poor acceptance of developed software applications and sometimes project failure. The implementation of agile methods in Lebanon was explored by Rahy and Bass (2020) who identified factors enabling and impeding agile information systems development as described by practitioners. Some of the identified impediments include a misunderstanding of agile methodology by both management and team members, and the impact of the political and economic crisis in the country. Sebega and Mnkandla (2017) explored the issues of agile requirements engineering in the South African software industry and discovered a strong likeness for agile principles among practitioners. The study, however, found divergent views among practitioners with regards to integrating non-functional requirements (i.e., security, reliability, and performance) in the agile SEAP. Thus, a common problem among all agile methods studies in developing countries some of which have been discussed in this section is the lack of research exploring the intersection between agile software and the integration of security practices in the development process. However, some studies have considered the implementation of agile security practices in countries such as Finland (Rindell et al., 2018; Rindell et al., 2021).

2.3 | Agile security practices

The implementation of agile methods in the software industry appears to conflict with security practices and requirements (Rindell et al., 2021). Developing secure software means performing a set of security practices or activities while following a software development lifecycle process. However, an agile development process is not always determined in advance. Thus, the main objective of integrating security practices within the development process is to address any security vulnerabilities identified to build an efficient and effective software system.

Rindell et al. (2021) conducted a survey among Finnish agile developers to determine the security practices they adopt during software development. The study used 40 security engineering practices in conjunction with 16 ASD activities to investigate their perceived impacts. The study findings showed discrepancies between the perceived impact of the security activities used and their level of use. The study findings were not, however, compared with any existing baseline studies in other contexts. Newton et al. (2019) presented 12 critical success factors (CSF) used by agile practitioners to address problems associated with traditional SEAPs. Although the CSFs also helped to address tension between ASD and software security teams, they remain conceptual as none of the constructs were validated to determine their relationship. While the emergence of the agile manifesto was a potential source of motivation for researchers to adopt agile development techniques, practitioners still report challenges associated with building secure agile software.

2.4 | Challenges of secure ASD

Existing studies have highlighted some of the challenges of developing secure agile software which include: a lack of cross-team collaboration around security issues (Sánchez-Gordón & Colomo-Palacios, 2020), high costs (Venson et al., 2019), compliance issues (Agbali et al., 2020; Presthus et al., 2018), and a lack of practitioners with skills in both cybersecurity and ASD (Alshaikh, 2020; Egere, 2020; Nägele et al., 2022).

Nägele et al. (2022) conducted an empirical investigation of security issues in large-scale agile environments and discovered a conflict between security governance and the autonomy of agile teams. While the authors tried to encourage cross-team collaboration through regular meetings and training, security knowledge sharing remained a key challenge. Similarly, Tøndel et al. (2022) referred to the collaborative nature between agile teams and security specialists as 'not always optimal'. This was potentially influenced by unclear security requirements and the lack of integration of security practices in the development process.

Sánchez-Gordón and Colomo-Palacios (2020) considered team collaboration in ASD as a culture. They introduced the idea of DevSecOps in their study as a way of integrating security principles by collaborating with cross-teams (security, development, and operations). However, the review highlighted a lack of collaboration between agile and security teams.

A systematic literature review was conducted by Venson et al. (2019), to classify existing studies related to secure software development project costs. The study used different search strategies such as manual, automated, and snowballing to select 54 papers from 47 distinct software engineering or security journals and conferences relating to sources of cost in secure software development. While there was evidence that agile teams adopted security practices which added to the cost of developing secure software, only a few cost estimation models considered security, which could be due to the prevailing organizational cybersecurity culture.

Cybersecurity culture has been described by Da Veiga et al. (2020) as the behavior of humans to protect the information they process through compliance with organizational security policy. This is achieved through regular and effective communication, awareness, training, and educational programs. Security skills and training are considered vital for building a good organizational cybersecurity culture. Alshaikh (2020) reported that five initiatives were adopted by some Australian companies to develop cybersecurity culture. However, a lack of security skills and knowledge continued to be a major cause of software vulnerability (Egere, 2020; Jøsang et al., 2015).

To develop better and more secure software systems, organizations are required to adhere to baseline security practices. However, compliance does not automatically equate to secure practices (Zerlang, 2017). As cybercrimes globally continue to evolve at a fast pace, it is difficult for government policies to keep pace with the dynamic security landscape. The implications of the NDPR launched recently in Nigeria were investigated by Agbali et al. (2020). The study used an institutional theory lens to interpret the policy's level of adoption and implementation. The issues confronting Norwegian practitioners in complying with GDPR legislation were explored by Presthus et al. (2018) while Li et al. (2019) investigated the impact of GDPR legislation on global technology development and how the world leading economies, such as China and the USA could respond to GDPR challenges and opportunities. Thus, we observed from all the studies discussed in this section that both the GDPR and NDPR face certain implementation challenges peculiar to their contexts. In this paper, our focus is on the challenges confronting agile practitioners as a result of the implementation of NDPR.

2.5 | Nigerian Data Protection Regulation (NDPR) Act 2019

The introduction of the NDPR is expected to have significant implications for practitioners since companies are required to protect client data against cyberattacks. With the recent incidences of cyberattacks in Nigeria, there is a need for companies to increase their efforts to protect themselves from cybersecurity threats. The NDPR has further increased the need for companies to engage professional cybersecurity experts as well as data protection officers. However, there is currently a shortage of these professionals in Nigeria (Egere, 2020). This shortage is not particular to Nigeria, as a study by Li et al. (2019) noted the need for both governments and company technology managers to invest more in cybersecurity education and training to comply with the GDPR policy.

3 | RESEARCH METHOD

The main objective of this study is to explore practitioners' perceptions of the NDPR policy in relation to building secure agile software in Nigeria. To achieve this, a method informed by grounded theory was adopted. Grounded theory is a qualitative method for developing theory from qualitative data (Glaser & Strauss, 1967). The method is especially relevant in software engineering research as it enables the development of theories relevant to practitioners. The method also enabled a deep understanding of a phenomenon in a unique context (Glaser et al., 1968). There is a lack of literature on the intersection between the development practices of secure agile information systems and practitioner compliance with the NDPR in Nigeria. Thus, the use of a grounded theory method is relevant and makes the study context unique to information systems research.

In exploring the Nigerian software industry, agile practitioners were recruited through personal contacts and professional networks, such as LinkedIn, as has been done in existing studies (Sharma & Bawa, 2020; Terpstra et al., 2017). This study applied two sampling techniques, namely, snowball (Miles & Huberman, 1994; Patton, 2002) and intensity (Patton, 2002). A snowball technique was used at the initial stage to recruit participants and establish a broad perspective on the phenomena investigated from different stakeholder views. Also, due to confidentiality and non-disclosure agreements, practitioners were unwilling to talk to strangers about the sensitive security issues of their organization. This made snowball sampling a suitable technique to adopt (Sharma & Bawa, 2020). Intensity sampling was adopted in the later stages of the data collection to select practitioners in managerial roles, such as CTOs, to provide in-depth knowledge of the phenomenon. The combination of multiple sampling approaches provided some elements of methodological triangulation. This offered insights into both the current problems of the phenomenon investigated and the motivation for adopting existing practices. The underlying motivations for adopting these practices are difficult to obtain in studies using survey methods (Bass, 2013).

3.1 | Research sites

Nigeria was chosen because it is inarguably an emerging powerhouse for software development in Africa and considered the largest economy on the continent (Nigeria Bureau of Statistics: Nigerian Economy Largest in Africa (2019/2021); Sowunmi et al., 2016). Most Nigerian software development companies are based in Lagos and Abuja, hence we decided to limit our data collection to those sites. According to Ogunyemi et al. (2018) and Soriyan et al. (2001), these two cities represent the heart of the Nigerian software industry. Consequently, we chose sample research sites comprising a mixture of small, medium, and large sized companies which have implemented agile methodologies as shown in Table 1. The research sites included companies operating across key sectors, such as IT services, healthcare, financial services, manufacturing, digital solutions, and educational software solutions. The selected companies are involved in secure ASD using agile methods. Among the firms is a company registered in the UK that offers digital innovative solutions with deep financial expertise. Engaging with diverse research sites provided richness to the data collected and established the legitimacy of our findings.

3.2 | Grounded theory

Grounded theory is defined as a methodology of inductively generating theory from data. The methodology was first introduced by Glaser and Strauss (1967) and aims to develop new theories rather than verify or extend existing ones. In our research, grounded theory was chosen for three main reasons. Firstly, the research phenomena of secure agile development—focuses on individuals and interactions. As such, the use of grounded theory allows for the in-depth study of the practitioners' interactions and their behaviors (Parry, 1998). Secondly, grounded theory is the most appropriate for investigating a phenomenon like the intersection of government policies and secures ASD where there is a lack of the existing literature. Thirdly, there is evidence of the increasing adoption of grounded theory to study the behavior of agile teams (Coleman & O'Connor, 2007; Shastri et al., 2021).

The grounded theory method, as developed by its originators first evolved into two main approaches, namely: Glaserian (Glaser, 1978, 1992) and Straussian (Strauss & Corbin, 1990). The two initial approaches differ with regards to the research problem formulation, data analysis techniques and coding methods. While the Glaserian approach proposes starting the research process with a general interest in the phenomenon, the Straussian method recommends a more careful definition of the research problem. Other grounded theory strands include Constructivist (Charmaz, 2000, 2006, 2008) and Critical realist approaches (Bunt, 2018; Kempster & Parry, 2011; Oliver, 2012).

TABLE 1 Participants' and organizations description.

Participant code	Job title	Experience in agile (years)	Interview date	Interview location	Business type	Organization
ESSco1_DE	DevOps Engineer	9	07/02/2021	Abuja	Educational Software Solutions	Medium-sized
ESSco1_PROJ-MGR	Product Manager	16	07/02/2021	Abuja	Educational Software Solutions	Medium-sized
ESSco1_SSE1	Senior Software Engineer	9	14/02/2021	Abuja	Educational Software Solutions	Medium-sized
ESSco1_BEE	Back-end Engineer	9	22/02/2021	Abuja	Educational Software Solutions	Medium-sized
ESSco1_PROD-MGR	Product Manager	13	02/03/2021	Online	Educational Software solutions	Medium-sized
ITSERVco2_SE1	Software Developer	6	05/03/2021	Abuja	IT Services & Consulting	Small
ITSERVco3_QAA	Quality assurance analyst	8	17/04/2021	Lagos	IT Services & Consulting	Small
HSCco1_SSE2	Senior Software Engineer	8	18/04/2021	Lagos	Healthcare Services Company	Large
ITSERVco1_STP-MGR	Security Technical Program Manager	9	11/05/2021	Online	IT Service Management Company	Large
FSSco1_SDE1	Senior DevOps Engineer	11	17/05/2021	Lagos	Financial Services & Solutions	Medium-sized
FSSco1_FLM	Frontline Manager	11	20/05/2021	Online	Financial Services & Solutions	Medium-sized
ESSco1_PROD-MGR	Project Manager	11	13/06/2021	Abuja	Educational Software Solutions	Medium-sized
DSco1_SETL1	Software Security Team Lead	10	25/06/2021	Online	Digital Solutions	Medium-sized
ESSco1_CTO1	Chief Technology Officer	24	01/07/2021	Abuja	Educational Software Solutions	Medium-sized
MFGco1_ITA	Manager, Internal IT Audit & Operational Risk	17	04/07/2021	Lagos	Manufacturing	Large

In this research, we adopted the Glaserian grounded theory since we investigated the phenomena by excluding any prior knowledge and started with a general awareness of the topic, similar to other previous studies (Hoda et al., 2012a; Stray et al., 2016). We ensured that the codes, concepts, and categories emerged from data collected in contrast to the approach prescribed by Straussian grounded theory.

3.3 | Data collection

Data were collected over a period of 6 months (February–July 2021) by conducting 15 face-to-face and virtual interviews with Nigerian agile practitioners who were engaged in developing secure software. Interviews were adopted for the data collection in this study because it provided the authors with in-depth knowledge of the phenomena (Kvale & Brinkmann, 2009). Also, interviewing can provide compelling evidence of an individual's worldview. To protect the anonymity of interviewees and their companies, abbreviations for their business sectors and job titles were used. The interviewees' software development experience ranged from 6 to 24 years, as detailed in Ardo, Bass, and Tarek (2022).

We started the data collection by first interviewing six practitioners before the remaining nine were recruited and interviewed. The rationale for starting then pausing the data collection was to provide the authors with an opportunity to transcribe the early interviews and analyze the data using coding and memo writing. This gave the authors a chance to collect their thoughts regarding the data collected and refine the interview guide. We adopted an open-ended semi-structured questioning technique which afforded the interviewer an opportunity to observe interviewees' actions and mannerisms in addition to their verbal information. It also helped the authors to identify issues of real concern to practitioners instead of forcing a topic on them. All interviews were conducted in English and audio recorded with the permission of the interviewees. Although Glaserian grounded theory approach advises against recording interviews, it afforded the authors an opportunity to capture all the information without losing any details, which was similar to the approach used by Hoda et al. (2012a). All interviews and transcriptions were conducted by the first author while the two other authors performed the review and validation. The first author began each interview by explaining the purpose of the session and then assuring participants of the confidentiality. The average duration of the interviews was 45 minutes. The interviewer used probing questions to explore relevant topics not included in the interview guide, but which appeared important to the interviewees.

The initial interview guide for this study consisted of three parts. The first part contained questions on practitioner perceptions of the adoption of agile methods. The second part explored the security practices adopted by agile practitioners. The third part consisted of questions on the interviewees' personal details such as their educational level and professional background, current job titles and the nature of the projects they were handling.

As the data collection progressed, the interview guide was modified and refined to focus more on agile security practices, NDPR compliance and the challenges confronting practitioners during the development process. A closing question was added to give interviewees the chance to make comments or add anything else they considered relevant which had not been addressed. Both the initial and adapted interview guides are publicly available on Figshare (Ardo, Bass, & Tarek, 2022).

Another important source of data collection in the grounded theory approach used was document reviews. Some participants provided access to their company policy documents and guidelines for developing secure agile software. Other documents on secure agile practices were publicly available on some of the participating companies' websites. However, it was not possible to access documentation for the security designs and architecture of projects as all companies regarded these as highly commercially sensitive.

The direct observation of teams was not considered due to the cross-sectional time horizon of the study. As participants were engaged for a short period, this technique could have been intrusive and behaviors may have been altered which would have been difficult to determine since they would not have been observed over a long period of time (Stray et al., 2016).

3.4 | Data analysis techniques

Data analysis involved the four key techniques of Glaserian grounded theory: coding (open and selective), constant comparison, memoing and theoretical saturation (Glaser, 1978, 1992, 2001).

3.4.1 | Coding

This study adopted the two coding stages of Glaserian grounded theory, namely open and selective. Open coding is the initial analysis phase of a grounded theory study. The selective coding builds on the initial phase by continuing the analysis process but is limited to categories discovered from the core category (Glaser, 1992). The three levels of abstraction in this approach are codes, concepts, and categories. Codes are formed from statements in the interview transcripts while groups of codes are known as concepts and the grouping of different concepts form categories.

Open coding: We began the open coding process by reviewing each of the interview transcripts line-by-line to allow codes to emerge from the data collected (Glaser, 1978, 1998). We identified codes (themes) by reviewing the interviewees' responses. In total, 36 codes were identified after reviewing the first interview transcript. After analyzing the second interview, 13 new codes emerged. At the initial stage, we opted for the manual coding of interviews to ensure accuracy which also reminded us of the social and emotional aspects of an interview (Vaivio, 2012). All transcripts were completely coded as the author had no idea at the initial stage which data would be relevant as the analysis progressed. As the dataset grew, the transcribed data was imported to the NVivo-12 qualitative data analysis tool (Edhlund & McDougall, 2019).

Selective coding: After a core category was established, the researchers moved to selective coding. This is a technique that focuses on selectively coding only variables related to the core category in interview transcripts. While open coding was time consuming, selective coding was quicker because we only focused on categories related to the core. It was also less challenging because we were familiar with the constant comparison method. As the selective coding progressed, less categorization was noticed, highlighting the fact that data collection and analysis was reaching theoretical saturation (Glaser, 1992).

3.4.2 | Memoing

Memoing in grounded theory denotes written accounts capturing the relationships between codes and concepts as they develop into categories (Glaser & Strauss, 1967). Based on the codes identified, statements were written on each topic with embedded quotations as supporting evidence. Adhering to the recommendation by Glaser (1978), the analysis process was often paused so that the researchers could reflect on the data collected. Fifteen memos were written while conducting this study. However, only five which relate to the theme of this article have been presented in Section 5. The memos helped to converge ideas as more transcripts were analyzed. This process represents an important stage of the theory generation process (Adolph et al., 2011).

3.4.3 | Constant comparison

This stage involves iteratively comparing among different interviewees, their job titles, and organizations. It starts by forming codes from interview statements, grouping codes to form concepts and combining related concepts to create categories (Glaser, 1992). As the analysis progressed, the constant comparison technique was used across the dataset. Codes from the same interview were continuously compared with others from subsequent transcripts. In total, 584 codes were generated at the end of the open coding process.

The data analysis process is illustrated in Figure 1. Practitioners described the paucity of software hosting companies as some requirements have not been fulfilled in the industry. These requirements include the low security level of companies, infrastructure deficits

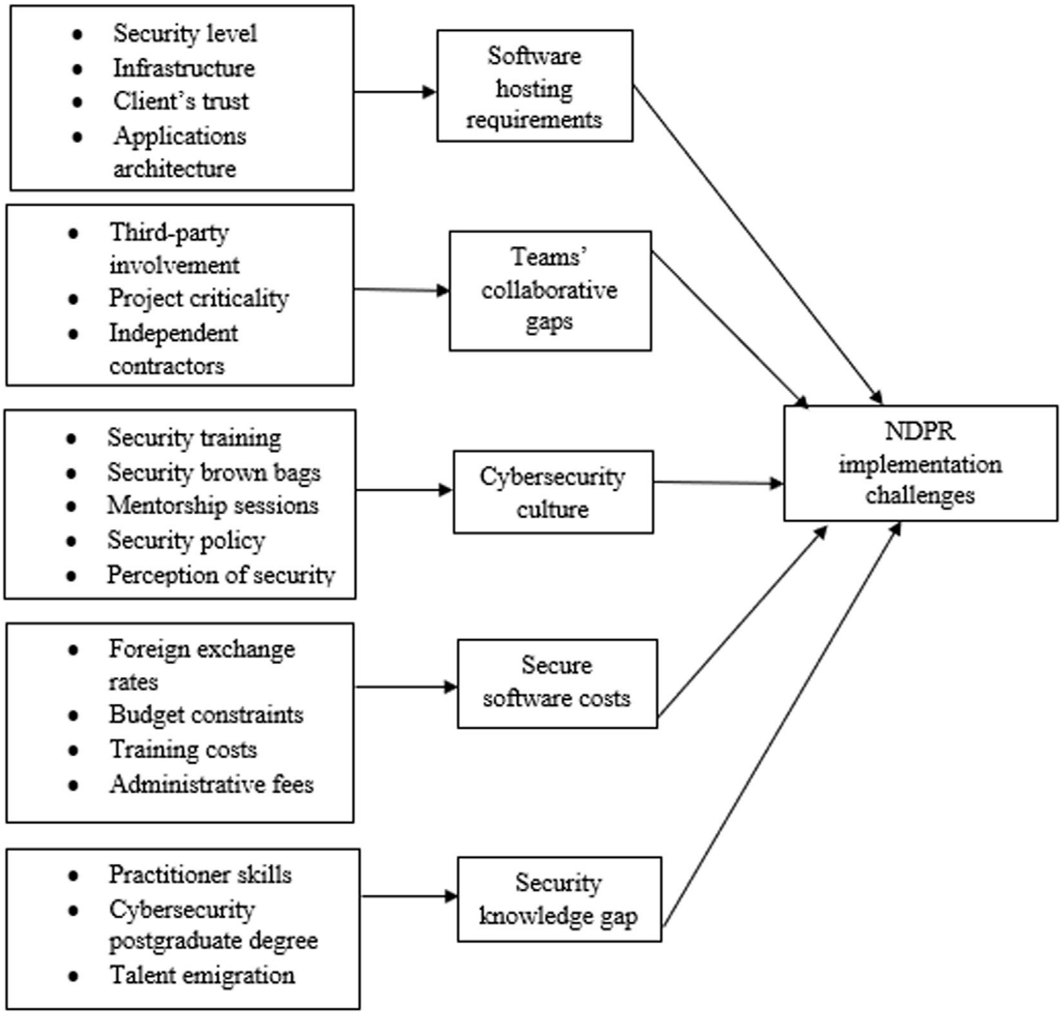


FIGURE 1 Grounded theory development example. It illustrates the data analysis process that led to development of the study grounded theory. The items in the boxes on the left represent different concepts identified in interview transcripts. The middle boxes represent categories which were derived through the constant comparison technique of GT. Finally, the box on the right labeled "NDPR Implementation Challenges" is the GT core category derived from iteratively grouping of identified categories.

(i.e., electricity and internet connectivity), a lack of client trust in the capacity of indigenous hosts, and the absence of appropriate architecture for their applications. Similarly, practitioners describe some practices as responsible for widening the collaborative gap between agile and security teams. These practices in companies include involving external security experts instead of the security team and independent work practices between teams. Using the constant comparison technique, we iteratively grouped and categorized the concepts identified from the interview transcripts.

To further illustrate how the results presented in this article were derived, we provided a snapshot of the emergence of the category "NDPR implementation challenges." The category describes some of the challenges confronting the agile practitioner in their efforts to adhere to a new regulatory policy.

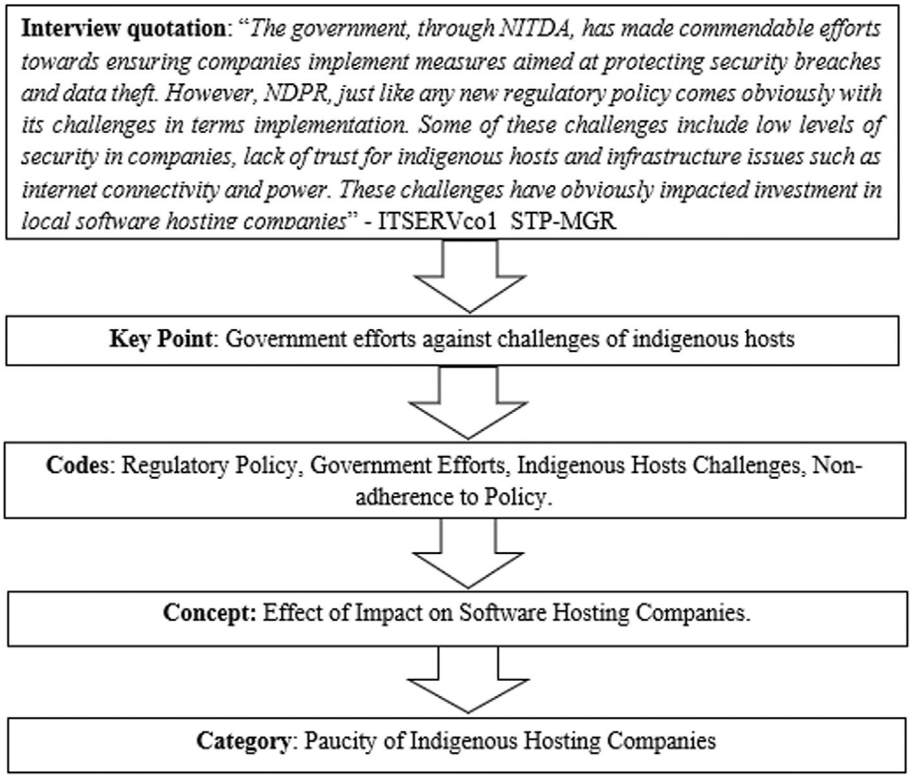
We first began the analysis process by gathering the key points from each interview transcript. We used two-to-three-word phrases to summarize the points known as codes. An example is illustrated in Figure 2:

3.4.4 | Theoretical saturation

In this study, we adopted Glaser's concept of theoretical saturation which is a point in the data collection process where new categories no longer emerge. The authors stopped conducting additional interviews when we noticed that analyzing more transcripts would not lead to the emergence of new categories and no new information seemed to emerge. The already identified categories formed the basis for the grounded theory presented in the findings section of this paper. At this point, evidence from the data collection can be said to have converged. Thus, conducting and analyzing additional interviews would have no impact on the categorization (Glaser & Strauss, 1967).

3.4.5 | Grounded theory write-up

The writing-up phase represents the final activity of the grounded theory development process after finalizing all the coding and sorting of memos. The grounded theory developed in this paper represents an explanatory theory (Gregor, 2006) which also has been the approach of



Interview quotation: "*The government, through NITDA, has made commendable efforts towards ensuring companies implement measures aimed at protecting security breaches and data theft. However, NDPR, just like any new regulatory policy comes obviously with its challenges in terms implementation. Some of these challenges include low levels of security in companies, lack of trust for indigenous hosts and infrastructure issues such as internet connectivity and power. These challenges have obviously impacted investment in local software hosting companies*" - ITSERVco1 STP-MGR

Key Point: Government efforts against challenges of indigenous hosts

Codes: Regulatory Policy, Government Efforts, Indigenous Hosts Challenges, Non-adherence to Policy.

Concept: Effect of Impact on Software Hosting Companies.

Category: Paucity of Indigenous Hosting Companies

FIGURE 2 Data analysis process. Further describes steps which led to the emergence of the study core category "NDPR implementation challenges". The process began with gathering key points from interview data. The key points were iteratively grouped to form higher levels of abstraction known as codes, concepts and finally category.

previous studies on ASD (Bass, 2016a; Hoda et al., 2012a; Masood et al., 2020; Shastri et al., 2021). Our results describe the challenges involved in adhering to regulatory policy—*NDPR adherence challenges*—to further demonstrate the outcome of the grounded theory. In the next section, we present the emergent theory and relationships between its constructs.

4 | RESULTANT GROUNDED THEORY

The emergent theory developed from the data analysis is presented in this section. As explained earlier, the selective coding technique was used to trim categories unrelated to the core. This study engaged practitioners involved in secure ASD from companies operating in different business sectors. The practitioners described the challenges they faced in their attempts to adhere to the NDPR policy. ESSco1_SSE1 explained:

> When they first released [the] NDPR and we were reviewing the document, my CTO said this policy is not realistic. Looking at the architecture of the applications we build, Nigeria lacks the indigenous hosting capabilities that we need. Power and internet interconnectivity remains big challenges to date. (Senior Software Engineer, Company B)

From the above quotation, which was echoed by others on a similar theme, the researchers coined the concept "adherence" to the NDPR which represented a major concern for agile practitioners. This is due to the challenges of implementing the policy. Categories of our emergent theory that represent challenges to adherence include unawareness, distrust, compromise, and culture. These categories represent the building blocks of our grounded theory which explains the difficulties confronting agile practitioners. The categories and relationships among them are explained in this section.

4.1 | Unawareness

It emerged that many practitioners are unaware of the existence and capabilities of the few Tier IV software hosting companies in Nigeria. These companies are in the major cities of Lagos, Abuja, Port-Harcourt, and Kano. In most situations, because of the infrastructure challenges bedeviling the nation, practitioners automatically choose foreign hosts without exploring the capabilities of indigenous companies. This inclination toward foreign companies suggests unawareness among practitioners of what exists in the country. While very few of them may be aware of local hosts, practitioners still prefer offshore companies which is partly due to concerns regarding distrust. Hence, this creates tension between the government and practitioners which may also be attributed to the limited number of hosting companies.

Apart from a lack of awareness of the existence of indigenous hosts, some of the interviewed practitioners were not particularly conversant with the provisions of the NDPR Act. ITSERVco2_SE1 stated:

> Yes, I have heard of it but do not know the details of the regulation. Although I know the part about software and data hosting but sincerely, we have always hosted offshore… (Software Developer, Company L).

4.2 | Choice and trust

Policy adherence has a positive relationship with the choices available to practitioners. The same was found with client trust. When different choices were available to practitioners, they chose the one they trusted the most. In this study, the agile practitioners chose foreign hosting companies because they trusted their capabilities. FSSco1_SDE1 explained:

> Clients will not be willing to listen to excuses when they get hacked or can't transact due to downtime. Most of them don't care how you do it, so we make decisions based on what we trust and are comfortable with… (Senior DevOps Engineer, Company K).

4.3 | Distrust

We discovered that both the agile practitioners and their clients do not trust the capabilities of indigenous software hosting companies which we refer to as "Distrust". HSCco1_SSE2 explained: Medical software are safety critical and so clients prefer we host it offshore which we are also more comfortable with due to downtime issues of local hosts. (Senior Software Engineer, Company O).

4.4 | Compromise

This category emerged because of the instances of practitioners compromising software security due to cost. Most companies—especially SMEs—compromise software security and data privacy by hosting applications offshore because it is cheaper and saves on administrative fees. This poses a choice, namely, to adhere to the NDPR regulatory policy or prioritize cost considerations. Adherence to the NDPR, which is a requirement for software development stipulated by the Nigerian government, is then compromised by practitioners due to budget constraints. As described by MFGco1_ITA, "The truth is the few Nigerian hosting companies' charges are too high since their service charges are based on dollar-to-naira exchange which is very unstable as you know…" (Manager, IT Security & Operational Risk, Company H). Other concepts impacting on the "compromise" category include company policy, available skills and knowledge, security maturity and cost.

4.5 | Culture

Our research revealed a desire by agile practitioners to adhere to the NDPR by building an organizational culture. In certain organizations, the concept of culture is propagated through company policy documents. MFGco1_ITA stated: "Employees are guided by our security policy document during software development…" (Manager, IT Security & Operational Risk, Company H). It further emerged that non-adherence to the NDPR can be attributable to the concepts of "unawareness" of the policy and "compromise" meaning compromising essential skills over the cost of security training.

4.6 | Organizational culture and costs

Cost is a big determinant that shapes organizational culture when developing secure software. Sources of cost include the security training for team members, the purchase of software and hardware tools and the payment of fees to regulatory agencies. While adherence to regulatory policies is important and necessary, budget constraints represent a significant barrier. ESSco1_CTO1 explained: "We pay for services and purchasing hardware and software in dollars which can be very expensive you know" (Chief Technology Officer, Company B).

4.7 | Policy adherence challenges

Our study's social process sought to understand practitioner behavior toward a regulatory policy. We termed the social process, "policy adherence challenges", which explains the reasons agile practitioners are refusing to comply with the NDPR during the SEAP. This paper introduced the policy adherence challenges (PAC) model. The constructs and relationships in our emergent theory, PAC, were derived using a method informed by grounded theory. Figure 3 shows the different stages of the model. The model starts with the current state of non-adherence at stage 1. Investigating the reasons why practitioners are not compliant moves the policy adherence process to stage 2. Through practitioner engagement, the challenges to non-adherence were identified. The adherence process either moves to Stage 3 (tension phase) or the pursuit state (NDPR adherence achieved). When the challenges identified are not resolved, the process moves to Stage 3 where tension is observed, and the circle backtracks to stage 1 (state of non-adherence) of the NDPR. However, when there is positive collaboration between the government and agile practitioners, practitioner resistance can be overcome, and adherence achieved to terminate the process.

The emergent theory presented in this paper provides a foundation to study regulatory policy adherence in Nigeria. Further research could focus on other developing countries to investigate whether any tension exists between the government and agile practitioners.

4.8 | Overcoming adherence challenges

While seeking to promote adherence to NDRP, the problem of practitioner resistance (seen in stage 1), means that some of the challenges observed include unawareness, distrust, compromise, and culture. The recommended solution to the identified challenges is for the government to make practitioners aware of and encourage trust in the capabilities of local hosting companies. This will also be dependent on government investment to close the infrastructure gap and support SMEs and other start-ups to reduce the risk of making compromises. An increase in government sensitization of practitioners on the significance of NDPR compliance will steer the non-adherence process in a positive direction. Thus, the more practitioners are aware and trust indigenous hosts, the more likely adherence will be achieved.

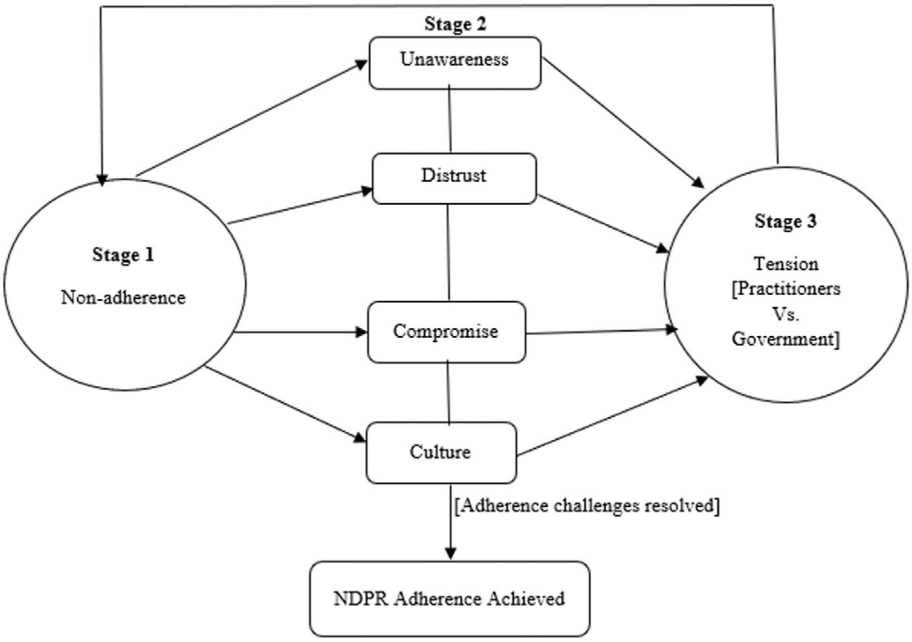


FIGURE 3 Policy adherence challenges (PAC) model. Describes the different stages of the model. Stage 1 shows the current state where there is a general non-adherence to policy regulation by interviewed practitioners. Stage 2 revealed the factors impeding compliance in practice. The model can either move to the pursuit state (where NDPR adherence is achieved) or Stage 3 (tension) depending on the regulatory environment and practitioners' behavior. When the Stage 3 is reached, it backtracks to stage 1 and the circle continuous until there is a positive collaboration between government and practitioners for it to get to the pursuit state.

5 | FINDINGS

The grounded theory analysis conducted in this study resulted in the identification of some security challenges which confront agile practitioners in Nigeria. These security challenges are due to the paucity of software hosting companies, the existence of a collaborative gap between security and development teams, the cost factor in building secure software, a knowledge gap concerning secure software, and a poor cybersecurity culture in organizations.

5.1 | Tension within a changing regulatory landscape

In this study, we notice a tension between software development companies' adherence to some sections of the government's NDPR Act 2019 and the guidelines for Nigeria's content development in ICT. One of the senior software engineers from an educational software solutions company explained the impracticability of hosting their applications locally.

> "Honestly, it would be very difficult to find a suitable company that can host our software applications here in Nigeria with all the challenges, and even if there are, it must be very expensive…" (ESSco1_SSE1).

While practitioners find that it is almost impossible to host their applications locally, the government claims it has made significant progress toward the adoption and utilization of ICTs. The government has also further claimed that data and information hosting should be inevitable in both the public and private sectors due to the adoption of ICT in Nigeria. The Former Director General of NITDA and currently the Nigerian Minister of Communications and Digital Economy, Professor Isa Ali Pantami, was reported in the Vanguard Newspaper of December 5, 2017, to have said, "We condemn the current practices by both public and private organizations hosting data offshore, despite having highly reliable Tier III & IV Data centres, certified by various international organizations."

According to section 14.1 of the Nigerian content development for ICT guidelines, the government has made it mandatory for all Information Technology companies to host data locally.

> It is mandatory for data and information management firms to host government data locally within the country and shall not for any reason host any government data outside the country without an express approval from NITDA.

While the government has claimed that existing data centers guarantee almost 100% availability with multiple levels of security, practitioners in this study do not agree with this assertion. One of the Chief Technology Officers (CTO) interviewed in this study, cited the infrastructure deficit in terms of electricity and internet connectivity among the impediments to adhering to government policies.

> We have [an] infrastructure deficit and so the system is going to be subject to those challenges. So, … you need constant power supply, physical security, you have to provide connectivity and so on and you know these are serious costs… (ESSco1_CTO1).

Apart from the widely recognized infrastructure deficit in Nigeria, there are instances where clients prefer their applications to be hosted offshore. Clients working in certain sectors, such as healthcare, tend to trust offshore companies to manage their software applications. They identified issues of downtime which could be disastrous to their business. According to one of the software engineers interviewed, Clients doing business in the safety critical sectors prefer to host offshore as they don't trust the local companies, we have in Nigeria … (ITSERVco2_SE1). While adhering to government policies is good, securing client trust is also important. Thus, if clients continue to distrust local hosting companies, they will still prefer their data and software applications to be hosted offshore.

5.2 | Security and development teams collaboration

Team collaboration is an essential principle of the agile values which align with the major aims of secure software development and SecDevOps. In this study, we discovered a gap in the collaborative nature between agile teams and the security specialist as highlighted by one of the interviewees:

> Currently at the company where I am working, it is mainly the role of the two security specialists to handle the security aspect of things although as the software team lead, I do get called sometimes but the rest of my team members are not involved… (DSco1_SETL1).

In companies without security teams, senior engineers collaborate with contracted third parties. However, this is dependent on the project's sensitivity and available budget as described by a CTO:

> We don't have like a security team, but the senior engineers base their experience on understanding the security needs of the system and if there is a need we get some security experts, and they handle it together depending on the nature of the application and our budget (ESSco1_CTO1).

Similarly, in SMEs with smaller security teams, experts are contracted to work with those responsible for handling security issues as highlighted by a Frontline Manager:

> At XXX since the size of our security team is small, we have partners that are experts in cyber security, cyber investigation and OSINT intelligence whom our security team works with through the whole software development process (Frontline Manager).

In some companies, the development and DevOps teams work collaboratively while an information security team functions independently, as explained by a Senior DevOps Engineer, "The Developers and DevOps team actually work together on all issues while the information security team test for security alone" (FSSco1_SDE1). Consequently, practitioners highlighted existing gaps between the agile teams and security specialists.

5.3 | The cost factor of building secure software

The practitioners interviewed in this study mentioned cost as an impediment to their companies when building secure software. One of the IT Managers said, "It is cheaper for you to just send your data outside and have Microsoft or someone protect you…" (MFGco1_ITA). Firstly, there are additional costs associated with building secure software such as the administrative fees of maintaining a data center in Nigeria. He further emphasized the administrative cost of secure software:

> You have to consider the administrative cost of having a company like that in Nigeria. You are going to run at a loss. How much will the subscribing companies pay you? You need to have many customers to break-even. It doesn't make any sense… (MFGco1_ITA).

Secondly, some practitioners discussed the need for a budget in projects involving outsourced security experts or consultancy services. One of the CTOs explained this is a challenge as software development projects, especially in developing countries, are constrained by resources:

> When you want to build a very good and secure software and assuming you have to involve security experts then of course you are going to need a budget for that from the meagre resources allocated… (ESSco1_CTO1).

The project budget also needs to make provision for costs such as the purchase of security tools, cloud services, and the training of employees. A senior DevOps engineer from a financial services and solutions company mentioned that security tools and the adoption of cloud services provide an extra level of security in software development, "At times you have to buy some security tools and pay for cloud services just to have [an] extra level of security. It's expensive but better than … the consequences" (FSSco1_SDE1). While practitioners are of the opinion that the cost of tools and cloud services are high, the consequences of security breaches are far more devastating to an organization. In terms of training, the DevOps engineer suggested appropriating some funds for staff training on the use of tools when none of the company's employees are competent in their use. Thus, the DevOps engineer said, "The company needs to provide … for training one or few persons who will train the others over time" (FSSco1_SDE1).

Thirdly, building secure software is expensive which can force practitioners to compromise the quality of their applications. In addition, issues around exchange rates and payment for security services continue to be challenging. A practitioner from an educational services and solutions company explained, "If you want to get good services then you pay them in dollars and the exchange rate is actually not on our side which makes these services quite expensive…" (ESSco1_CTO1). Thus, software companies sometimes make compromises and work with the available resources at their disposal to develop applications which are as secure as possible.

5.4 | Cybersecurity culture

Practitioners in this study discussed different ways of building a cybersecurity culture which were subsequently categorized into artifacts and values. Firstly, practitioners discussed artifacts in terms of increasing awareness through security training and encouraging employee behaviors or mindsets. According to one of the interviewed CTOs, "We don't actually do security specific training; however, occasionally we invite experts to talk about basic security like handling passwords, data encryption and other stuff to raise security awareness…" (ESSco1_CTO1). However, depending on a project's security requirements, some senior engineers may be trained. The CTO said, "If we are building a mission critical application, certain senior engineers may be trained…" (ESSco1_CTO1). Some of the CTO's views on offering basic training to practitioners were corroborated by a software engineer from an IT consulting company: "We have not really been involved in security training…" (ITSERVco2_SE1). In contrast, other companies offer lots of security training. A Senior DevOps Engineer said, "We implement security through lots of trainings. We call it train and un-train or learn and un-learn. It's just a way to increase awareness which happens every quarter…" (FSSco1_SDE1).

Apart from formal training, practitioners in this study shared several effective strategies which they used internally to train other colleagues such as brown bag sessions and mentorship. For example, a senior software engineer described, "We do engage in brown bag session to share knowledge with other colleagues about security or any development although it's a rare practice here…" (ESSco1_SSE1). The practitioner also explained the concept of mentorship by saying, "In XXX Senior engineers do mentor junior and mid-level engineers on strategies of building secure applications based on the experience garnered over the years…" (ESSco1_SSE1). The security mindset remains an issue because the priority for practitioners is always to deliver working software. A practitioner from a manufacturing company described the developer mindset by saying, "The main concern for most practitioners is getting working software. They don't think much about users' protection because they mostly don't have the right security training to understand the implications…" (MFGco1_ITA).

Secondly, what practitioners share in the form of values, such as company security policy or code of conduct, is an important element for building cybersecurity culture. In this study, one of the interviewed IT Managers described their company policy, "We got our software development documentation which guides developers on how to secure their code and implement security in the design of their applications…" (MFGco1_ITA). In contrast, some organizations have general policies or guides on software development with some sections devoted to building secure systems. One practitioner explained that some security policies are only introduced after experiencing breaches,

> We don't have security specific documentation but ensure our policy include issues of building secure software. Some policies are introduced after we had a nasty experience and then realised, we needed this and that… (ESSco1_CTO1).

Practitioners in this study highlighted the lack of a cybersecurity culture in their organizations due to the absence of security specific training and a security policy.

5.5 | Secure software knowledge gap

Developing secure software largely depends on the knowledge and skills of practitioners. In this study, an application security engineer highlighted the paucity of cybersecurity courses in Nigerian Universities, "We don't have cybersecurity as a course in most Universities… I don't think we got any school offering it at M Sc level…" (ITSERVco1_ASE1). This practitioner considered education a key component that needs to be acquired before addressing industry-based issues. A practitioner working at a financial services company explained that a knowledge gap also manifests in many companies when they acquire new security tools, "Apart from cost constraints, [a] knowledge gap comes when we acquire some security tools and don't really have anybody that can deploy it …" (FSSco1_SDE1). While the issues around cost are a major impediment to software development projects in Nigeria, a software engineer considered the skillset a greater challenge to building secure applications. "Cloud services could actually be expensive and other resources problem are there. However, resources are not the major thing, it's the problem of skills…" (ITSERVco2_SE1).

Neglecting security in the development of software applications has also created emigration challenges among the very few skilled practitioners available in the context. Since skilled practitioners are sought globally due increasing cyberattacks cases, the number of professionals leaving Nigeria continues to increase. As explained by an application security engineer, "Security is one space where there are lots of job opportunities and the skillset is rare. So, there is a lot of migration of talents from Nigeria to different parts of the world…" (ITSERVco1_ASE1).

Thus, the practitioners interviewed assert there is a knowledge gap in secure software development.

6 | DISCUSSION

This section compares existing literature on regulatory policy adherence for secure ASD with our study findings from the practitioner interviews. Urquhart et al. (2010) highlighted that theoretical integration is among the five techniques for assessing grounded theory research. Thus, comparing our emergent theory with existing literature contributes to the generalizability of our study.

6.1 | Practitioners' perceptions of the NDPR for secure agile software development in Nigeria (RQ1)

To answer the research question "What are agile practitioners' perceptions of the NDPR for secure agile software development in Nigeria?", our data highlighted the tension between the government and agile practitioners. Interviewed practitioners were unanimous in their opinion that adhering to the NDPR remained a challenge when hosting data for ASD activities with the intention to improve the security of applications. They stated that the lack of infrastructure, such as electricity and internet connectivity, made the use of Nigerian hosting companies risky. While there is a paucity of empirical evidence on practitioner adherence challenges regarding the NDPR, we found such studies on the GDPR. Poritskiy et al. (2019) examined the challenges of adhering to the GDPR by Portuguese Information Technology (IT) companies. The key challenges impeding adherence to the GDPR were the execution of system audits, and the rights to erasure. Another study by Presthus et al. (2018) explored GDPR adherence by Norwegian companies. Although the study did not find any tension between the government and practitioners, it reported a challenge in understanding the financial sanctions of the regulation with regards to how fines were calculated. Practitioners' greatest concern in this study was adhering to Article 17 of the GDPR which relates to individual rights to have some of their data erased by organizations. While adherence to regulatory policies such as the GDPR is well-established in literature, empirical evidence on challenges to adhering to regulatory policies (such as the GDPR and NDPR) is still lacking (Usman et al., 2020). Thus, this study complements the existing body of knowledge through the development of the PAC model.

6.2 | Challenges of developing secure software using agile methods (RQ2)

The second question answered in this study is "What are the challenges of developing secure software using agile methods in Nigeria?" We discovered that agile practitioners face challenges in complying with the NDPR when hosting software applications locally. They complained of an infrastructure deficit and sometimes the preference for offshore companies among their customers. The challenges identified include the lack of collaboration between agile and security teams, the high cost of building secure software, a poor organizational cybersecurity culture, and the existence of secure software knowledge gaps.

Firstly, we discovered from practitioner interviews that there is a lack of collaboration between agile and security specialist teams. We noticed a collaborative gap between teams in most SMEs where our data were collected. Our findings corroborate the study by Tøndel et al. (2022) which described the collaborative nature between agile and security teams as often being sub-optimal. Conflict was reported in balancing team autonomy and security governance when collaborating between different teams. While Nägele et al. (2022) proposed two new roles to improve collaboration between teams, this is only feasible in large-scale agile development (LSAD). While the roles appear helpful, it might not be suited for our study context as most software companies in Nigeria are SMEs. Even in the LSAD context, these new roles are just starting to emerge with the capacity required to improve security competence around cross-teams.

Secondly, the data collected in this study identified three major sources of cost when building secure agile software in Nigeria. These sources include the additional administrative costs of maintaining data centers, the cost of outsourcing security experts, and issues around the exchange rate of security goods and services. Some aspects of our results corroborate the findings of Sebega and Mnkandla (2017) who highlighted that non-functional requirements (i.e., security and safety) in secure ASD were not always considered due to costs. This leads to the potential compromise of poor software quality. However, in contrast to our findings, Venson et al. (2019) identified that conducting security reviews, applying threat modeling, and performing security testing were the three most significant sources of cost associated with building secure software. The study reported that the "security-by-design" paradigm was the lowest source of cost, as noted by only one study in the literature.

Thirdly, we discovered that the cybersecurity culture in participating organizations is self-taught where practitioners from our study learn and teach their colleagues. This finding aligns with earlier research by Bodin and Golberg (2021), where interested persons learn through reading blogs and watching videos. Our study further revealed that practitioners rely on two elements of cybersecurity culture to ensure adherence to regulatory policies. These elements include artifacts (awareness training and employee behavior or mindset) and values (security code of conduct or guidelines). There were practitioners in our study with very few opportunities to engage in security training who relied on senior engineers for security knowledge. They stated this was due to the cost which most companies—especially SMEs—in Nigeria could not afford. A study by Alshaikh (2020) advocated that practitioners move beyond just security education, training, and awareness (SETA). The study implemented the SETA approach in three Australian firms to show the transformation from compliance to building a cybersecurity culture. However, most research sites involved in this study have not reached the desired SETA compliance level.

Fourthly, we discovered that the knowledge gap and lack of necessary security skill sets were major challenges to securing ASD. Our findings corroborate the study by Nägele et al. (2022) which found that a lack of knowledge and experience in security governance was one of the challenges confronting practitioners developing secure software using agile methods. The lack of security knowledge in ASD has also contributed to increases in cybersecurity vulnerability (Jøsang et al., 2015). The study proposed a secure agile method that requires team members to have adequate cybersecurity training and education. The study views the integration of security modules in university curricula as more important than investing in sophisticated detection and filtering tools.

While our findings in this study are comparable to some of the existing literature discussed in this section, our PAC model presents valuable insights into practitioner challenges in a developing country context where existing empirical studies are lacking. Our developed theory explains how the challenges to adherence were discovered based on emergent empirical evidence.

6.3 | Limitations

As in any grounded theory research, this study is specific to its studied context—and would therefore be modified if data from a different context were collected (Adolph et al., 2008; Glaser, 1992). Transferability studies the applicability of the research findings to another context (Lincoln & Guba, 1985). While this study was conducted in Nigeria, the results may be transferable to similar contexts in the Sub-Saharan Africa region in countries like Ghana, Rwanda, and Tanzania. Nevertheless, the unit of analysis in this study is the agile practitioner and their perceptions of the challenges to NDPR adherence. Thus, the research sites do not contribute to the generalizability of our study.

To address the potential limitations of our study, we considered some of the principles of conducting interpretive research proposed by Klein and Myers (1999). To ensure the reliability of the data collected, we recruited the study participants from diverse business sectors, ranging from financial services, healthcare, manufacturing, and IT services and educational software solutions. Using the theoretical sampling technique of grounded theory, we interviewed participants from a variety of roles such as back-end engineer, senior DevOps engineer, and senior software engineer. We also included various management roles such as Chief Technology Officers, security managers and project managers to ensure multiple perspectives on the research phenomenon. While data collected through interviews may be subjective, interviewing a wide range of participants reduces bias (Diefenbach, 2009).

7 | CONCLUSION AND FUTURE WORK

Cyber attackers are continuously looking for ways to exploit vulnerabilities in software systems. The lack of security awareness, non-adherence to regulatory policies, and the existing knowledge gap of practitioners—especially in the global South—makes them vulnerable to attacks.

This study investigated the impact of the NDPR on agile practitioners and their secure development activities. The paper adopted a data analysis approach that was informed by a grounded theory method. Fifteen semi-structured interviews were conducted to understand the impact of the NDPR on security practices and activities. The research respondents were recruited using snowball and intensity sampling techniques.

Our research discovered tensions between the government and agile practitioners due to the introduction of a new security regulatory environment in the IT industry in Nigeria. Participants stressed the lack of indigenous software hosting companies for the type of applications they were developing. Also, neither the practitioners interviewed, nor their clients trust the capacities of existing hosting companies even though they are deemed good by the government. Instead, many clients prefer their applications to be hosted by offshore companies, especially those developing software for critical industries, such as the medical sector. While the NDPR discourages offshore software hosting especially of government data, we discovered that most participating companies in our study act contrary to the policy. Additionally, this study observed a gap between ASD and security teams. The findings therefore provide a potential pathway to mitigate the skills shortages among Nigerian agile practitioners. We have also provided empirical evidence of the sources of cost when developing secure software, the knowledge gap, and the level of compliance with cybersecurity culture.

In this article, we explored agile practitioners' perceptions of the NDPR when developing secure software. Our primary contribution is the PAC model we created using practitioner interview data. As such, we perceived the model and the grounded theory presented as especially valuable to researchers studying adherence to regulatory policies, and to practitioners wishing to evaluate their adherence process and practices in developing countries, specifically Nigeria.

In conclusion, we recommend that the government focuses more on building trust in the capacity of indigenous software hosting companies. By building trust, people will be encouraged to adhere to the regulations and keep their data locally. This study also advocates closer collaboration between agile and security teams. By using the collaborative capabilities of agile skills acquired by a few security specialists, these can be spread among other practitioners to help raise awareness.

Nevertheless, further research is required on how regulatory policies such as the NDPR can be more practicably tailored for use in a developing country context. Understanding the challenges will enable the government to create a more conducive environment for agile practitioners to develop more secure software systems.

ACKNOWLEDGMENTS

We are grateful to all interviewees for agreeing to participate in this study. This research is funded by the Petroleum Technology Development Fund (PTDF) Nigeria through the overseas scholarship scheme. Thanks also goes to the University of Salford Research Ethics Committee for reviewing and approving our ethics application – STR1920-32.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are openly available in figshare at https://doi.org/10.17866/rd.salford.21708029.v1.

ORCID

Abdulhamid A. Ardo https://orcid.org/0000-0002-0829-570X
Julian M. Bass https://orcid.org/0000-0002-0570-7086

REFERENCES

Adolph, S., Hall, W., & Kruchten, P. (2008). *A methodological leg to stand on: Lessons learned using grounded theory to study software development*. Proceedings of the 2008 Conference of the Center for Advanced Studies on Collaborative Research: Meeting of Minds, pp. 166–178.

Adolph, S., Hall, W., & Kruchten, P. (2011). Using grounded theory to study the experience of software development. *Empirical Software Engineering*, *16*(4), 487–513.

Agbali, M., Dahiru, A. A., Olufemi, G. D., Kashifu, I. A., & Vincent, O. (2020). Data privacy and protection: The role of regulation and implications for data controllers in developing countries. In *Information and communication technologies for development: 16th IFIP WG 9.4 International conference on social implications of computers in developing countries (ICT4D 2020), Manchester, UK, June 10 and 11, 2020, Proceedings 16* (pp. 205–216). Springer International Publishing.

Akinnuwesi, B. A., Uzoka, F. M., Olabiyisi, S. O., Omidiora, E. O., & Fiddi, P. (2013). An empirical analysis of end-user participation in software development projects in a developing country context. *The Electronic Journal of Information Systems in Developing Countries*, *58*(1), 1–25.

Alshaikh, M. (2020). Developing cybersecurity culture to influence employee behavior: A practice perspective. *Computers & Security*, *98*, 102003.

Ardo, A., Bass, J., & Gaber, T. (2022). *EJISDC paper 2022 – appendix*. University of Salford.

Ardo, A., Bass, J., & Tarek, G. (2022). Towards secure agile software development process: A practice-based model. In *48th Euromicro conference series on software engineering and advanced applications (SEAA), Maspalomas, Gran Canaria, Spain* (pp. 149–156). IEEE.

Baca, D., Boldt, M., Carlsson, B., & Jacobsson, A. (2015). A novel security-enhanced agile software development process applied in an industrial setting. In *2015 10th international conference on availability, reliability and security* (pp. 11–19). IEEE.

Bass, J. M. (2013). Agile method tailoring in distributed enterprises: Product owner teams. In *2013 IEEE 8th international conference on global software engineering* (pp. 154–163). IEEE.

Bass, J. M. (2016a). Artefacts and agile method tailoring in large-scale offshore software development programmes. *Information and Software Technology*, *75*, 1–16. https://doi.org/10.1016/j.infsof.2016.03.001

Bass, J. M. (2016b). *Large-scale offshore agile tailoring: Exploring product and service organisations*. Proceedings of the scientific workshop proceedings of XP2016 (pp. 1–5).

Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., & Jeffries, R. (2001). The agile manifesto.

Bodin, N., & Golberg, H. K. B. (2021). *Software security culture in development teams: An empirical study* (Master's thesis). NTNU.

Bunt, S. (2018). Critical realism and grounded theory: Analysing the adoption outcomes for disabled children using the retroduction framework. *Qualitative Social Work*, *17*(2), 176–194.

Canedo, E. D., Calazans, A. T. S., Cerqueira, A. J., Costa, P. H. T., & Masson, E. T. S. (2021). Agile teams. In *Perception in privacy requirements elicitation: LGPD's compliance in Brazil. 2021 IEEE 29th international requirements engineering conference (RE)* (pp. 58–69). IEEE.

Charmaz, K. (2000). Grounded theory: Objectivist and constructivist methods. In N. K. Denzin & Y. S. Lincoln (Eds.), *Handbook of qualitative research* (2nd ed.). Sage Publications.

Charmaz, K. (2006). *Constructing grounded theory: A practical guide through qualitative analysis*. Sage.

Charmaz, K. (2008). Constructionism and the grounded theory method. In J. A. Holstein & J. F. Gubrium (Eds.), *Handbook of constructionist research*. The Guilford Press.

Chika, D. M., & Tochukwu, E. S. (2020). An analysis of data protection and compliance in Nigeria. *International Journal of Research and Innovation in Social Science (IJRISS)*, *IV(V)*, 377–382.

Coleman, G., & O'Connor, R. (2007). Using grounded theory to understand software process improvement: A study of Irish software product companies. *Information and Software Technology*, *49*(6), 654–667.

Da Veiga, A., Astakhova, L. V., Botha, A., & Herselman, M. (2020). Defining organisational information security culture—perspectives from academia and industry. *Computers & Security*, *92*, 101713.

Diefenbach, T. (2009). Are case studies more than sophisticated storytelling?: Methodological problems of qualitative empirical research mainly based on semi-structured interviews. *Quality & Quantity*, *43*(6), 875–894.

Dybå, T., & Dingsøyr, T. (2008). Empirical studies of agile software development: A systematic review. *Information and Software Technology*, *50*(9–10), 833–859.

Edhlund, B., & McDougall, A. (2019). *NVivo 12 essentials*. Lulu Press, Inc. Lulu.com.

Egere, A. (2020). Cybersecurity atlas, Nigeria. (2020). *International Journal of Computer Science Trends and Technology*, *8*(6), 95–100.

Glaser, B., & Strauss, A. (1967). *The discovery of grounded theory*. Aldine Publishing Company.

Glaser, B. G. (1978). *Theoretical sensitivity: Advances in the methodology of grounded theory*. The Sociology Press.

Glaser, B. G. (1992). *Emergence vs. forcing: Basics of grounded theory analysis* (1st ed.). Sociology Press.

Glaser, B. G. (1998). *Doing grounded theory: Issues and discussions*. Sociology Press.

Glaser, B. G. (2001). *The grounded theory perspective: Conceptualization contrasted with description*. Sociology Press.

Glaser, B. G., Strauss, A. L., & Strutzel, E. (1968). The discovery of grounded theory: Strategies for qualitative research. *Nursing Research*, *17*(4), 364.

Gregor, S. (2006). The nature of theory in information systems. *MIS Quarterly*, *30*, 611–642.

Hoda, R., Noble, J., & Marshall, S. (2012a). Developing a grounded theory to explain the practices of self-organizing agile teams. *Empirical Software Engineering*, *17*(6), 609–639.

Hoda, R., Noble, J., & Marshall, S. (2012b). Self-organizing roles on agile software development teams. *IEEE Transactions on Software Engineering*, *39*(3), 422–444.

Jøsang, A., Ødegaard, M., & Oftedal, E. (2015). Cybersecurity through secure software development. In *Information Security Education Across the Curriculum: 9th IFIP WG 11.8 World Conference, WISE9, Hamburg, Germany, May 26–28, 2015, Proceedings 9* (pp. 53–63). Springer International Publishing.

Kempster, S., & Parry, K. W. (2011). Grounded theory and leadership research: A critical realist perspective. *The Leadership Quarterly*, *22*(1), 106–120.

Klein, H. K., & Myers, M. D. (1999). A set of principles for conducting and evaluating interpretive field studies in information systems. *MIS Quarterly*, *23*, 67–93.

Kvale, S., & Brinkmann, S. (2009). *Interviews: Learning the craft of qualitative research interviewing* (2nd ed.). Sage Publications.

Leite, L., dos Santos, D. R., & Almeida, F. (2021). The impact of general data protection regulation on software engineering practices. *Information & Computer Security*, *30*(1), 79–96.

Li, H., Yu, L., & He, W. (2019). The impact of GDPR on global technology development. *Journal of Global Information Technology Management*, *22*(1), 1–6. https://doi.org/10.1080/1097198X.2019.1569186

Lincoln, Y. S., & Guba, E. G. (1985). *Naturalistic inquiry* (1st ed.). Sage Publications.

Masood, Z., Hoda, R., & Blincoe, K. (2020). How agile teams make self assignment work: A grounded theory study. *Empirical Software Engineering*, *25*(6), 4962–5005.

McGhee, G., Marland, G. R., & Atkinson, J. (2007). Grounded theory research: Literature reviewing and reflexivity. *Journal of Advanced Nursing*, *60*(3), 334–342.

Miles, M. B., & Huberman, A. M. (1994). *Qualitative Data Analysis: An Expanded Sourcebook* (2nd ed.). Sage Publications.

Mohallel, A. A., & Bass, J. M. (2019). Agile software development practices in Egypt SMEs: A grounded theory investigation. Information and communication technologies for development. Strengthening southern-driven cooperation as a catalyst for ICT4D. In *15th IFIP WG 9.4 international conference on social implications of computers in developing countries, ICT4D 2019, Dar es Salaam, Tanzania, proceedings, part I 15* (pp. 355–365). Springer International Publishing.

Moyón, F., Almeida, P., Riofrío, D., Mendez, D., & Kalinowski, M. (2020). Security compliance in agile software development: A systematic mapping study. In *2020 46th Euromicro conference on software engineering and advanced applications (SEAA), Portoroz, Slovenia* (pp. 413–420). IEEE.

Nägele, S., Watzelt, J.-P., & Matthes, F. (2022). Investigating the current state of security in large-scale agile development. In *Agile processes in software engineering and extreme programming: 23rd international conference on agile software development, XP 2022, Copenhagen, Denmark, Proceedings* (pp. 203–219). Springer International Publishing.

Nerur, S., Mahapatra, R., & Mangalaraj, G. (2005). Challenges of migrating to agile methodologies. *Communications of the ACM*, *48*(5), 72–78.

Newton, N., Anslow, C., & Drechsler, A. (2019). Information security in agile software development projects: A critical success factor perspective. In *27th European conference on information systems (ECIS), Stockholm & Uppsala, Sweden*. Association for Information Systems.

Nigeria Bureau of Statistics: Nigerian Economy Largest in Africa (2019/2021). Nigeria Bureau of Statistics. https://nigerianstat.gov.ng/

Ogunyemi, A., Lamas, D., & Eze, E. (2018). Exploring the state of human-centred design practice in software development companies: A cross-case analysis of three nigerian software companies. *Interacting with Computers*, *30*(5), 444–467.

Oliver, C. (2012). Critical realist grounded theory: A new approach for social work research. *British Journal of Social Work*, *42*(2), 371–387.

Olukoya, O. (2022). Assessing frameworks for eliciting privacy & security requirements from laws and regulations. *Computers & Security*, *117*, 102697.

Parry, K. W. (1998). Grounded theory and social process: A new direction for leadership research. *The Leadership Quarterly*, *9*(1), 85–105.

Patton, M. Q. (2002). *Qualitative research & evaluation methods* (3rd ed.). Sage Publications, Inc.

Poritskiy, N., Oliveira, F., & Almeida, F. (2019). The benefits and challenges of general data protection regulation for the information technology sector. *Digital Policy, Regulation and Governance*, *21*(5), 510–524. https://doi.org/10.1108/DPRG-05-2019-0039

Presthus, W., Sørum, H., & Andersen, L. R. (2018). GDPR compliance in Norwegian companies. In *Norsk Konferanse for Organisasjoners Bruk av IT (NOKOBIT). Svalbard, Norway* (pp. 1–14). Semantic Scholar.

Rahy, S., & Bass, J. M. (2020). Implementation of agile methodology in developing countries: Case study in lebanon. In *Information and Communication Technologies for Development: 16th IFIP WG 9.4 International Conference on Social Implications of Computers in Developing Countries, Manchester, UK, June 10 and 11, 2020, Proceedings 16* (pp. 66–77). Springer International Publishing.

Rahy, S., Kreps, D., Bass, J. M., Gaber, T., & Ardo, A. (2020). A post-colonial analysis of agile software development methods in ICT4D. In *Information and Communication Technologies for Development: 16th IFIP WG 9.4 International Conference on Social Implications of Computers in Developing Countries, Manchester, UK, June 10 and 11, 2020, Proceedings 16* (pp. 66–77). Springer International Publishing.

Regassa, Z., Bass, J. M., & Midekso, D. (2017). Agile methods in Ethiopia: An empirical study. In *14th IFIP WG 9.4 international conference on social implications of computers in developing countries, ICT4D Yogyakarta, Indonesia, Peoceedings 14* (pp. 367–378). Springer International Publishing.

Riisom, K. R., Hubel, M. S., Alradhi, H. M., Nielsen, N. B., Kuusinen, K., & Jabangwe, R. (2018). Software security in agile software development: A literature review of challenges and solutions. In *Proceedings of the 19th international conference on agile software development: Companion, Porto, Portugal* (pp. 1–5). Association for Computing Machinery.

Rindell, K., Ruohonen, J., Holvitie, J., Hyrynsalmi, S., & Leppänen, V. (2021). Security in agile software development: A practitioner survey. *Information and Software Technology*, *131*, 106488.

Rindell, K., Ruohonen, J., & Hyrynsalmi, S. (2018). Surveying secure software development practices in Finland. In *Proceedings of the 13th international conference on availability, reliability and security, Hamburg, Germany* (pp. 1–7). Association for Computing Machinery.

Rygge, H., & Jøsang, A. (2018). Threat poker: Solving security and privacy threats in agile software development. In *Nordic conference on secure IT systems, Oslo, Norway, proceedings 23* (pp. 468–483). Springer International Publishing.

Sánchez-Gordón, M., & Colomo-Palacios, R. (2020). Security as culture: A systematic literature review of DevSecOps. In *Proceedings of the IEEE/ACM 42nd international conference on software engineering workshops* (pp. 266–269). Association for Computing Machinery.

Sebega, Y., & Mnkandla, E. (2017). Exploring issues in agile requirements engineering in the south African software industry. *The Electronic Journal of Information Systems in Developing Countries*, *81*(1), 1–18.

Sharma, A., & Bawa, R. (2020). Identification and integration of security activities for secure agile development. *International Journal of Information Technology*, *14*, 1–14.

Shastri, Y., Hoda, R., & Amor, R. (2021). The role of the project manager in agile software development projects. *Journal of Systems and Software*, *173*, 110871. https://doi.org/10.1016/j.jss.2020.110871

Sophos Group plc. (2021). The state of cloud security 2020. https://secure2.sophos.com/en-us/content/state-of-cloud-security.aspx

Soriyan, H. A., Mursu, A. S., Akinde, A. D., & Korpela, M. J. (2001). Information systems development in Nigerian software companies: Research methodology and assessment from the healthcare sector's perspective. *The Electronic Journal of Information Systems in Developing Countries*, *5*(1), 1–18.

Sowunmi, O. Y., Misra, S., Fernandez-Sanz, L., Crawford, B., & Soto, R. (2016). An empirical evaluation of software quality assurance practices and challenges in a developing country: A comparison of Nigeria and Turkey. *Springerplus*, *5*(1), 1–13.

Strauss, A., & Corbin, J. (1990). Basics of qualitative research. *Sage*, *2*, 272.

Stray, V., Sjøberg, D. I., & Dybå, T. (2016). The daily stand-up meeting: A grounded theory study. *Journal of Systems and Software*, *114*, 101–124.

Terpstra, E., Daneva, M., & Wang, C. (2017). Agile practitioners' understanding of security requirements: Insights from a grounded theory analysis. In *2017 IEEE 25th international requirements engineering conference workshops (REW)* (pp. 439–442). IEEE.

Tøndel, I. A., Cruzes, D. S., Jaatun, M. G., & Sindre, G. (2022). Influencing the security prioritisation of an agile software development project. *Computers & Security*, *118*, 102744.

Tøndel, I. A., Jaatun, M. G., Cruzes, D. S., & Williams, L. (2019). Collaborative security risk estimation in agile software development. *Information & Computer Security*, *27*(4), 508–535.

Tunggal, A. T. (2022). *What is the cost of a data breach in 2022?* UpGuard, Inc https://www.upguard.com/blog/cost-of-data-breach

Urquhart, C., Lehmann, H., & Myers, M. D. (2010). Putting the 'theory' back into grounded theory: Guidelines for grounded theory studies in information systems. *Information Systems Journal*, *20*(4), 357–381.

Usman, M., Felderer, M., Unterkalmsteiner, M., Klotins, E., Mendez, D., & Alégroth, E. (2020). Compliance requirements in large-scale software development: An industrial case study. In *Product-focused software process improvement: 21st international conference, PROFES 2020, Turin, Italy, November 25–27, 2020, Proceedings 21* (pp. 385–401). Springer International Publishing.

Vaivio, J. (2012). Interviews –learning the craft of qualitative research interviewing. *European Accounting Review*, *21*(1), 186–189. https://doi.org/10.1080/09638180.2012.675165

Venson, E., Guo, X., Yan, Z., & Boehm, B. (2019). Costing secure software development: A systematic mapping study. In *Proceedings of the 14th international conference on availability, reliability and security* (pp. 1–11). Association for Computing Machinery.

Zerlang, J. (2017). GDPR: A milestone in convergence for cyber-security and compliance. *Network Security*, *2017*(6), 8–11.

AUTHOR BIOGRAPHIES

Abdulhamid A. Ardo is a final year PhD student at the University of Salford, Manchester, UK. He received his MSc in Computer Science from Universiti Teknologi Malaysia, Johor Bahru, Malaysia, and BSc in Information Systems from American University of Nigeria. His experience includes teaching IT/Computing courses at the Federal University of Dutse, Nigeria from 2013 to 2019. From January 2020, he has been a Teaching Assistant at the University of Salford. He has also supervised over 20 M Sc Cybersecurity dissertation projects. Ardo is an Associate Fellow of the UK Higher Education Academy (AFHEA). He has published peer-reviewed articles and conference papers in IEEE Software, European, Mediterranean, and Middle Eastern Conference on Information Systems, (EMCIS), and IFIP WG 9.4 International Conference on Social Implications of Computers in Developing Countries. His research interests are in secure agile information systems development, ICT4D, cybersecurity and blockchain technology.

Julian M. Bass is a Professor of Software Engineering at the University of Salford, Manchester, UK. He is also a Senior Partner of Red Ocelot Ltd., a software consultancy associated with University of Salford. His research interests are in cloud-hosted software application architectures, agile information system development methods and ICT4D. He has published over 150 peer-reviewed articles and conference papers including in IEEE Software, Empirical Software Engineering and Information and Software Technology. His book "Agile Software Engineering Skills" is published by Springer. He has been a co-chair of the IFIP Working Group 9.4 conferences on the social implications of computers in developing countries. He is a Chartered Engineer, Fellow of BCS, the Chartered Institute of IT and a Distinguished Contributor to the IEEE Computer Society.

Tarek Gaber is a Senior Lecturer at the University of Salford, UK. He received his PhD in Computer Science (information security) from the University of Manchester in 2012. He has served as a co-chair and PC member in many international conferences and reviewed many scientific papers and participated in many scientific events (national/international conferences and workshops). He also served as Lead Guest Editor in many SCI international journals, including *Applied Sciences*, *Wireless Communications and Mobile Computing*, *Sustainability*, *and Electronics*. He has more than 90 publications in international journals, conferences, and book chapters. In addition, He has published five edited books and served as a keynote speaker at many international conferences. His major research interests include cybersecurity, machine learning, Internet of Things, and biometric authentication.

How to cite this article: Ardo, A. A., Bass, J. M., & Gaber, T. (2023). Implications of regulatory policy for building secure agile software in Nigeria: A grounded theory. *The Electronic Journal of Information Systems in Developing Countries*, e12285. https://doi.org/10.1002/isd2.12285