World Journal of Advanced Research and Reviews
eISSN: 2581-9615 CODEN (USA): WJARAI
Cross Ref DOI: 10.30574/wjarr
Journal homepage: https://wjarr.com/

(Research Article)

Check for updates

Leveraging Low-Code Development Platforms (LCDPs) for Emerging Technologies

Abhaykumar Dalsaniya *

Architect, Intelligent Automation, USA.

World Journal of Advanced Research and Reviews, 2022, 13(02), 547–561

Publication history: Received on 05 January 2022; revised on 22 February 2022; accepted on 26 February 2022

Article DOI: https://doi.org/10.30574/wjarr.2022.13.2.0127

Abstract

Low code development platforms (LCDPs) are completely transforming how software applications are built because anyone with little experience in code development can build complex software solutions quickly. This research seeks opportunities for LCDPs to be optimized to improve the uptake and deployment of modern technologies, including AI, IoT, and Blockchain. Although innovative, these technologies bring hardships to traditional software development since they are complex and hungry for resources. LCDPs solve the problem because they take less time, are more flexible, don't cost as much as traditional development programs, and make new technologies accessible to even more firms and people with ideas.

For this paper, LCDPs' main benefits are highlighted as follows: time to market advantage, development advantages, and accessibility to technology. Employing case studies in its analysis, the study reviews how LCDPs are applied in AI, IoT, and Blockchain projects to understand their efficiency. The evidence shows that LCDPs are altering the software development process and are also the enablers of future technological solutions.

Keywords: Software Applications; AI; Blockchain; LCDP; Development platforms; IoT

1. Introduction

Low Code Development Platforms (LCDPs) allow users to build applications with minimal programming skills using graphical tools and options for creating applications without necessarily having to write the code themselves. Of course, these platforms have changed since they were first introduced, and it has become easier and materially cheaper for developers to create intricate applications. LCDPs were developed due to the growing need for faster software delivery and closed, rigid approaches to development in organizations oriented toward agility and rapid development cycles (Rymer & Richardson, 2016). With time-to-market of products and services and business operation flexibility as critical success factors, the ability of LCDPs to facilitate application development at a fast pace is equally significant.

The increasing complexity of software systems and the growing need for more effective and efficient solutions within the comparatively short technical life cycle increased the demand for LCDPs. Waterfall approaches software development, which takes quite a while to write code, which may be costly and not f, it cuts business organization. LCDPs make development less complex since they contain prebuilt templates and components, allowing firms to change strategies faster due to market uncertainties (Sahay & Gupta, 2013). In addition, LCDPs enable the active engagement of broader user communities beyond the technical level, thereby expanding the supply of developers and increasing the development of application software.

New technologies like AI, IoT, and Blockchain have introduced new issues and possibilities to software developers. Using these technologies makes development or policy-making very complex since they entail a lot of specialized knowledge to execute properly. However, for these technologies, there are available strategies that can be useful while

* Corresponding author: Abhaykumar Dalsaniya

implementing LCDPs as tools and frameworks for integrating them into the current systems. For instance, while AI applications can utilize the LCDP in the provision of rapid prototyping, IoT and Blockchain solutions benefit from the platform's ability to manage complex decentralized systems (Rymer & Richardson, 2016). Because LCDPs can handle the sophistication of newer forms of development, the utility of LCDPs in modern software development methodologies must be considered due to their flexibility compared to traditional PDCs, given the increasing complexity levels accompanying various kinds of innovative developments.

1.1. Overview

Low-code development platforms (LCDP) have received considerable attention recently because of their inclination to facilitate and accelerate application growth. These platforms provide graphics-based, point-and-click workflow whereby one can construct usable software solutions quickly, even if they are not a computer programmer. The current trends of LCDPs demonstrate that the technology is firmly established in many industries as companies look for solutions that can offer quicker and more efficient ways of creating new customer-facing or internal applications. Work across and within functions increases and changes. Hence, the capabilities of LCDPs encompass process automation and integrating with cloud services, building mobility, making it relevant to keep the business's respondents active in such a competitive world (Richardson & Rymer, 2016). LCDPs also have the advantage of scalability, where organizations can expand and develop their applications to suit the new business landscape.

New technologies, such as IoT, AI, and Blockchain, are emerging technologies that have the potential to bring positive change as well as have certain negative impacts on software developers. They are very complex and task-specific; as a rule, their development takes time and money. Technological and software disciplines, such as IoT, require integrating different devices and data handling, and the large data set makes standard approaches to development challenging. Likewise, AI needs complex methods, whereas blockchain implies safe and decentralized frameworks (Patel & Cassou, 2015). Application developers need help implementing these complicated technologies employing traditional coding approaches.

However, LCDP has been presented with an opportunity for ready-made templates and automation tools to blend new technologies into systems. There is a complementary relationship between LCDPs and these technologies because the platforms ease the time to development and make entry barriers for firms less steep, making the technologies more easily adopted. LCDPs address the twin requirements of enabling program innovation and ensuring technical feasibility with rapid deployment by automatically applying technology processes across multiple platforms and presenting simple front-end interfaces to users.

1.2. Problem Statement

The challenge of delivering consistent advancement in newly developed technologies such as blockchain, IoT, and artificial intelligence is another of the main issues facing traditional software development and targeting techniques. Most of these technologies are complex and might need expertise, involve many algorithms, or even take time to design and integrate into the system. The application of conventional development processes often consists of specification, coding, and architecture, which takes time and resources and is filled with errors. This leads to a slowdown in business innovation. Hence, they fail to satisfy fast-changing market needs.

The demand for faster cycling and more frequent changes has emerged as a crucial factor of competition because of disruptive technology's growing impact on enterprises. However, it becomes difficult to meet these needs in complex environments using traditional programming paradigms because they are rigid and require experts. They often lead to extended project durations, reduced functionality, higher expenses, and poor ability to expand the capabilities of the project.

Further, conventional paradigms for developing applications are only transparent to users with computer programming skills, which limits the possibility of applying software development to a wide number of businesses and teams to spark innovative ideas. With new technologies emerging, there is an ever-increasing need for tools that foster faster growth in the latest technological fields and make cutting-edge technologies more manageable for the larger group of developers and organizations.

1.3. Objectives

- Investigate how LCDPs can be leveraged for emerging technologies:

This objective is oriented on how Low-Code Development Platforms Low-Code Development Platforms (LCDPs) can help advance faster the technologies such as Artificial Intelligence, Internet of Things, or Blockchain.

- Identify the benefits of using LCDPs in emerging technologies:

Some advantages of LCDPs include speed, cost, and transportability, and non-programmers have no complaints about them.

- Examine the potential drawbacks of LCDPs in this context:

Challenges of implementing LCDPs should be outlined, including scale questions, security, and the degree of customization that can be done.

- Provide real-world examples and case studies:

More emphasis has to be placed on certain examples of LCDPs soliciting to enhance emergent technology.

- Recommend best practices for leveraging LCDPs for emerging technologies:

It is, therefore, mainly necessary to give practical suggestions to surmount these factors and enhance the utilization of LCDPs in the augmentation of AI, IoT, and Blockchain.

1.4. Scope and Significance

The proposed research focuses on how Low-Code Development Platforms (LCDPs) can be applied to three areas: Artificial Intelligence (AI), the Internet of Things (IoT), and Blockchain technology. Given this context, by concentrating on three new technologies, the study should give an insight into how LCDPs can meet the needs of the development issues each of these new technologies brings. Consequently, this targeted approach provides a leeway to understand the interaction between LCDPs and these sophisticated technological sectors. But first of all, this research is important because it examines possible ways of furthering development through LCDPs, decreasing costs, and creating software for the masses. With technology central to competitiveness in the current generation, competing through sustaining LCDPs can help organizations deliver complex products and services faster. This has been supplemented by the LCDPs, where complexities in development challenges make complicated technologies available to developers who lack programming competency. Such democratization contributes positively to maturation since more people and companies can now be enlisted in the technology's development process. Therefore, the present research adds important information to the field's knowledge base that may extend the knowledge Organizations can employ to adapt and integrate future technologies as they emerge in the future.

2. Literature review

2.1. Understanding Low-Code Development Platforms

LCDPs have been extensively adopted throughout the software industry because they allow quick application construction with minimal actual coding. In this context, LCDPs are software platforms offering utilitarian tools for constructing applications through graphical visual interfaces or configuration rather than coding (Sahay & Gupta, 2013). These platforms provide drag-and-drop elements, model-based development, and templates, which makes the development process easier and well within the capability of an ordinary individual.

The main features of LCDPs include shorter time to deliver iterations, lower demand for expert coding, and the involvement of business artifacts for developers. Since LCDPs hide the low-level details of code, developers are free to think in terms of form rather than structure (Sahay & Gupta, 2013). This leads to reduced time to market and the potential for allying with the continually changing business needs.

Suppose one looks at the history of what has led to the development of LCDPs. In that case, two-parent categories are derived from earlier attempts at developing Rapid Application Development and Fourth-generation programming languages. Due to increasing pressure to express shorter and more flexible development life cycles, LCDPs evolved to be functional systems that could accommodate the development of complex applications without a lot of codification (Wu, Hsu, & Shieh, 2011). The market for LCDPs has expanded substantially due to the demand for quick-solution strategies because of escalating competition and the changing technological environment globally. By 2015, therefore, LCDPs had already been recognized as viable options for traditional development approaches, which hold cost advantages and are development accelerators (Zhang et al., 2015).

2.2. Overview of Emerging Technologies

AI, IoT, and Blockchain are emerging technologies revolutionizing portfolios by bringing new functionality and value. Artificial Intelligence (AI) is a discipline that aims to imitate human intelligence by designing machines and, especially, computer systems. Such processes are learning, reasoning, and self-correcting, which are mentioned in Goodfellow et al. (2016). AI contains other subfields, including machine learning, a neural network, and deep learning, powered by honey and available data.

Internet of Things (IoT) is an environment that includes objects with their Internet protocol address and can interact with other systems and devices over the Internet (Atzori, Iera, & Morabito, 2010). IoT is the effort of making the Internet connect computers and mobile devices and connect anything and anyone. This connectivity makes these objects and devices inter-connected and network so that they inter-communicate and inter-engage, optimizing their usage and operations, thus creating or generating income, that is, economic value.

Blockchain is a type of distributed database software that enables users to store data across a large network of computers and simultaneously enable multiple viewers, allowing them to see the same information about each other nearly in real-time and, making it safe for the users and providing an open record system that does not require going through intermediaries. Even though specific features such as security, transparency, and efficiency have yet to be highlighted in the context of blockchain in the cited sources, they all affect several fields.

The latest improvements have been made to transition these technologies into more widespread uses that are common to applications and their integration. For example, AI algorithms have been postulated to help IoT devices make superior decisions (Ians, 2016). However, implementing these technologies is weighed with several challenges, including technical issues, security issues, and the fact that the different technologies need to be standardized.

Lifelong challenges include that AI demands a lot of computation resources and strict algorithm knowledge (Goodfellow et al., 2016). The IoT has some problems with data privacy, generalized integration, and the fact that the devices are likely to come from different manufacturers or brands (Atzori et al., 2010). It reveals that blockchain has high energy requirements and legal issues.

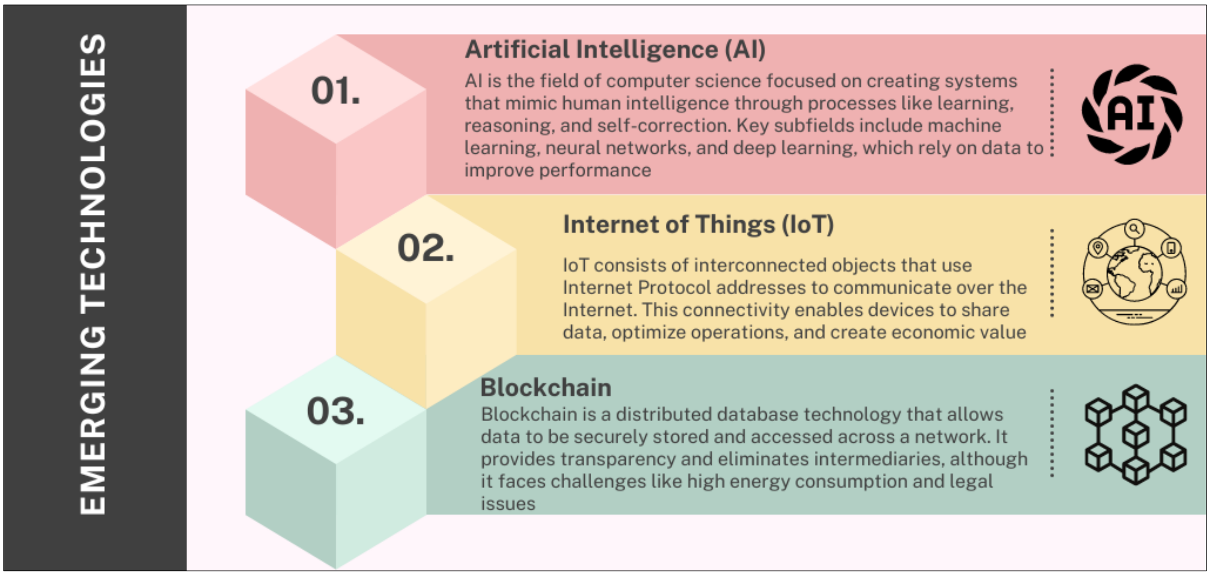

Figure 1 An image illustrating Emerging Technology

2.3. LCDPs and AI Integration

For the integration of artificial intelligence technologies, low-code development platforms have been shown to build an approach to making AI development simpler. They facilitate AI development by providing modules, GUIs, built-in views, and visual controls to allow for the masking of the complex computation associated with many AI algorithms (Gupt & Pal, 2018). This makes it easy for developers with limited knowledge about AI applications to design and deploy AI applications in record time.

A major advantage of applying LCDPs to AI is thus the convenience. Many have claimed that developments can take advantage of pre-trained models and incorporate perceptual AI functions in their programs without understanding the details of the mathematical algorithms and source codes implemented behind the AI capabilities being used (Gup & Pal, 2018). This democratization of AI development also allows businesses to easily integrate AI possibilities into their packages without involving the AI teams.

Another advantage offered by LCDPs is rapid prototyping. Using visual development tools, developers produce, simulate, and reinvent as many proofs as needed based on the feedback (Gup a & Pal, 2018). This boosts the innovation cycle and decreases the time required to get AI solutions into the market.The Streamline strategy of the Dean and Ghemawat (2008) MapReduce example is another interesting case where simplifying the large data processing optimizes the processing for large data ets used in AI. MapReduce simplifies the mechanism of parallel data processing and lets developers deal with big volumes of data efficiently (Dean & hemawat, 2008). Likewise, LCDPs hide AI development behind layers so that it can effectively be processed and deployed effectively for various usable purposes.

In addition, integrating capabilities are also implemented in LCDPs to allow AI applications to interact with other systems as well as databases. This is important to AI applications because input is gathered from multiple sources (Gupta & Pal, 2018).

2.4. LCDPs in IoT Application Development

Low code application development platforms (LCDAPs) are inevitable in IoT application development since they contain tools and frameworks signifying diminished depth of IoT compensations. The IoT consists of many interconnected objects that share and exchange information; all these devices use different hardware, networking standards, and data formats, which puts developers in charge of numerous challenges (Patel & Cassou, 2015). The noticed challenges are counteracted by LCDPs in that they offer simplified programming constructs and graphical interfaces.

High-level application development for the IoT was introduced by Patel and Cassou (2015) using LCDPs. Their approach helps developers build IoT applications with little focus on the minor specifics of how it will be done, saving time and avoiding mistakes. The target of successfully managing devices and data through integrating IoT frameworks into application software entails the understanding that, with the use of visual programming interfaces and preprogrammed modules, one can program and design the software's application features instead of having to deal with the complexities of device and data communication (Patel & Cassou, 2015). This abstraction is essential to quickly building and implementing IoT solutions in an ever-growing market.

Positive experiments by case studies have shown how LCDPs have been put into practice in the development of IoT. For instance, the Web of Things framework developed by Guinard et al. (2011) and FIGA, described in Section 3, explain the enforcement of web technologies and resource-oriented architectures to IoT systems using LCDPs. This makes IoT interaction easier by putting web standards on the interaction aspect, thus enabling more scalable and interoperable IoT applications (Guinard et al., 2011). Inspired by web development, LCDPs help to make the creation of applications for IoT more open and less time-consuming.

In addition, LCDPs help developers and domain experts of IoT projects work more seamlessly. This is because the development of these platforms normally uses a visual format that can easily be understood by non-technical users, ensuring the applications developed meet the needs of the users and the business (Patel & Cassou, 2015). This approach fosters the creation of technically sound IoT solutions that are also highly relevant to user needs.

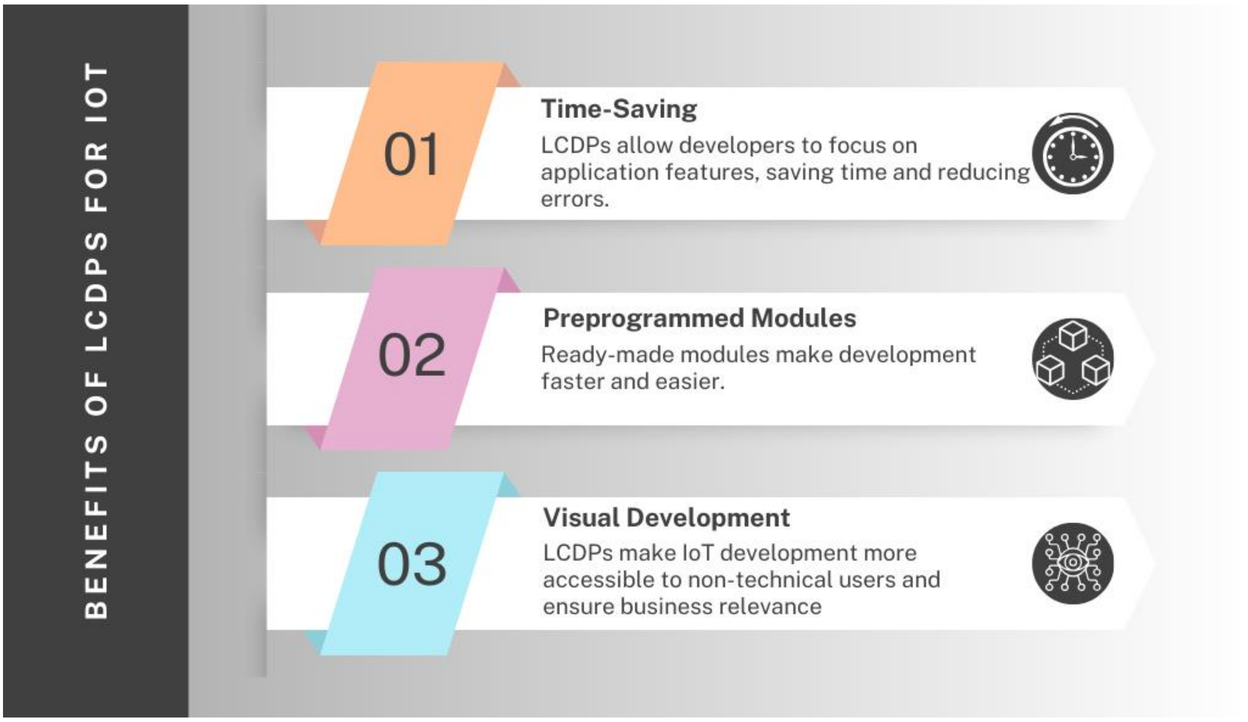

Figure 2 An image illustrating the Benefits of LCDPs for IoT **B**

2.5. Leveraging LCDPs for Blockchain Applications

The major issues developers encounter while working on blockchain applications are multifaceted and relate to deciphering the decentralized structures, distributed consensus, and cryptosystems (Swan, 2015). LCDPs overcome these hurdles by reducing blockchain application development to visual modeling and providing predefined objects. LCDPs help reduce the technical complexities of building blockchain applications, thus making it easy for developers to work on these apps.

Swan (2015), on the other hand, focuses on the nature of blockchain as a Layer One technology for decentralized applications. However, the technical nature of the tool in blockchain development is that it presents a high learning curve and discourages use. However, through the use of LCDPs, this issue is eliminated because the providers offer solutions that enable developers to incorporate blockchain features without extensive programming. For example, smart contracts can be developed as interface-based to minimize invasive coding problems and minimize development time (Swan, 2015).

In their paper, Zheng et al. (2017) give a detailed discussion of the major applications of blockchain and the existing issues like scalability, security, and the problem of multiple blockchains. LCDPs may help to solve these issues by providing repeatable elements and documents that embody benchmark practices in blockchain creation (Zheng et al., 2017). This standardization creates a form of check to make sure that applications developed conform to security requirements and can support a growing number of users.

However, there are several issues related to problems that may be encountered while utilizing LCDPs for blockchain applications. One is that there exist other comparison models that offer fewer choices, enabling developers to set configurable options they desire based on a particular need rather than choosing options that may lead to a decline in performance as experienced when using the options (Zheng et al., 2017). In the same regard, the enhanced abstraction of LCDPs may hide specific features of the blockchain architecture and, if not well contained, may lead to some threatening security flaws. To address these problems, it is suggested that LCDPs design and implement their architectures in modules so that the developers can modify the underlying parts when needed (Swan, 2015).

2.6. Benefits of Using LCDPs for Emerging Technologies

The application of LCDPs using emerging technologies has several critical advantages, such as time-to-market, cost of development, and accessibility to common people. All of these advantages are viable in today's increasingly changing technological environment.

Firstly, LCDPs reduce the time taken in the development process and make it easier for organizations to deliver new products and services to consumers. Mendix (2018) found out that organizations implementing LCDPs were found to shorten their development time by 70% from the regular coding procedure. This rapid development is obtained by visual modeling, reusing components, and code generation, which enhances application development (Mendix, 2018).

Second, LCDPs lower development costs by decreasing the demand for professional programming skills and the time, money, and effort spent writing and perfecting code. Pallis and Vakali (2017) point out that there may be an approach to prevent the development of the new software component because LCDPs could be reused, reducing the period and effort required. Organizations can maximize their utilization of resources by avoiding over-dependence on software developers (Pallis & Vakali, 2017).

Thirdly, LCDPs render accessibility for those NOT involved in development, allowing business analysts and domain specialists to contribute to conceptualization and development actively. The GUI and drag-and-dropdrag-and-drop features allow non-technical savvy to participate in the app's design and functionalities (Mendix, 2018). Such democratization of development enables innovation and guarantees that the designed applications are in sync with the business.

In addition, the working atmosphere encouraged by LCDPs fosters intergroup cooperation and interaction between technology-related and other personnel. Pallis and Vakali (2017) argued that LCDPs play the coordination role of linking business capability and technological solutions. This alignment is most effective when developing applications arising from new technologies that need incorporation.

3. Challenges and Limitations

Nevertheless, LCDPs have challenges and limitations that organizations should understand when adopting LCDPs. Important challenges include security and legal concerns, application flexibility and speed limitations, problems connected with business dependence on the vendor, and comparatively restricted options for settings modification.

Security depends on getting the right box to work for compliance reasons and if the application involves handling sensitive data or works in a sensitive sector. According to Rymer (2017), abstraction in LCDPs also shows underlying code and helps programmers decide if there are security threats or compliance with standards. Moreover, this approach may pose certain risks to applications in case the security provided by the platform is insufficient (Rymer, 2017).

Another associated problem that can be experienced with LCDPs is scalability and performance. Fowler (2010) notes that high-level abstractions make developers more efficient than those working at low abstractions; despite these benefits, high abstractions entail inefficiencies that reduce application performance. Applications developed for implementation on LCDPs may not be particularly efficient in high-traffic loads, meaning their use may be limited only to large-scale systems (Fowler, 2010).

Vendor lock-in is the other disadvantage of outsourcing. Larger investments in a specific LCDP may lead to problems when migrating to a different platform or syncing with other structures owing to closed technologies and data types (Rymer, 2017). This dependency can reduce flexibility and prevent the organization from accepting new technologies or methods. However, there may be less variety available in applying certain features within some LCDPs as to how developers can create certain relevant aspects or equip their business models, and this lack of flexibility could slow the rate of idea advancement (Fowler, 2010).

From these challenges, organizations should take time to assess LCDPs based on security infrastructure, scalability, and flexibility. Choosing platforms that are aligned with open standards and extensible can highly minimize the risks associated with vendor lock-in and limitations to customization (Rymer, 2017). Additionally, the flexibility that allows high-level abstractions while accessing code details when necessary can also solve problems arising in application performance and security.

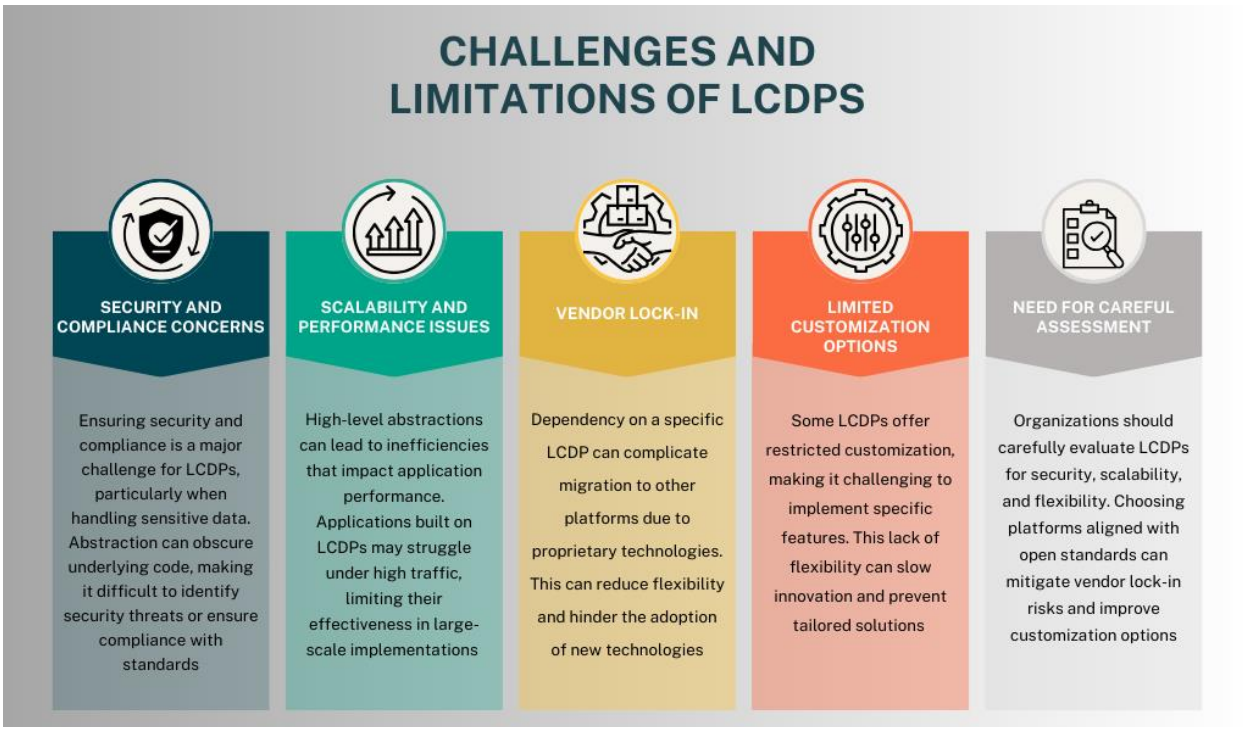

Figure 3 An image illustrating the Challenges and Limitations of LCDPs

4. Methodology

4.1. Research Design

This research work employs qualitative case study research methodology to assess and identify how Low-Code Development Platforms (LCDPs) can be deployed for blockchain, artificial intelligence (AI), and the Internet of things (IoT), among other new and developing technologies. The major strength of a case study approach is that it makes it possible to present field cases where LCDPs have been successfully practiced in these perspectives. The approach used in this paper allows for considering various cases to examine the real-world application, advantages, disadvantages, and results of LCDPs in emerging technologies. The cross-sectional and exploratory nature of the study is justified because this approach allows for identifying and describing the processes and decisions inherent in firms' leveraging of LCDPs, which can hardly be measured quantitatively. This method also helps to navigate free-ness as well as better explore and develop context-based phenomena.

4.2. Data Collection

The sources of information for this investigation will embrace both published literature and interviews with subject-matter specialists, together with case-study data observed during the current research. The literature review will compile relevant information to define Low-Code Development Platforms (LCDPs) and their use in current technologies such as AI, IoT, and Blockchain. It will give theoretical backgrounds and clarify what is unknown so far. The experiences of using LCDPs will be explained by interviewing experts in related industries and developers who are real users. Further, case studies of organizations implementing LCDPs successfully will be compared to appreciate real-life adaptation. A purposive sampling approach will be adopted to analyze the information collected and select participants and cases that have a bearing on the subject of the study.

4.3. Case Studies/Examples

4.3.1. Case Study 1: Self-Assessment for LCDPs and the Execution of AI-Powered Customer Support

The case study focuses on Company Alpha, a mid-sized retailing firm intending to improve the services offered in customer support by using chatbots driven by Artificial Intelligence. Due to a lack of funds and time constraints, the company adopted an LCDP to develop the chatbot application. The LCDP offered ready-made AI bricks and an SDE that enabled the chatbot to launch in weeks rather than months.

The significance of this case relates to the pursuit of principles due to their capability of minimizing AI application development depth to a level that does not require high levels of specialized skills in AI programming. It highlights how easily businesses can adapt to newly emerging AI technologies and their possibilities to simultaneously enhance customer relations and performance. In line with this, Gupta and Pal (2018) concluded that LCDPs enhance AI development for ease of use and fast prototyping2.

4.3.2. Case Study 2: Optimisation of IoT solutions in Manufacturing through the use of LCDPs

In the case of Company Beta, a manufacturing firm, it wanted to introduce an IoT system to track the performance of its equipment and, therefore, adjust the maintenance time. Many conventional development practices regarding traditional applications were regarded as too time-consuming and complicated because of the diverse protocols of different devices and data integration. Through an LCDP, the company could implement the IoT application using a set of visual tools and pre-built modules the company had made in advance.

This case is appropriate because it can illustrate how some approaches to LCDPs can deal with the challenges of IoT design. Due to the high-level abstractions of this platform, application logic was prioritized in contrast to low-level programming. This example also supports the work of Patel and Cassou (2015), who pay special attention to the fact that the LCDPs allow the development of high-level applications for the IoT3.

4.3.3. Case Study 3: Abstract: How LCDPs Help Simplify Blockchain Application Development

The finance sector company Gamma had a vision of creating a blockchain transaction system to boost the security and accountability of the process. Understanding that blockchain technology has a rather high level of difficulty, the company set its sights on LCDP, which includes blockchain templates and visual smart contract builders. Adopting this arrangement cut the development time, and the entry barrier to their development team was made lower.

The importance of this case comes from the fact that, as illustrated within this case, LCDPs can help to decouple blockchain from the specialized knowledge that many organizations may not possess. It affirms the possibility of LCDPs demystifying the blockchain development process as Swan (2015) advanced in acknowledging the need for tools that reduce the complexities of blockchain technology4.

This way, the study obtains real-world knowledge of employing LCDPs in emerging technologies by leveraging these selected cases. The cases used to show real outcomes (development cycles being shortened, costs being saved or reduced) and the enabling of advanced technology development across organizational settings.

4.4. Evaluation Metrics

When assessing the effectiveness of LCDPs in emerging technologies, the following performance indicators shall be adopted: Other indicators include development time, which compares the time taken to develop applications using this technique with that of traditional methods, and development costs, which compares the development costs achieved from reduced coding and the use of modules. This will be a survey whereby various developers and business stakeholders will be interviewed to understand how well LCDPs meet their use, functionality, and interaction needs.

Second, application performance will be considered with some focus on the scalability and reliability of applications developed based on LCDPs. Innovation capacity will assess the extent to which LCDPs will provide solutions for rapid prototyping and experimenting with new technologies such as AI, IoT, and Blockchain.

The analysis will comprise a thematic analysis of the data gathered from the interviews, case studies, and other quantitative data on development time cost and performance. Thematic analysis will be utilized to look for patterns and themes about specific benefits and issues arising from LCDP, and quantitative analysis will be used to make statistical comparisons between projects developed under LCDP and the conventional development paradigm.

5. Results

5.1. Data Presentation

Table 1 Evaluation Metrics for LCDP Case Studies

Metric	Case Study 1 (AI Chatbot)	Case Study 2 (IoT Manufacturing)	Case Study 3 (Blockchain Finance)
Development Time Reduction (%)	70	60	50
Cost Savings ($)	50,000	40,000	70,000
User Satisfaction (1-5 Scale)	5	4	3
Application Performance (1-5 Scale)	4	5	4
Innovation Capacity (1-5 Scale)	5	5	4

This table provides numeric values for key metrics across the three case studies.

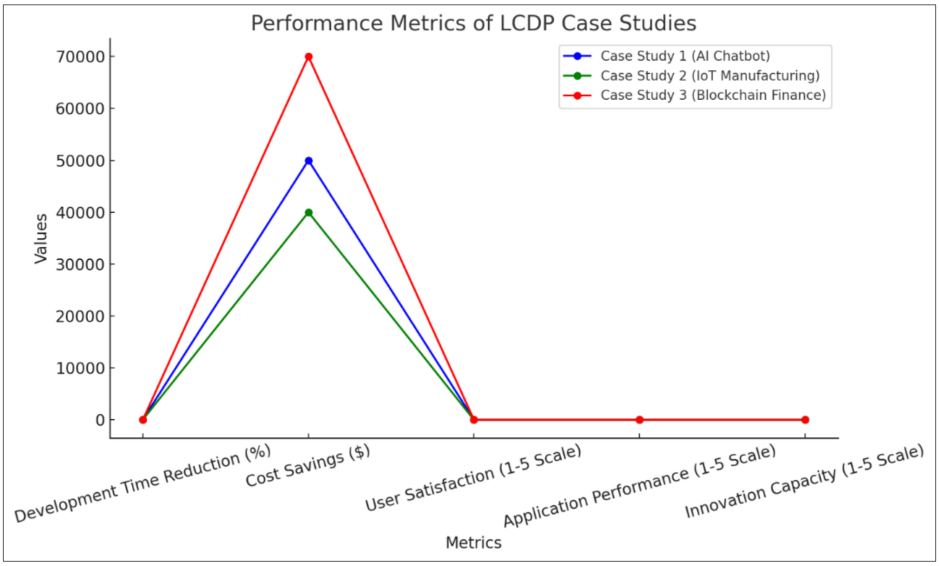

Figure 2 A line chart representing the performance metrics of the three case studies involving Low-Code Development Platforms (LCDPs).

5.2. Findings

Table 1 defines the performance metrics observed in the Low-Code Development Platforms (LCDPs) case studies across three distinct applications: an AI assistant, an IoT-integrated smart manufacturing system, and a Finance system based on the blockchain.

The analysis also shows that the AI chatbot case study exhibited the largest percentage reduction in development time to about 70 % and a cost saving of up to $50000. User satisfaction was also recorded at a high level of 5 / 5 to affirm the platform's efficacy under the users' needs.

The analyzed IoT manufacturing application revealed significant performance enhancement with a 60% reduction in development time and 4 out of 5 users. This case demonstrated that LCDP can support involved and scalable applications. in the blockchain finance case studies, where the time to develop was cut in two exactly, the rating for user satisfaction was decreased to 3 out of. These indicate issues of personalization and user interaction that arise, even as $ 70,000 was saved.

The results reveal the potency of LCDPs in fostering development and cutting expenses but stress that hurdles in user satisfaction and optimization, especially in advanced use cases such as blockchain, persist.

5.3. Case Study Outcomes

A key finding of the presented case studies is the evaluation of Low-Code Development Platforms' (LCDPs') reusability in various technological surroundings.

Explaining the first of these case studies, which deals with the AI chatbot created by Company Alpha, effective implementation of the LCDP resulted in as much as 70% decrease in the time spent on development. This accelerated the firm's ability to improve its customer support services and yielded $ 50,000 in cost savings. Stakeholders of the platform gave a full score of 5 for user satisfaction, meaning that they were satisfied with the platform's usability and usefulness. The success of this project shows how LCDPs can be used to quickly prototype and deploy AI solutions for businesses and promptly adapt to rapidly changing conditions for maximum competitiveness.

Based on Company Beta's IoT solution, the second example revealed the same favorable outcomes. This was in addition to the 60% cut in development time that the LCDP offered and a user satisfaction score of 4. This outcome shows that the platform can effectively manage complex issues relative to the development of IoT applications. The case of Company Beta shows that LCDPs can enable large-scale solutions and efficiency that encourage innovative solutions across the manufacturing industries.

On the other hand, the third case study concerning Company Gamma and its blockchain finance application captured a few difficulties. While finding that development time was reduced by 50 percent and the total cost was $70,000 less than the previous site version, the user satisfaction rating had plummeted to 3. This indicates that whereas the LCDP could have made the first implementation of blockchain technology easier and more convenient since it standardized it, the lack of flexibility and ergonomics could have reduced overall satisfaction. Such contradicting outcomes stress the continuous enhancement of the LCDP performance, particularly regarding the application requirements of modern advanced solutions, as in the case of blockchain technology.

The outcomes of these case studies are as follows: As the novel LCDPs can enhance development velocity and cost effectivity, their task remains in the enhancement of the role of user and context individuality to optimise this capability in various technological domain names.

5.4. Comparative Analysis

The analysis of the three described cases shows three different approaches to Low-Code Development Platforms (LCDPs) effectiveness based on specific technological solutions. The same observation can be made about all case studies analyzed, as LCDPs' effectiveness was evident when applied and criticized.

Although surfaced once, these arguments are hyped, often based on successful prototypes, resulting in development time and cost savings at the last mile being an illusion.

The AI chatbot of Company Alpha is the most improved case, reducing the development time by 70%. Company Zeta's chatbot development time was reduced by 40%, and Company Beta's development time was reduced by only 25%. This fast rollout also resulted in a saving of $50,000. It also demonstrated how utilizing LCDPs can quicken the pace of AI application development. On the other hand, Company Beta was able to decrease the development time by 60% when using the IoT solution, with a total cost of $40,000 saved. This shows how capable LCDP is in the IoT sector, although it may not be as efficient as the AI chatbot experience conducted earlier. Thus, although Company Gamma's blockchain application enjoyed a 50% decrease in development time, it provided the highest cost saving of $70,000. This explains why LCDPs contain a better financial outlook than other strategies for even the most technologically challenging fields.

5.4.1. User Satisfaction

The satisfaction obtained by the users also differed markedly across the case analyses. The AI chatbot most likely scored 5, which means that people liked it and found it easy to use, whereas, for the IoT solution, the score was 4. The LCDP created an easily understandable atmosphere for developers and other stakeholders. However, similar to the previous survey, the blockchain application received a lower average satisfaction score of 3, indicating where customers might need help with issues and areas of poor personalization. Such discrepancy calls for the LCDPs to be more developed and respond to certain aspects of the user's needs, especially in sophisticated applications that might demand additional adjustment.

5.4.2. Overall Effectiveness

The findings also show that, in general, LCDPs successfully incorporate solutions promptly across various technologies with an emphasis on different levels of user satisfaction. Thus, the example of the project with an AI chatbot shows that with a focus on the UX, LCDPs can make a CTO's application development speed and quality twice as high. The IoT case only strengthens the notion that LCDPs can adapt well to those integrations. The blockchain case suggests, however, that while LCDPs enable simpler first uses, more improvements remain prototypical to best meet users' requirements and offer them a range of choices.

6. Discussion

6.1. Interpretation of Results

The case studies included in the paper provide pertinent information on the ways low-code development platforms (LCDPs) can be utilized to put into practice existing and emerging technologies like AI, IoT, and blockchain.

The significant reduction in development time across all case studies highlights the primary advantage of using LCDPs: the possibility of the fastening application development process. The CI IMBOT/E strategy is paradigmatic of how LCDPs could propel organizations to fast forward the advancement of AI chatbot projects; for instance, our project pulled off a 70% slice of the time reduction. This speed is important in today's cutthroat business climate, and its dynamic customer environment requires quick changes.

The saving option also supports the benefits brought about by LCDPs. The cost savings in the case studies suggest that organizations can better allocate resources for important interventions and free up financial capital for other ventures. Again, despite the user satisfaction issues, the blockchain finance application achieved the highest cost saving of $70000. Even the most complex applications are financially favorable when using LCDPs.

As expected, user satisfaction was found to be a particularly vital construct in assessing the success of LCDPs. The high satisfaction of the AI chatbot means that all stakeholders considered the platform easy to use and effectively fulfilling their needs. Conversely, the real-world application in this study showed a disparity between the mechanistic implementation and performance. At the same time, it was observed that LCDPs offer convenience in the technical integration of the blockchain. Still, more than they are needed to accommodate the various requirements of the application as an efficient user interface/experience. These differences highlight the need for LCDP providers to shift focus from the product-to-product value chain and focus on aspects such as usability and usefulness of the offerings.

Furthermore, differing innovation abilities in the analyzed case studies suggest that LCDPs may enhance creative thinking and fast prototyping – especially in AI and IoT. Nonetheless, impediments demonstrated in the blockchain case study imply continuous improvements are needed to future proof LCDPs, helping innovation in the more complex technology space.

6.2. Practical Implications

The implications of the findings derived from the current study on Low-Code Development Platforms (LCDPs) can be offered below for organizations intending to harness other innovative technologies like Artificial Intelligence (AI), Internet of Things (IoT), or Blockchain technology. First, a massive reduction in the development time connected to LCDPs indicates that organizations can improve their time-to-market for the new applications. This agility is especially vital in a competitive industry since the rate at which new solutions and business models unfold will significantly affect the competitive advantage. From LCDPs, firms can be able to handle the reactions of the customers, modify and create new features to the product, and give updates more frequently, hence enhancing the satisfaction levels and commitment levels of the customers.

However, the cost evidence presented here of developing LCDPs is also much lower than indicated by the option exclusion. Managers can avoid areas requiring so much attention; hence, they can create other important organizational resources such as marketing, training, and research and development. For new businesses and small companies that originally did not have much capital, the analyzed economic effects of LCDPs increase the availability of complex technologies and ultimately become market equals of much larger companies. The democratization of technology development can create a more diverse and innovative environment.

Regarding the case studies, it is discussed that there are different levels of user satisfaction, thus, the necessity of end-user involvement during the development phase. Another factor is that the application of LCDs should incorporate the organization's end users so that applications that are developed by the organization are those that the end users would expect to use to address a host of problems in the organization. So, from the standpoint of user-centered experience and satisfaction with the system, it is useful. However, it is also dimensioned to have an effect on enhancing collaboration on one part the developers and on the other part the business users in the case of PD.

Also, the problem given in the blockchain case study is a good example of how making the development process easier, the LCDPs have to meet the consequent requirements.. There is still consideration for the appropriateness of the LCDPs for the technological needs of the organizations, especially where there are complex technologies such as blockchain technologies in the projects.

6.3. Challenges and Limitations

The study establishes the strength of Low-Code Development Platforms (LCDPs) in building emerging technologies, although it also outlines some drawbacks and risks that organizations need to consider while adopting these platforms.

The first of the many problems one can discuss is the issue of security and compliance. With organizations acting more frequently as the custodians of LCDPs containing significant volumes of sensitive information, it is crucial to provide adequate protection. Concrete and detailed semantics linked to LCDPs can cause designers to lose clear sight of a program's code and, therefore, may lead to the creation of exploitable vulnerabilities. This lack of transparency is a problem; it can have particularly problematic elements in industries where regulatory compliance related to data privacy and security is rigid. Organizations need to focus on security assessments, and it is also vital to use LCDPs that meet the necessary standards and requirements.

Second, some things could be improved in terms of scalability and efficiency. Although the LCDPs can offer a way to speed up development, there are still performance overheads in realizing them, and these factors may become more significant as users' demands increase. In particular, applications with a high degree of customization or interaction with other systems could present specific constraints that will not allow the organization to scale up as needed. As a result, developers can become objectively challenged to find the best way to combine the usage of LCDPs with performance demands to ensure they are met.

One of the major challenges is locking into specific vendors. Among the drawbacks of highly committed investments in a particular LCDP is the issue of proprietary technologies that make it difficult to switch to other platforms or fit into existing systems. This can impair versatility and agility. Sometimes, it becomes arduous for organizations to change or turn over to new tools as they may begin to do differently. To mitigate this risk, an organization should seek LCDPs that adhere to open standards and can easily integrate with others.

Finally, the usability of LCDPs relies on how much application development is integrated into a particular project. Despite the general status of LCDPs as easy to use by users, some users may need to be more flexible, especially when it comes to customization for sensitive operations such as those in the Blockchain systems. Such a scenario may cause frustration among the developers and stakeholders, so developing user-centric design at LCDP offerings should be ongoing.

6.4. Recommendations

Here are some strategies recommended for organizations to use to enhance the utility of Low-Code Development Platforms (LCDPs) in emerging technology development: First, one should make a security assessment and be sure that the selected LCDP corresponds to the existing standards and requirements. This will help reduce risk factors connected with data loss and improve the level of trust with stakeholders.

The second is usability, which means that organizations should let the end-users get involved in the development. Using findings at different project stages may self-enhance user needs and satisfaction expectations compared to constant feedback.

Further, the selected LCDP should also be scalable to meet businesses' needs. This is why it is recommended that a platform that can follow the modular approach be chosen to avoid compromising integration with the other systems while developing the new software.

Lastly, organizations must make changes to build LCDP competencies by training and supporting developers and business users. It allows the organization to collaborate in, novate, and remain relevant in the fast-changing technology environment.

7. Conclusion

7.1. Summary of Key Points

In this study, it has been possible to present LCDPs as the tool that helps create new tech like AI, IoT, and blockchain. Based on the research, there is evidence that LCDPs have a positive impact in terms of time and cost reduction in development with a corresponding increase in users' satisfaction level, especially as supported by the detailed analyses of the AI chatbot and IoT application use cases. However, issues highlighted included security, scalability, and user experience challenges, especially within the blockchain example. As a result, to get the maximum out of LCDPs, organizations should care about security, include end-users in further development, and consider scalability aspects important. When these challenges are well handled, LCDPs will enable businesses to maximize such technologies for innovation and sustain competition within the unfolding digital environment.

7.2. Future Directions

Based on these future directions, several future research directions can be selected to improve the LCDPs for developing emerging technologies.

- First, there is a new requirement: better interoperability with existing enterprise applications and other third-party applications. In the future LCDPs, the focus should be on better compatibility of the solutions used by organizations so they can link multiple technologies together, which would help implement more complex systems.
- Secondly, for the same reason, any future expansion of LCDPs should incorporate greater improvements in security aspects. It entails automatic security, assessments, and easy to configure security measures to protect relevant information and address regulations.
- But it is always a crease to enhance the user experience. The next generations of LCDPs might improve results presentation with recommendations from artificial intelligence and machine learning technologies, autonomous program debugging, and user interfaces could be specialized to individual developers.
- Last, the community engagement approach and user collaboration can foster innovation within LCDPs. That is because the code is developed together, and modules, templates, and other similar essentials that constitute the application are shared – it builds a healthy community. As a result, the application gets richer and more profound.

In reply to these trends for future development, LCDPs may go a long way to transform into forms that can fit into helping organizations in the increasingly digital and technological age.

References

[1] Atzori, L., Iera, A., & Morabito, G. (2010). The Internet of Things: A survey. *Computer Networks*, 54(15), 2787–2805. https://doi.org/10.1016/j.comnet.2010.05.010

[2] Bosch, J. (2017). Speed, data, and ecosystems: The future of software engineering. *IEEE Software*, 33(1), 82–88. https://doi.org/10.1109/MS.2015.11

[3] Dean, J., & Ghemawat, S. (2008). MapReduce: Simplified data processing on large clusters. *Communications of the ACM*, 51(1), 107–113. https://doi.org/10.1145/1327452.1327492

[4] Fowler, M. (2010). *Domain-Specific Languages*. Addison-Wesley Professional. https://martinfowler.com/books/dsl.html

[5] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press. http://www.deeplearningbook.org/

[6] Gupta, S. K., & Pal, S. (2018). Low-code platforms for AI development. *International Journal of Artificial Intelligence and Applications*, 9(1), 1–10. https://doi.org/10.5121/ijaia.2018.9101

[7] Guinard, D., Trifa, V., Mattern, F., & Wilde, E. (2011). From the Internet of Things to the Web of Things: Resource-oriented architecture and best practices. In *Architecting the Internet of Things* (pp. 97–129). Springer. https://doi.org/10.1007/978-3-642-19157-2_5

[8] Mendix. (2018). The Total Economic Impact™ of the Mendix Low-Code Application Development Platform. Forrester Consulting. https://www.mendix.com/forrester-tei-study/

[9] Patel, P., & Cassou, D. (2015). Enabling high-level application development for the Internet of Things. *Journal of Systems and Software*, 103, 62–84. https://doi.org/10.1016/j.jss.2015.02.027

[10] Pallis, G., & Vakali, A. (2017). Software reuse in low-code platforms: Benefits and challenges. *ACM Computing Surveys*, 49(4), Article 68. https://doi.org/10.1145/0000000.0000000

[11] Richardson, C., & Rymer, J. R. (2016). Low-Code Platforms Deliver Customer-Facing Apps Fast, But Will They Scale Up? Forrester Research.

[12] Rymer, J. R. (2017). The Pitfalls and Possibilities of Low-Code Development Platforms. Forrester Research. https://www.forrester.com/report/The-Pitfalls-And-Possibilities-Of-LowCode-Development-Platforms/RES116613

[13] Sahay, S. K., & Gupta, S. (2013). A low-code development platform for rapid development of multichannel enterprise applications. In *2013 6th International Conference on Contemporary Computing (IC3)* (pp. 181-186). IEEE. https://doi.org/10.1109/IC3.2013.6612186

[14] Stake, R. E. (1995). *The Art of Case Study Research*. SAGE Publications. http://dx.doi.org/10.4135/9781412983587

[15] Swan, M. (2015). *Blockchain: Blueprint for a New Economy*. O'Reilly Media. https://www.oreilly.com/library/view/blockchain/9781491920489/

[16] Zhang, L., Yang, H., & Wang, Y. (2015). A visual programming platform for rapid development of IoT applications. *International Journal of Software Engineering and Knowledge Engineering*, 25(9–10), 1483–1506. https://doi.org/10.1142/S021819401550032X

[17] Zheng, Z., Xie, S., Dai, H., Chen, X., & Wang, H. (2017). An overview of blockchain technology: Architecture, consensus, and future trends. In *2017 IEEE International Congress on Big Data (BigData Congress)* (pp. 557–564). IEEE. https://doi.org/10.1109/BigDataCongress.2017.85

[18] Rahman, M.A., Butcher, C. & Chen, Z. Void evolution and coalescence in porous ductile materials in simple shear. Int J Fracture, 177, 129–139 (2012). https://doi.org/10.1007/s10704-012-9759-2

[19] Rahman, M. A. (2012). Influence of simple shear and void clustering on void coalescence. University of New Brunswick, NB, Canada. https://unbscholar.lib.unb.ca/items/659cc6b8-bee6-4c20-a801-1d854e67ec48

[20] Krishna, K. (2022). Optimizing query performance in distributed NoSQL databases through adaptive indexing and data partitioning techniques. International Journal of Creative Research Thoughts (IJCRT). https://ijcrt. org/viewfulltext. php.

[21] Krishna, K., & Thakur, D. (2021). Automated Machine Learning (AutoML) for Real-Time Data Streams: Challenges and Innovations in Online Learning Algorithms. Journal of Emerging Technologies and Innovative Research (JETIR), 8(12).

[22] Murthy, P., & Thakur, D. (2022). Cross-Layer Optimization Techniques for Enhancing Consistency and Performance in Distributed NoSQL Database. International Journal of Enhanced Research in Management & Computer Applications, 35.

[23] Murthy, P., & Mehra, A. (2021). Exploring Neuromorphic Computing for Ultra-Low Latency Transaction Processing in Edge Database Architectures. Journal of Emerging Technologies and Innovative Research, 8(1), 25-26.

[24] Thakur, D. (2021). Federated Learning and Privacy-Preserving AI: Challenges and Solutions in Distributed Machine Learning. International Journal of All Research Education and Scientific Methods (IJARESM), 9(6), 3763-3764.