International Journal of Innovative Research in Computer Science and Technology (IJIRCST)
ISSN(Online): 2347-5552, Volume-12, Issue-2, March 2024
https://doi.org/10.55524/ijircst.2024.12.2.8
Article ID IRP1433, Pages 47-58
www.ijircst.org

A Comprehensive Review of the Significance of Low-code Automation in Risk Management for Banks

Deepa Ajish

IT Security and Compliance, ServiceNow Automation, Los Angeles, California, USA

Correspondence should be addressed to Deepa Ajish; deepajish@gmail.com

Received 26 February 2024; Revised 9 February 2024; Accepted 20 March 2024

ABSTRACT- This study provides a comprehensive examination of the influence of low-code development platforms in the risk management area within the banking sector. This research delves into the tactical utilization of low-code automation in the realm of risk management, with a particular emphasis on compliance monitoring, fraud detection, credit risk evaluation, and operational risk management. Additionally, the study centers on the metamorphosis of software design, development, and deployment processes, made possible by the easy accessibility of low-code platforms. The objective of the study is to illustrate how the digitization and automation of manual tasks in these domains augment service. provision, expedite the identification of compliance violations, and simplify credit risk evaluation, thereby conferring advantages to both banks and their clientele. Through meticulous analysis, this investigation seeks to furnish important perspectives on the uptake and influence of low-code development platforms in the banking sector, illuminating the potential for improved operational efficiency, regulatory adherence, and superior risk management.

KEYWORDS - Low-code, Risk management, Banking, Cybersecurity

I. INTRODUCTION

Low-Code Development Platform (LCDP) represents a paradigm shift in software development, characterized by the rapid creation of business applications with minimal manual coding. It employs user-friendly visual interfaces, pre-configured templates, and integrated components to accelerate the development cycle [1], [2]. LCDPs emerge as an apt technological solution for the implementation of the Citizen Development (CDE) concept [3]. LCDPs, with their simplified and user-friendly interfaces [2], empower non-technical users to create applications, thereby democratizing the application development process. This aligns with the ethos of CDE, which advocates for the active participation of end-users in software development. Citizen Developers (CDs), as they are commonly referred to, represent a new paradigm in the realm of application and workflow development. These end users, despite lacking a formal programming background, are anticipated to have the capability to develop applications and workflows for themselves or others. This is achieved without the direct involvement of professional developers, thereby democratizing the process of software development [4]. The emergence of CDs underscores a shift in the traditional developer-user dynamic, potentially leading to increased efficiency, customization, and user satisfaction. The banking sector faces numerous obstacles in the realm of risk management. It is incumbent upon banks to adhere to intricate regulatory mandates, fulfill escalating client demands, and maintain effective procedures. In an effort to optimize their functions, banks are perpetually in pursuit of groundbreaking solutions. Risk management in banking is a pivotal operation in the financial sector that guarantees the solidity, safety, and adherence of banking procedures. The regulatory framework supervising the banking sector has grown progressively complex in recent times. Financial establishments are obligated to conform to a multitude of rules and reporting stipulations enforced by government entities and regulatory agencies. These regulations, including Basel III, Dodd-Frank, and Anti-Money Laundering (AML) directives, are perpetually evolving, thereby posing a challenge for banks to remain compliant.

In the banking sector, risk management is facilitated through the execution of structured frameworks encompassing policies, procedures, and controls. This necessitates meticulous planning, consistent reviews, and timely updates. To surmount the hurdles associated with implementation, banks have turned to the adoption of frameworks that align with industry standards [5]. A research conducted by McKinsey & Company indicates that by the year 2025, the risk functions within banks are projected to undergo substantial transformations [6]. The study emphasizes that the progression of technology and the advancement of analytics are paving the way for the development of novel products, services, and techniques for managing risk. Furthermore, it advocates that banks should promptly commence a series of initiatives that harmonize the urgency of short-term business objectives with the facilitation of long-term strategic goals [6]. A significant number of banks depend on manual and paper-oriented methods within conventional risk management procedures. These antiquated approaches frequently entail substantial paperwork, manual data input, and compartmentalized systems, resulting in inefficiencies and potential inaccuracies. Manual procedures are labor-intensive, susceptible to human error, and can obstruct the prompt identification and response to risks. By employing innovative tools and platforms driven by technology,

banks can proactively detect and alleviate risks, augment compliance surveillance, and foster a strong risk culture within their organizations. Low-code automation (LCA) could serve as an optimal solution as it offers several persuasive advantages to bank risk management procedures.Within the banking sector, particularly in the realm of risk management, LCA is of considerable importance. Banks are intricate institutions that grapple with a variety of risks such as credit, market, operational, and liquidity risks. Effective management of these risks is vital for preserving financial stability and ensuring adherence to regulatory standards. As per a report by Appian [7], in the realm of financial services, a significant majority of leading firms have demonstrated a preference for low-code platforms as a means to streamline their automation processes. Specifically, 81% of these industry frontrunners report that their adoption of low-code platforms places them ahead, or significantly ahead, of their competition [7]. According to a forecast by Gartner [8], by the year 2024, Low-Code/No-Code (LCNC) development is projected to constitute 65% of all application development. This indicates a heightened level of interest and investment in this technology by banking institutions. By automating routine tasks, minimizing human errors, and offering real-time risk evaluations, LCA can simplify risk management procedures in banks. This leads to enhanced efficiency in risk detection, evaluation, and mitigation strategies. Moreover, LCA can improve regulatory reporting by automating the processes of data gathering and report creation, thereby enhancing precision and promptness.Low-code automation, characterized by its efficiency and cost reduction capabilities, has been increasingly adopted in the banking sector. However, it is not without its challenges. The implementation of low-code automation can potentially introduce security risks and necessitates a certain level of coding knowledge. Therefore, banks must exercise due diligence and consider these factors carefully when integrating low-code automation into their operations. Despite these challenges, the benefits of low-code automation, when properly managed, can significantly enhance the operational efficiency of banks. Thus, it is imperative for banks to strike a balance between leveraging the advantages of low-code automation and mitigating its potential risks. This will ensure that they can fully harness the potential of low-code automation while maintaining robust security measures and upskilling their workforce as necessary.The purpose of this study is to enhance the current understanding and promote a more profound understanding of the potential utilization of low-code automation for risk mitigation in the banking industry. This study will explore the benefits and concerns of implementing low-code automation in the field of risk management within the banking sector. In doing so, it aims to offer a balanced viewpoint on the transformative capacity of low-code automation and its ramifications for the future of banking and financial services.

A. Background of low-code development platforms

- ***Low-code development platform***

A LCDP is a software development environment that provides a graphical user interface for programming, thereby significantly reducing the need for manual coding [9]. LCDPs are designed to expedite the application development process by offering pre-built functions and features that can be implemented through simple drag-and-drop operations [9], [10]. LCDPs democratize the field of software development by enabling "citizen developers"—individuals without extensive programming knowledge—to create functional applications. This is particularly beneficial in business environments, where such platforms can be used by non-technical staff to develop applications tailored to specific operational needs [4]. Moreover, LCDPs are not solely beneficial to non-technical users. Experienced developers can also leverage these platforms to rapidly prototype and deploy applications, thereby increasing their productivity.The incorporation of Artificial Intelligence (AI) and Machine Learning (ML) into LCDPs is revolutionizing the field of software development. It not only enhances the capabilities of these platforms but also empowers both technical and non-technical users to create sophisticated applications with ease [2], [11].However, it is important to note that while LCDPs offer numerous benefits, they may not be suitable for all types of application development, particularly those requiring complex, custom functionality. Therefore, the choice to use an LCDP should be made in the context of the specific requirements and constraints.In conclusion, LCDPs represent a significant advancement in the field of software development, offering a more accessible and efficient approach to application creation. Their potential to democratize software development and increase productivity makes them a valuable tool in today's fast-paced digital world [12].

B. History of low-code development platforms

LCDPs have a rich history, with their early forms originated in the early 2000s, coinciding with the emergence of rapid application development (RAD) platforms [13], [14]. RAD tools such as Visual Basic, Visual Café, Delphi, and Oracle Forms entered the scene promising faster application development without extensive code writing. The primary objective of these platforms was to facilitate a more visual and intuitive approach to application development, thereby minimizing the necessity for extensive coding. This approach enabled users to expedite the application development process significantly. In the year 2014, the term "Low-code" was introduced by Forrester to categorize those development platforms that prioritized simplicity in development and user-friendliness [15]. These platforms were distinguished by their focus on reducing the complexity traditionally associated with the development process.

C. Low-code development platforms

The LCDP market is characterized by the presence of a multitude of vendors. The following are some of the principal participants:

- ***OutSystems:***

OutSystems, recognized as a pioneer in the LCDP market, has designed its platform to facilitate developers in the efficient and swift delivery of applications, while ensuring they meet enterprise-grade standards. The platform leverages AI and ML to provide recommendations, automate processes, and validate the developed application. This integration of AI and ML significantly

enhances the quality of the delivered application and accelerates the development process [16]. OutSystems has been recognized as a leader in the 2023 Gartner Magic Quadrant for Enterprise Low-Code Application Platforms [17] and scored as the leading low-code development platform on current offering and strategy by Forrester [18].

- ***Appian:***

Appian, another potent LCDP, markedly boosts productivity by enabling the development of significant applications with continuous innovation. It is carefully engineered from its core for peak performance. Appian is acknowledged as one of the foremost authorities and distinguished leaders in the realm of low-code platforms and automation [19], [20]. The LCDP offered by Appian is designed to address intricate business processes and applications that necessitate advanced levels of automation and analysis. In the 2023 Gartner Magic Quadrant for enterprise low-code application platforms, Appian has been acknowledged as a leader [21]. Appian provides a variety of low-code design objects, functions, and smart services you can use to easily integrate artificial intelligence (AI) and machine learning (ML) functionality into your application [22].

- ***Salesforce:***

Salesforce provides a renowned cloud-based LCDP. It is recognized as one of the largest low-code platforms, with a primary focus on customer-related applications concept [3]. Utilizing the lightning framework, Salesforce enables the creation of applications swiftly and cost-effectively. Furthermore, Salesforce's components are designed for easy reuse, ensuring customization is not compromised. The platform offers a range of AI-powered features and ML tools that can be used to build custom AI models [23].

- ***Microsoft power apps:***

Microsoft power apps is a prominent player in the field of low-code development [19]. It is part of the Microsoft Power Platform, a suite of apps, services, connectors, and a data platform [24]. The platform is designed to address the growing need for top-of-the-line internal workflow applications, timesaving automations, better customer experiences, and seamless integrations. It helps eliminate repetitive tasks by automating time-intensive and complicated development areas [25].The platform provides a suite of AI and automation tools that are embedded in low-code platforms [26]. Infusing AI into solutions can improve processes in many ways, including efficiency, accuracy, and ease of use. Their AI builder feature allows model makers to connect their AI models into AI builder, making them available to power platform makers within the organization [27].

- ***ServiceNow:***

ServiceNow is esteemed as a significant contributor in the domain of LCDPs. It has gained recognition for its solutions in IT service management and has made substantial investments in its low-code tools and its emphasis on 'workflow platform' communication [28]. The low-code application development platform of ServiceNow, referred to as the now platform app engine, empowers organizations to construct custom applications for both desktop and mobile that can be utilized to automate and digitize workflows [29]. Developers across the spectrum, from novices to professionals, can expedite the deployment of new digital initiatives, achieving scalability and delivering enhanced value. AI capabilities in the ServiceNow platform offers to deliver relevant information, make predictions and recommendations, and automate repetitive tasks [30]. It also offers ML frameworks, natural language understanding, search and automation, and analytics and process mining [30].

II. LITERATURE REVIEW

The primary challenge faced during the literature review was the notable scarcity of existing research and the lack of relevant academic papers. This deficiency of scholarly resources considerably impeded a thorough investigation of the topic at hand. It highlights the imperative need for additional research and academic discourse in this specific area of study.An academic investigation conducted by ISACA delves into the realm of cybersecurity and technological risk within the context of virtual banking [31]. The study underscores the significant challenge that virtual banks face in managing cybersecurity and technological risks [31]. It proposes that these banks should implement a purpose-built risk management strategy that strikes a balance between the ease of use and accessibility of digital platforms and mobile applications, and the imperative for data protection, cybersecurity safeguards, and a robust IT infrastructure [31]. An article by McKinsey [32] elaborates on the adoption of a risk-based strategy in addressing technology and cyber risks within the banking sector. The article indicates a growing trend among financial institutions towards utilizing a risk-based approach to ascertain their control priorities [32]. It further underscores the necessity for explicit, quantifiable declarations concerning technology risk and cyber risk appetite, articulated in commercial terminology, with unequivocal ownership [32].Reference [33] conducted an examination of cybersecurity disclosures made by the 48 preeminent Canadian and US banks spanning the years 2014 to 2020. The study underscores that the banking sector has been a principal target for cyberattacks, attributable to the sensitive data it possesses [33].A report jointly published by the World Economic Forum and Deloitte unveils that systemic risk, stemming from the utilization of technology, is a principal apprehension for leaders in the financial services industry [34]. Concurrently, the report highlights that technology possesses the potential to alleviate systemic risk [34]. According to some researchers, the most complex part of the development process is the time spent by programmers to write the code. This approach is considered to be time-consuming [11]. Additionally, the developer must possess good experience and knowledge in development activities and programming languages to develop an application [11], [2], [19].LCDP facilitate the fast development and delivery of applications [15].These platforms not only benefit professional developers, but also enable beginners to develop applications without requiring experience or knowledge of programming languages or complex engineering activities [11]. Reference [35] suggests that the code generated by LCDP may be difficult to comprehend due to the absence of comments, vague variable names, and may have security issues. However,

Sachis argues that using LCDP can safeguard privacy since the end-user or organization's employees will develop the application instead of outsourcing the work [15].According to the ABA Banking Journal, banks use a wide range of applications to manage customer data, financial transactions, and regulatory compliance [36]. Low-code platforms can support many use cases in addition to application development, such as enabling and enhancing automation of workflows, process optimization, digital banking, and data analytics [36]. In response to the increasing demand for digital banking services and the need to develop and deploy new digital applications quickly and efficiently, low-code application development platforms have emerged as a promising solution [36]. This approach not only reduces the time and cost of development but also allows for the introduction of new business functionalities at shorter intervals [37]. As a result, banks have been able to expedite the delivery of new products and services and upgrade existing offerings to meet evolving customer demands.

III. METHODOLOGY

The objective of this research is to investigate the significance of low-code automation in risk management within the financial industry and other related sectors. In order to achieve this objective, a SLR was conducted. The SLR aimed to identify the key areas of focus within the domain. The process was rigorous and methodical, ensuring that the derived research was both comprehensive and reliable. The SLR began with the development of a review protocol, which outlined the research objectives and the criteria for inclusion and exclusion of literature. A systematic search strategy was then employed, encompassing not only electronic literature databases but also manual searches and the identification of yet-to-be-published literature. This ensured a comprehensive overview of the research topic. The identified studies were evaluated for quality, and relevant data was extracted. This data was then analyzed and synthesized to answer the research question about the impact of low-code automation in risk management within the financial industry.In conclusion, the SLR provided valuable insights into the domain and identifying areas for future research. The rigorous and systematic approach of the SLR ensured the reliability and comprehensiveness of this research, making it a valuable resource for further studies in this field.

A. Data Collection

To ensure the relevancy and comprehensiveness of the collected data, a systematic and targeted approach was used for the data collection process. A comprehensive search of the databases was performed using specific keywords such as "Low-code", "LCDP", "Banking", "Bank", and "Risk management". These keywords were used in combination with the logical operators 'AND' and 'OR' to refine the search results. Specifically, the search string was constructed as follows: "Low-code" AND "Banking" OR "Bank" AND "Risk management". This search string was designed to retrieve articles that discuss the impact of low-code development platforms in banking industries. The search was not limited to the body of the articles but also included the article titles and keywords. This ensured a wide coverage of relevant literature, capturing articles where the main focus was on the chosen topic.

B. Data Processing

Due to the nature of the data extraction process, it was recognized that the extracted data might contain impurities, and not all data obtained using the specified keywords would be relevant to the research objectives. To mitigate this, a manual sorting process was implemented post-extraction. This involved a comprehensive review of the extracted data to identify and eliminate any irrelevant or impure data. The sorting process was guided by the research objectives and the predefined inclusion and exclusion criteria, ensuring that only data pertinent to the research was retained for further analysis.

IV. RESULTS AND DISCUSSION

Risk management is a crucial operation in the financial sector that ensures the stability, security, and compliance of banking procedures [38]. This section provides an in-depth analysis of the importance of low-code automation in risk management in the banking sector.

- ***Benefits of Low-Code in Risk Management in Banking***

Low-code development accelerates the delivery of new functions and capabilities to meet changing regulatory requirements. AI-generated low-code within an AI-powered process platform provides strong protections for digital solutions while preserving the privacy and security of the data that powers those solutions. Enterprise-grade platforms enforce authorization, governance, and data privacy, specifying who can modify or use artifacts and managing the modularity of functional capabilities and data privacy settings. Low-code platforms empower non-technical users to architect and construct software solutions [19]. Low-code applications can be seamlessly integrated with databases and application programming interfaces (API). Owing to the premade connectors, the integration process of low-code applications consumes significantly less time compared to traditional coded integrations.These are the analyzed areas in risk management where low-code automation can be effectively deployed.

A. Compliance Monitoring

Compliance monitoring is an integral component of banking operations [39]. It entails the procedure of guaranteeing that banks comply with regulatory mandates and internal policies. According to the Basel Committee, non-compliance risk refers to the possibility of facing legal or regulatory penalties, significant financial losses, or reputational damage due to a bank's inability to adhere to the compliance laws, rules and standards that govern its banking operations [40]. A report by [41] indicates that compliance risk has emerged as one of the most substantial ongoing apprehensions for executives in financial institutions. The report advocates that banks need to metamorphose the role of their compliance departments from an advisory capacity to one that underscores active risk management and monitoring. The regulatory compliance process comprises four categories:

Identification, assessment, monitoring, and reporting. By utilizing low-code platforms and capitalizing on workflows, banks can automate and digitize manual procedures to automate workflows and approvals, and enhance service delivery. Figure 1 depicts a high level flow of risk monitoring using vulnerability scanners like Qualys, Rapid7, etc. and ServiceNow. Automating the collection, analysis, and reporting of compliance data will guarantee the timely identification of any compliance violations and ensure compliance with regulatory requirements. In the context of compliance monitoring, low-code automation significantly impacts several components.

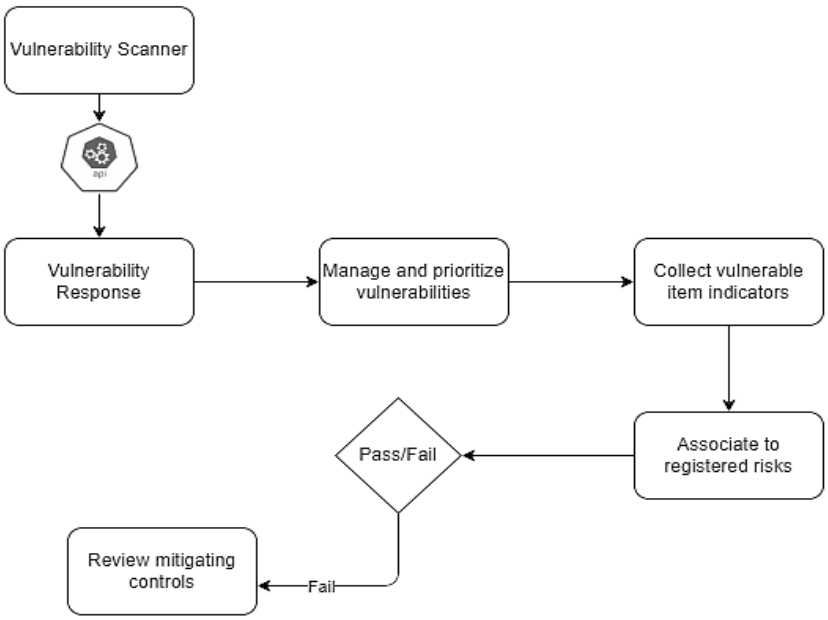

Figure 1: A high level diagram of risk monitoring

- ***Automated workflows***

Low-code compliance automation software provide compliance-related processes such as self-evaluation, corrective action design, and control examination and verification. Low-code tools employ artificial intelligence, advanced analytics, and machine learning techniques to safeguard and encrypt specific data categories according to regulatory norms. They also offer suggestions for resolving discrepancies and mitigating hazards.

- ***Simplified evidence collection***

Low-code enables easy evidence collection across multiple frameworks via user-friendly integrations. Top compliance software provide built-in content for common standards, such as General Data Protection Regulation (GDPR), PCI DSS (Payment Card Industry Data Security Standard), and National Institute for Standards and Technology (NIST). Compliance software can accommodate local, national, and international standards, facilitating quick and easy data collection, monitoring, and assessment.

- ***Continuous monitoring***

Low-code compliance monitoring software facilitate continuous monitoring and alerting mechanisms that enhance the company's overall security posture. The automation liberates the compliance team from trivial and redundant tasks, allowing them to concentrate on more complex and intensive compliance tasks such as creating new policies.

- ***Real-time data***

Low-code automation provides internal auditors and senior leadership with timely and precise information on compliance status. The compliance automation software sends automated alerts of emerging changes to auditors and leadership, enabling them to adjust internal policies and tasks accordingly to prevent non-compliance.

B. Fraud Detection

Fraud prevention describes the security measures to avoid unauthorized individuals from initiating transactions on an account for which they are not authorized [42]. Banks can leverage low-code applications to automate the process of detecting and preventing fraudulent activities by integrating data sources, instituting rule-based engines, and capitalizing on machine learning algorithms. By integrating data sources, banks can gain a comprehensive view of their customers' transactions, which can help them identify suspicious activities. Rule-based engines can be used to identify specific patterns or behaviors that suggest fraudulent activities. For example, a rule-based engine can be used to identify transactions that deviate from a customer's typical spending patterns or that occur in unusual locations. Machine learning algorithms can be used to identify patterns and trends in customer transactions that are not easily detectable by rule-based engines.

- ***Fraud monitoring***

Low-code allows for the detection of various types of fraud by setting up a new analytics model or modifying an

existing one, rather than relying on fixed solutions that are limited to specific fraud types.

- ***Data analytics***

Low-code automation can be used to identify patterns and trends in customer transactions, enabling banks to detect fraudulent activities more quickly.

- ***Easy integrations***

With the assistance of user-friendly integrations, low-code fraud monitoring models can access data from any of the data sources in the organization. This enables automation of redundant tasks, data verification, alert creation, and accuracy enhancement.

- ***Automated KYC and AML processes***

Low-code obviates the need for data migration to a new proprietary data model. Data can be accessed in its original format and transferred to the platform for users and AI to execute processes to accomplish tasks. It can identify financial crimes by conducting sanctioned screenings, reviewing suspicious activity reports, receiving alerts from detection engines for transactions/processing, and evaluating customer risk assessments. Low-code can connect with existing systems and data sources, facilitating smooth data communication and avoiding manual data input and reconciliation.

C. Credit Risk Assessment

Financial credit risk assessment is a crucial process that involves evaluating the creditworthiness of customers or assessing the likelihood of potential business failure [43]. The credit appraisal process can be expedited and made more precise by automating the collection, analysis, and scoring of data. Low-code applications, which enable banks and financial institutions to leverage cutting-edge technologies, can aid in the accurate assessment of credit risk and facilitate informed lending decisions. By reducing the need for manual intervention, these platforms can help banks process more applications in less time, thereby improving their operational efficiency. Figure 2 exemplifies the application of low-code automation in the domain of credit risk assessment. Low-code platforms provide the flexibility to continuously improve and update the credit risk assessment process. As new risks emerge or as regulations change, banks can quickly update their assessment process to reflect these changes.

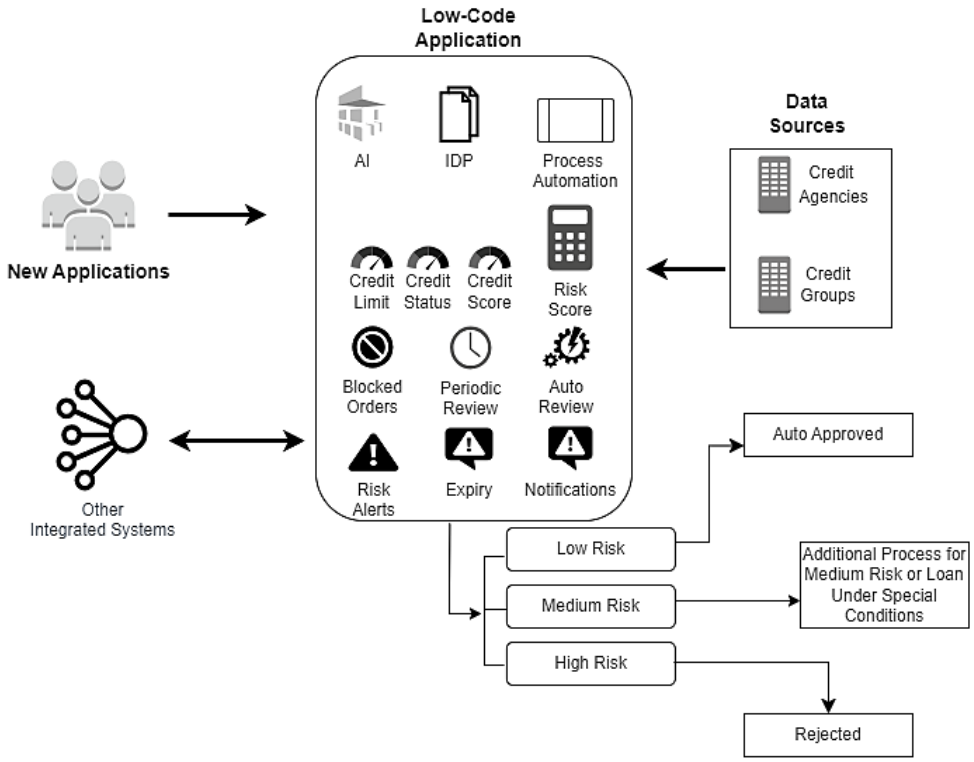

Figure 2: Low-code automation for credit risk assessment

- ***Automated credit risk workflows***

Low-code application platforms provide a seamless environment for designing and implementing workflows that enhance efficiency and reduce manual intervention. By leveraging low-code development, financial institutions can achieve greater agility in managing credit risk processes, ultimately leading to improved decision-making and operational effectiveness.

- ***Online credit application***

Low-code can simplify the credit application process by using pre-built templates tailored by sector or area, adaptable to suit the specific needs. It allows document uploads, trade reference handling and job-level credit applications, effectively collecting prospect data.

- ***Automated credit assessments***

By utilizing pre-built and customizable credit risk scoring algorithms, institutions can expedite the evaluation process. These algorithms allow quick analysis of creditworthiness. Low-code systems enhance decision-making by providing real-time risk alerts. This is particularly valuable for assessing low-risk customers promptly. Low-code tools facilitate streamlined handling of complex customer structures. The automated credit risk assessment process relies on predefined rules and algorithms, ensuring consistency. Human bias is minimized, as all applications are evaluated against the same criteria. Speeding up credit risk assessments not only improves operational efficiency but also enhances the overall customer experience. Quicker turnaround times and transparent assessment procedures benefit applicants.

- ***Prioritized credit worklists***

By leveraging intelligent algorithms, organizations can enhance their credit processes. These algorithms prioritize customers based on factors such as blocked orders, new customer onboarding, risk alerts, periodic reviews, and collateral expiration.

- ***Fast decisions***

In the realm of credit institutions, the adoption of low-code automation platform has facilitated swift responses to dynamic changes and accelerated digitization efforts. By incorporating technologies such as Robotic Process Automation (RPA), AI, Intelligent Document Processing (IDP), and seamless data integrations within a low-code development environment, banks and financial institutions empower transformative shifts. These advancements enable institutions to adapt to evolving circumstances, meet growing demands, and streamline manual and intricate processes.

D. Operational Risk Management

Operational risk management is a critical component of banking operations [44] that involves identifying, assessing, mitigating, and managing risks throughout the infrastructure. To proactively identify and manage operational risks in banking institutions, automated workflows for incident reporting, risk assessment, and mitigation can be implemented. Figure 2 shows a high-level diagram of automatic security incident response using Splunk as the Security information and event management (SIEM) tool and ServiceNow. Low-code platforms enable banks to quickly build and launch risk management applications that can assist in the timely identification and mitigation of potential risks.

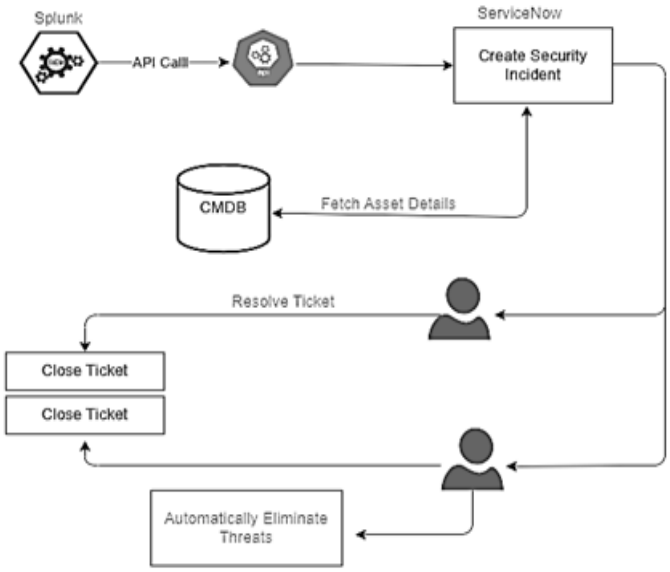

Figure 3: Automatic security incident response

A Security Orchestration, Automation, and Response (SOAR) system allows cybersecurity and IT teams to collaborate more effectively in addressing overall network environment. By integrating internal data and external threat information, SOAR helps identify the underlying issues in each security situation. Its automation features distinguish it from other security systems, streamlining processes and reducing manual effort. Security automation within SOAR can handle tasks like managing user access and query logs etc. Additionally, SOAR serves as an orchestration solution, automating tasks that would typically require multiple security tools. Both orchestration and automation form the foundation for the response capability in a SOAR system. Low-code automation can be efficiently implemented in the below areas:

- ***Automatic playbooks***

Low-code SOAR solutions empower the development of playbooks for automating a wide spectrum of repetitive tasks that would otherwise require manual execution by analysts. These SOAR playbooks consist of structured workflows that lead security teams through various procedures in an automated or semi-automated fashion. By doing so, they mitigate false positive alerts, alleviate alert fatigue and analyst burnout, and enable security teams to concentrate on more profound detection, analysis, strategic planning, and critical business functions .

- ***Automatic Incident Response***

Many organizations still handle critical incident procedures manually, relying on phone calls, emails, spreadsheets, and even paper forms. Low-code automation allows teams to build, implement and accelerate automated incident response workflows around repetitive tasks. Low-code automation enables organizations to maintain control and visibility before, during, and after incidents to respond effectively, while limiting risks and safety issues. It enforces standards and orchestrates response procedures through automated notifications, actions, and resolutions.

- ***Phishing detection and investigation***

Phishing emails are among the oldest and most common types of cyberattacks. In a low-code automated system, a potential phishing email triggers a specific workflow pattern to thwart the attack before it fully materializes. For instance, suspicious emails are promptly deleted from an inbox to prevent users from opening them. Additionally, the most time-consuming aspects of an investigation can be automated by extracting indicators of compromise (IOCs) from various parts of an email, including headers, body (HTML, text, RTF), sender, and subject. The reputation of each mail transport agent (MTA) within the "Received From" headers is also considered. Once the reputation is determined and IOCs are extracted, security teams can further automate correlation and threat intelligence lookups for artifacts such as IPv4, IPv6, URLs, file hashes (MD5, Sha1, SHA256, SHA512, SSDeep), and domains [45].

- ***Vulnerability management and asset discovery***

Low-code security automation assists the organization with better tracking of assets and risk management. It provides full lifecycle management to continuously identify risks related to unpatched, misconfigured, and unknown systems within an entity. Endless integrations with vulnerability management and patching tools streamline preexisting processes with automation. It allows to build gated processes and workflows into a vulnerability management program. The powerful workflows can be easily customized to meet any use case or business processes in use now or in the future.

- ***Alert Triage***

SIEM systems possess the capability to collect, aggregate, store, and correlate events originating from a managed infrastructure [46]. SIEM systems concentrate on handling security-related information and events [47]. Event management entails detecting potential security issues, such as suspicious user activity, potential malware or ransomware incidents, and Distributed Denial of Service (DDoS) attacks [47]. Organizations can seamlessly integrate with various SIEM, Endpoint Detection and Response (EDR), and Extended Detection and Response (XDR) platforms using low-code application platforms. By automating responses to potential security threats, organizations can mitigate alert fatigue, allowing security analysts to concentrate on authentic and critical security risks.

V. DISCUSSION

The author analyses the findings of this study to elucidate the importance of low-code automation within the realm of risk management in the banking sector. The advantages of low-code, particularly in relation to the diminution of complexity and the enhancement of agility in the developmental process, serve as catalysts for the digitalization process. In the academic context, numerous organizations continue to utilize discrete and non-real-time systems for data processing, with a significant reliance on spreadsheets [48]. For the extraction of insights and valuable information within the risk management domain, it is imperative that data undergoes appropriate processing. This procedure, when executed through discrete and manual methods, is not only time-intensive but also susceptible to inaccuracies. It has been observed that the substantial effort necessitated for the capture and processing of data from discrete sources impedes real-time risk mitigation endeavors. In this regard, low-code plays a pivotal role in facilitating the transition from raw data to actionable information, thereby enabling respective teams to swiftly access pertinent information and respond accordingly.Although LCDPs are intended to be intuitive, the CDs still require knowledge about data modeling and requirements management to deliver an application. As per my experience, along with understanding how to use the low-code platform, knowledge about data modeling like usage of tables, data relationships, and requirements engineering are required for optimal output.Low-code development platforms are designed to provide simplicity and ease of use, but they have limitations when it comes to scalability. While they offer a broad range of pre-existing templates, they can be limiting for more complex solutions and don't always offer the ability to scale along with a business's growth and changing needs [49], [50]. Therefore, it is important to evaluate the scalability of low-code platforms before choosing one [51].There are two types of scalability to consider when evaluating low-code platforms: runtime scalability and platform scalability [51]. Runtime scalability refers to the ability to increase the capacity of deployed applications and provide fast user experiences for large numbers of users and compute-intensive operations [51]. Platform scalability refers to the ability to build and manage large, interconnected portfolios of applications that address multiple use cases simultaneously all the time or when needed [51].Ultimately, the value of low-code automation in risk management lies in its ability to streamline processes, enhance real-time risk mitigation efforts, and support the overall growth and digital transformation of the banking industry. However, its successful implementation requires a nuanced understanding of the platform's capabilities and limitations.

A. Recommendations

In order to ensure that low-code development aligns with organizational structure, banking industry norms, and an institution's policies and objectives, it is recommended that appropriate policies and procedures be established, similar to those used in conventional code development. This serves as a protective measure that ensures low-code development remains on the intended trajectory, retains control over the process, diminishes risks, and protects the

integrity and security of applications.The below standards are highly recommended for low-code application development in the banking industry.

- ***Compliance requirements in low-code***

In order to ensure that low-code applications align with banking industry benchmarks, it is imperative to comply with data privacy legislations such as the GDPR and the Financial Privacy Rule. Furthermore, low-code applications should satisfy data protection and privacy norms in accordance with the policies and stipulations of the organization. The financial sector is subject to specific regulations that influence software development. By complying with these regulations, banks can ensure that customer information is properly protected and that the privacy of customers is respected.By establishing policies and procedures, banks can ensure that low-code development is conducted in a manner that is consistent with industry standards and best practices, while also ensuring that the development process is transparent and accountable.

- ***Security risks and data protection***

In the realm of software development, security emerges as a preeminent concern. Crucial security considerations encompass:

a) Threat Mitigation

Low-code applications ought to provide protection against data breaches, cyberattacks, and unauthorized access.

b) Data Encryption

Guaranteeing that data is encrypted, both in a dormant state and during transmission, is a fundamental aspect of application security.

c) Authentication and Authorization

Sturdy authentication and authorization mechanisms should be established, ensuring that only authorized users gain access to sensitive data and functionalities.

B. Testing and Validation

When delivering dependable and high-performing applications or solutions via low-code development, it is essential to establish standards for comprehensive quality assurance testing. This requires aligning testing strategies with the development methodology and ensuring that applications meet user expectations and conform to industry benchmarks. The process of validation is indispensable to ensure that low-code applications achieve their designated objectives.By establishing testing standards, banks can ensure that low-code applications are developed in a manner that is consistent with industry standards and best practices. Testing strategies should be aligned with the development methodology to ensure that applications are tested thoroughly and that all potential issues are identified and addressed. The process of validation is essential to ensure that low-code applications meet user expectations and achieve their designated objectives.By implementing comprehensive quality assurance testing, banks can ensure that low-code applications are reliable, high-performing, and meet the needs of their customers.

C. Application Permissions

Role-Based Access Control (RBAC) is a widely adopted approach for administering permissions in the sphere of low-code development. By defining roles and responsibilities within the development team and ensuring that developers have the necessary access based on their roles [52], the risk associated with unauthorized modifications or data breaches can be significantly reduced. This strategy highlights the importance of structured access control in maintaining the integrity and security of the development process.By implementing RBAC, banks can ensure that low-code applications are developed in a manner that is consistent with industry standards and best practices. RBAC provides a framework for managing permissions that is easy to understand and implement. By defining roles and responsibilities, banks can ensure that developers have the necessary access to perform their jobs while also ensuring that the development process is transparent and accountable.RBAC is an essential component of the low-code development process and should be given the same level of attention as other aspects of the development process.

VI. FUTURE PERSPECTIVE

Low-code development platforms have gained significant traction in recent years, and their popularity is expected to continue to grow in the future. These platforms are already being used by businesses of all sizes to build custom applications quickly and cost-effectively. As the demand for custom software continues to grow, it's likely that low-code platforms will become even more popular.In the future, low-code platforms will become widespread in the application development industry and change the whole tech market. People will use low-code technology for both customer-oriented and middle- and back-office processes [53]. Several key trends are shaping the future of low-code development, driven by technological advancements, changing user demographics, and the increasing demand for rapid application delivery [54].By leveraging low-code platforms, banks can streamline their software development process, reduce costs, and improve their operational efficiency. Low-code platforms provide the flexibility to continuously improve and update the software development process. As new risks emerge or as regulations change, banks can quickly update their development process to reflect these changes. This agility is crucial in the fast-paced financial industry [54], [55]. In the scholarly context, it is projected that low-code development platforms will persist in their evolution, becoming increasingly sophisticated in the future with the integration of artificial intelligence and machine learning capabilities. These enhancements are anticipated to simplify the construction of complex applications for developers and streamline the overall software development process. This democratization of software development is expected to empower businesses, irrespective of their size, to construct custom applications tailored to their unique requirements [53].

VII. LIMITATIONS

In the scholarly discourse, given the hitherto limited research on the subject of low-code automation in risk management within the banking industry, it can pose a challenge to compare and derive generalizable conclusions for low-code automation, particularly in relation to risk management practices in banking. This has further complicated the task of estimating the significance of the less frequently mentioned experiences, as additional studies are required to ascertain the magnitude and extent of these issues. While low-code development platforms typically offer similar features [19], the author acknowledges that the selection of a specific platform could potentially impact the benefits assessed in the study. Nevertheless, the exploration undertaken by this study has broadened the scope for studying low-code automation in risk management and may serve as a foundation for future researchers to investigate and compare the significance of low-code automation in risk management within the banking industry.

VIII. CONCLUSION

Low-code development embodies a contemporary technique in software engineering that markedly minimizes the dependence on conventional manual coding. Low-code platforms democratize the development process, making it accessible to all users, including those without technical proficiency. The utilization of these platforms does not necessitate familiarity with proprietary coding languages. The challenge of fulfilling the escalating demands for development is increasingly formidable especially in the banking and other financial industries. Low-code platforms, furnished with pre-existing code, visual development features, and advanced tools, significantly expedite the delivery process of banking software. These platforms encompass the comprehensive lifecycle of banking application development, from development and testing to debugging, and deployment. Low-code automation could potentially serve as an optimal approach, offering a plethora of compelling benefits to the procedures of risk management in banking, provided it is executed in alignment with the organization's policies, objectives, and industry standards.

REFERENCES

[1] E. Sahinaslan, O. Sahinaslan, and M. Sabancıoglu, "Low-code application platform in meeting increasing software demands quickly: SetXRM," AIP Conference Proceedings, vol. 2334, no. 1, pp. 070007, Mar. 2021.

[2] J. Metrôlho, R. Araújo, F. Ribeiro, and N. Castela, "An approach using a low-code platform for retraining professionals to ICT," in EDULEARN19 Proceedings, IATED, 2019, pp. 7200–7207.

[3] P. Vincent, K. Iijima, M. Driver, J. Wong, and Y. Natis, "Magic quadrant for enterprise low-code application platforms," Gartner report, 2019.

[4] J. Viljoen, M. Nguyen, M. Kauschinger, and A. Hein, "Fostering scalable citizen development in organizations: towards a guiding framework," in 29th Americas Conference on Information Systems, 2023, pp. 1802–1811.

[5] The London Institute of Banking & Finance, "Risk management in banking," [Online]. Available: https://www.libf.ac.uk/news-and-insights/news/detail/2022/09/08/risk-management-in-banking. Accessed May 6, 2023.

[6] McKinsey (December 2015) The future of bank risk management [Online]. Available: https://www.mckinsey.com/~/media/mckinsey/dotcom/client_service/risk/pdfs/the_future_of_bank_risk_management.pdf. Accessed May 06, 2023.

[7] Appian (November 2021) How Low-Code Platforms Are Democratizing Financial Services Automation [Online]. Available: https://appian.com/blog/2021/low-code-platforms-democratizing-financial-services-automation.html. Accessed May 07, 2023.

[8] FintechOs (February 2022) Why banks need to know about low-code/no-code [Online]. Available: https://fintechos.com/blogpost/why-banks-need-to-know-about-low-code-no-code/. Accessed May 07, 2023

[9] C. Y. Hyun, "Design and implementation of a low-code/no-code system," International journal of advanced smart convergence, vol. 8, no. 4, pp. 188–193, 2019.

[10] T. Margaria and B. Steffen, "Extreme model-driven development (xmdd) technologies as a hands-on approach to software development without coding," Encyclopedia of Education and Information Technologies, pp. 732–750, 2020.

[11] R. I. Strømsted, M. Marquard, and E. Heuck, "Towards low-code adaptive case management solutions with dynamic condition response graphs, subprocesses and data," in 2018 IEEE 22nd International Enterprise Distributed Object Computing Workshop (EDOCW), IEEE, 2018, pp. 12–16.

[12] B. Adrian, S. Hinrichsen, and A. Nikolenko, "App Development via Low-Code Programming as Part of Modern Industrial Engineering Education," in Advances in Human Factors and Systems Interaction, I. L. Nunes, Ed., Cham: Springer International Publishing, 2020, pp. 45–51.

[13] R. van Engelen, D. Whalley, and X. Yuan, "Automatic validation of code-improving transformations on low-level program representations," Science of Computer Programming, vol. 52, no. 1–3, pp. 257–280, 2004.

[14] Code or No Code (January 2023) Low Code History [Online]. Available: https://codeornocode.com/no-code/low-code-history/. Accessed May 07, 2023

[15] R. Sanchis, Ó. García-Perales, F. Fraile, and R. Poler, "Low-code as enabler of digital transformation in manufacturing industry," Applied Sciences, vol. 10, no. 1, p. 12, 2019.

[16] Outsystems (December 2021) Artificial Intelligence [Online]. Available: https://success.outsystems.com/documentation/11/extensibility_and_integration/artificial_intelligence/. Accessed May 20, 2023

[17] Outsystems (October 2023) 2023 Gartner Critical Capabilities for Enterprise Low-Code Application Platforms [Online]. Available: https://www.outsystems.com/1/gartner-low-code-platform-critical-capabilities/. Accessed November 20, 2023

[18] Outsystems (2023) The Forrester Wave™: Low-Code Development Platforms For Professional Developers, Q2 2023 [Online]. Available: https://www.outsystems.com/1/low-code-development-platforms-wave-/. Accessed November 20, 2023

[19] A. Sahay, A. Indamutsa, D. Di Ruscio, and A. Pierantonio, "Supporting the understanding and comparison of low-code development platforms," in 2020 46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), IEEE, 2020, pp. 171–178.

[20] Appian Low-code application development [Online]. Available: https://appian.com/products/platform/low-code.html. Accessed July 15, 2023

[21] Appian (2023) Appian Named a Leader in the Low-Code Gartner Magic Quadrant for 2023 [Online]. Available: https://appian.com/learn/resources/resource-center/analyst-reports/2023/gartner-magic-quadrant-for-low-code-2023.html. Accessed November 20, 2023
[22] Appian When to Use AI and ML [Online]. Available: https://docs.appian.com/suite/help/23.4/ai-use-cases.html. Accessed July 15, 2023
[23] Salesforce, Quickly build AI-powered apps for employees and customers on a complete artificial intelligence platform [Online]. Available: https://www.salesforce.com/products/einstein/features/. Accessed July 15, 2023
[24] Microsoft Power Apps, Low-code platform guide [Online]. Available: https://powerapps.microsoft.com/en-us/low-code-development-guide/. Accessed July 16, 2023
[25] Microsoft Power Apps, Low-code vs. traditional development [Online]. Available: https://powerapps.microsoft.com/en-gb/low-code-vs-traditional-development/. Accessed August 20, 2023
[26] Microsoft (2023) Low-code: One of the best investments for IT departments in 2023 [Online]. Available: https://www.microsoft.com/en-us/power-platform/blog/2023/04/13/low-code-signals-2023/. Accessed July 16, 2023
[27] Microsoft Power Automate (2021) Enhancing AI for Low Code Development with AI Builder [Online]. Available: https://powerautomate.microsoft.com/en-us/blog/enhancing-ai-for-low-code-development-with-ai-builder/. Accessed July 22, 2023
[28] ServiceNow, Accelerate innovation with low-code app dev [Online]. Available: https://www.servicenow.com/solutions/hyperautomation-and-lowcode/low-code-app-development.html. Accessed July 22, 2023
[29] ServiceNow (2022) App Engine [Online]. Available: https://www.servicenow.com/products/now-platform-app-engine.html. Accessed July 22, 2023
[30] ServiceNow (2023) AI at ServiceNow [Online]. Available: https://www.servicenow.com/standard/resource-center/infographic/ai-infographic-layout.html. Accessed July 22, 2023
[31] Isaca (January 2022) Cybersecurity and Technology Risk in Virtual Banking [Online]. Available: https://www.isaca.org/resources/isaca-journal/issues/2022/volume-1/cybersecurity-and-technology-risk-in-virtual-banking. Accessed July 22, 2023
[32] Mckinsey (August 2022) Creating a technology risk and cyber risk appetite framework [Online]. Available: https://www.mckinsey.com/capabilities/risk-and-resilience/our-insights/cybersecurity/creating-a-technology-risk-and-cyber-risk-appetite-framework. Accessed August 12, 2023
[33] M. Firoozi and S. Mohsni, "Cybersecurity disclosure in the banking industry: a comparative study," International Journal of Disclosure and Governance, vol. 20, no. 4, pp. 451–477, 2023.
[34] Deloitte (2022) Beneath the Surface: Technology-driven systemic risks and the continued need for innovation [Online]. Available: https://www.deloitte.com/global/en/Industries/financial-services/analysis/beneath-the-surface-technology-driven-systemic-risks-and-the-continued-need-for-innovation.html. Accessed on August 12, 2023
[35] M. Woo, "The rise of no/low code software development—no experience needed?," Engineering (Beijing, China), vol. 6, no. 9, p. 960, 2020.
[36] ABA Banking Journal (June 2023) Low-code no-code: A visual approach to tech innovation for banks. [Online]. Available: https://bankingjournal.aba.com/2023/06/low-code-no-code-a-visual-approach-to-tech-innovation-for-banks/. Accessed August 13, 2023
[37] Infopulse (August 2022) Low-code Development in Banking. https://www.infopulse.com/blog/low-code-benefits-use-cases-banking. Accessed August 19, 2023
[38] T. Kanchu and M. M. Kumar, "Risk management in banking sector–an empirical study," International journal of marketing, financial services & management research, vol. 2, no. 2, pp. 145–153, 2013.
[39] E. Losiewicz-Dniestrzanska, "Monitoring of compliance risk in the bank," Procedia Economics and Finance, vol. 26, pp. 800–805, 2015.
[40] BIS (April 2005) Compliance and the compliance function in banks [Online]. Available: https://www.bis.org/publ/bcbs113.pdf. Accessed November 12, 2023
[41] McKinsey (January 2016) A best-practice model for bank compliance [Online]. Available: https://www.mckinsey.de/~/media/McKinsey/Business%20Functions/Risk/Our%20Insights/A%20best%20practice%20model%20for%20bank%20compliance/A_best_practice_model_for_bank_compliance_2.pdf. Accessed November 12, 2023
[42] R. J. Bolton and D. J. Hand, "Statistical fraud detection: A review," Statistical science, vol. 17, no. 3, pp. 235–255, 2002.
[43] N. Chen, B. Ribeiro, and A. Chen, "Financial credit risk assessment: a recent review," Artificial Intelligence Review, vol. 45, pp. 1–23, 2016.
[44] W. S. Frame, P. McLemore, and A. Mihov, "Haste makes waste: Banking organization growth and operational risk," Available at SSRN 4412401, 2023.
[45] SC Magazine (February 2022) How low-code automation can help security teams mitigate phishing attacks [Online]. Available: https://www.scmagazine.com/perspective/how-automation-can-help-security-teams-mitigate-phishing-attacks. Accessed November 12, 2023
[46] G. González-Granadillo, S. González-Zarzosa, and R. Diaz, "Security information and event management (SIEM): analysis, trends, and usage in critical infrastructures," Sensors, vol. 21, no. 14, p. 4759, 2021.
[47] RedHat (September 2023) What is security information and event management (SIEM)? [Online]. Available: https://www.redhat.com/en/topics/security/what-is-SIEM. Accessed November 19, 2023
[48] G. Armstrong and C. Gilge, "Make it, or break it-reimagining governance, people and technology in the construction industry: Global construction survey 2017," KPMG International, 2017.
[49] Projectmanagers (December 2023) Top 10 Cons & Disadvantages of Low-Code No-Code Platforms [Online]. Available: https://projectmanagers.net/top-10-cons-disadvantages-of-low-code-no-code-platforms/. Accessed December 16, 2023
[50] S.pro (May 2023) Exploring Low Code Development Platforms: A Comprehensive Guide [Online]. Available: https://s-pro.io/blog/exploring-low-code-development-platforms-a-comprehensive-guide. Accessed August 20, 2023
[51] Outsystems (January 2023) Measuring Scalability: How Scalable Is a Low-Code Platform? [Online]. Available: https://www.outsystems.com/blog/posts/measuring-low-code-scalability/. Accessed August 20, 2023
[52] M. B. Gunjal and V. R. Sonawane, "Multi authority access control mechanism for role based access control for data security in the cloud environment," International Journal of Intelligent Systems and Applications in Engineering, vol. 11, no. 2s, pp. 250-264-250–264, 2023.

[53] Kyanon (March 2022) How Low-Code Automates Processes In The Banking Industry [Online]. Available: https://kyanon.digital/how-low-code-automates-processes-in-the-banking-industry/. Accessed August 20, 2023
[54] Finextra (February 2021) Banking on Low-Code: The Secret Sauce of Digital Banking Systems That Are Crushing It [Online]. Available: https://www.finextra.com/blogposting/19904/banking-on-low-code-the-secret-sauce-of-digital-banking-systems-that-are-crushing-it. Accessed August 26, 2023
[55] Fisglobal (August 2022) Low-code, no-code: Building apps for digital and mobile banking [Online]. Available: https://www.fisglobal.com/en/insights/what-we-know/2022/august/low-code-no-code-building-apps-for-digital-and-mobile-banking. Accessed August 26, 2023

ABOUT THE AUTHOR

Deepa Ajish, currently serving as a Vice President at MUFG Bank Ltd. in California, USA, holds a Bachelor of Technology (B.Tech) degree in Applied Electronics and Instrumentation Engineering from the University of Calicut, Kerala, India. With over 20 years of experience in the field of Information Technology, her professional interests encompass cloud computing, security, compliance, and automation.