Decision Making in Manufacturing and Services
Vol. 20 • 2026 • No. 1 • pp. 77–90

The Evolution of BPM Platforms towards Low-Code – New Opportunities and Challenges for Businesses and Software Providers

Agata Chodowska-Wasilewska*

Abstract. The current transformation of classic business process management (BPM) systems into innovative low-code platforms is changing the way companies can automate and implement business processes and applications. Low-code platforms enable rapid development of software solutions and allow business users to create and modify workflows or even small applications with minimal programming skills. This technology has a lot of advantages, including increased agility, cost savings, and faster time-to-market. However, it also introduces challenges such as integration difficulties and the need to upskill employees. On the other hand, from the perspective of software companies, developing low-code platforms presents a lot of new challenges compared to traditional applications. Building a stable and flexible environment requires advanced architecture planning, intuitive user interfaces, and a powerful application engine that is simple to use but still scalable. We should also not forget about security, modularity, reusability, and integration issues. Using a qualitative, descriptive research methodology along with a thorough literature review, conceptual analysis and case study this article explores the challenges behind the transition from BPM to low-code, its impact on business organisations and technology providers, and the critical factors for successful implementation.

Keywords: Business Process Management, BPM, process automation, digital transformation, enterprise software, workflow management, IT governance, software development, technology providers, integration challenges, scalability

Mathematics Subject Classification: 68M01, 68N30, 90B50, 68U35

JEL Classification: L86, O31, O32, M15

Submitted: September 18, 2025

Revised: October 29, 2025

* Military University of Technology, Cybernetics Faculty, Warsaw, Poland., e-mail: agata.chodowska@gmail.com. ORCID: 0009-0006-2134-5851.

DOI: https://doi.org/10.7494/dmms.2026.20.1.7590

1. INTRODUCTION

In recent years, organisations have increasingly turned to business process management (BPM) and low-code development platforms (LCDPs) to accelerate digital transformation. While BPM provides a structured methodology for analysing and optimising business processes, low-code platforms enable rapid application development through visual interfaces and automation tools (Dumas et al., 2018; Waszkowski & Bocewicz, 2022). The convergence of these two paradigms has reshaped enterprise IT strategies by allowing both professional developers and business users to participate in process innovation.

The current wave of digital transformation emphasises agility and user empowerment. According to Gartner (2024a), the low-code application platforms market is forecast to reach USD 16.5 billion by 2027, with a CAGR of 16.3% for 2022–2027. This rapid expansion coincides with the emergence of artificial intelligence (AI)-assisted no-code tools that integrate process modelling with generative technologies, further democratising development (Forrester, 2024). Consequently, understanding how BPM principles are embedded within modern low-code platforms has become a critical area of both research and practice.

This paper aims to analyse the evolution of BPM towards low-code architectures, exploring the conceptual, technological, and organisational dependencies between the two domains. The research identifies the extent to which BPM constructs (such as process modelling, orchestration, and optimisation) are translated into low-code environments. The findings contribute to the theoretical understanding of the evolution of BPM and provide practical insights for organisations adopting low-code solutions to automate complex business processes.

2. RESEARCH METHODOLOGY

This article uses a qualitative, descriptive research methodology centred on a broad literature review and conceptual analysis. Given the relatively new relation between BPM and low-code platforms, a primary empirical study (e.g., case studies) would provide specific insights but struggles to show the full picture of the transformation of the industry at the macro level. Therefore, a synthesis of existing academic and industry literature provides a solid foundation for understanding the current state, trends, and implications.

The research used a systematic literature review and conceptual analysis based on academic databases (Scopus, Web of Science, IEEE Xplore, ACM DL) and industry sources (Forrester, 2023; Gartner, 2024a; 2024b). A qualitative case study followed the principles outlined by Yin (2018), combining semi-structured interviews with product teams, documentation, and a comparative analysis of a low-code platform before and after its transformation. Data were triangulated across literature sources and empirical evidence. The conceptual framework emerged through iterative coding and pattern matching to existing BPM–low-code theories. Analytical generalisation, rather than statistical inference, guided interpretation.

This study follows a **qualitative, descriptive design**, integrating literature synthesis with an embedded particular case study. The goal is to conceptualise the evolution

of BPM platforms towards low-code systems by linking theoretical insights with an empirical industry example. The following research objectives and questions were identified:

Research Objectives

- To analyse the conceptual relationship between BPM and low-code platforms.
- To identify opportunities and challenges emerging from this convergence for both businesses and software providers.
- To illustrate these dynamics through the case study of Aurea BPM's transformation into the Aurea Low-Code Platform.

Research Questions

- RQ1: What are the key differentiating characteristics between traditional BPM and low-code platforms?
- RQ2: What opportunities and challenges arise for organisations adopting low-code solutions?
- RQ3: How do software providers adapt their architectures and processes when transitioning from BPM to low-code platforms?

The research approach involved combining theoretical and empirical components to ensure an understanding of the evolution of BPM platforms towards low-code solutions. The approach was based on the following steps:

- Systematic literature search – identification of key academic papers, industry reports, and reputable publications focusing on business process management, low-code development platforms, rapid application development, enterprise architecture, software engineering, and digital transformation. Databases such as Scopus, Web of Science, IEEE Xplore, ACM Digital Library, and prominent industry analyst reports (e.g., Forrester, 2023; Gartner 2024a) were researched. Keywords included "BPM evolution", "low-code platforms", "no-code", "process automation", "citizen developer", "enterprise architecture", "software provider challenges", and *"digital transformation".
- Conceptual analysis – critical examination and synthesis of the identified literature to extract key concepts, identify trends, analyse stated benefits and challenges.
- Identification of opportunities and challenges – analysing the impact of specific opportunities and challenges for both businesses and software companies. This involved mapping the main functions of low-code against traditional BPM.
- Discussion of success factors – based on the analysed literature, formulation of key factors that are essential for the successful implementation and adoption of low-code solutions.
- Case study – to better understand the theoretical findings in a real-world context, an embedded particular case study was conducted. The selected case concerns Tecna sp. z o.o. (www.tecna.pl), a Polish software company specialising in business process automation, workflow systems, and enterprise software solutions. One of the company's core products is Aurea, originally developed as a BPM platform and subsequently redesigned into a broader low-code environment. Therefore, the transformation of Aurea BPM into the Aurea Low-Code Platform was selected as an empirical example of the BPM-to-low-code transition analysed in this paper.

The case provides a practical illustration of how a traditional BPM system can evolve into a modern low-code platform. The study draws on internal documentation, interviews with the product team, and an analysis of platform capabilities across both stages of development. Key dimensions examined include: (1) architectural complexity, (2) user interface, (3) flexibility of modelling, and (4) implications for software delivery workflows. The Aurea case highlights both the strategic motivations for transformation, such as meeting market demand for rapid application development, and the technical and organisational challenges, particularly in areas such as component reusability, developer enablement, and platform governance. The case study approach followed the methodological guidance of Yin (2018), emphasising real-world context, multi-source evidence collection, and analytical generalisation over statistical inference.

3. LITERATURE REVIEW

The evolution of BPM and low-code development have increasingly gained academic and industry attention. An understanding of their connections requires examining the foundational principles of both BPM and low-code platforms. Recent literature also treats low-code platforms as a distinct research area. Bock and Frank (2021) conceptualise low-code platforms from the perspective of information systems research, while Pinho et al. (2023) identify usability as one of the key issues in low-code platform adoption. In addition, Di Ruscio et al. (2022) discuss the relationship between low-code development and model-driven engineering, which is particularly relevant for BPM-based platform evolution.

3.1. Foundations of business process management (BPM)

BPM, as a discipline, has been extensively studied, with the literature defining its scope, methodologies, and benefits. Dumas et al. (2018), in their seminal work Fundamentals of Business Process Management, provide a comprehensive overview of BPM concepts, including process modelling, analysis, automation, and monitoring. They emphasise BPM's role in aligning business strategy with operational execution and driving continuous improvement. Similarly, van der Aalst (2013), in 'Business Process Management: A comprehensive survey', outlines the historical development, key technologies, and future directions of BPM, highlighting its evolution from workflow management systems to advanced process orchestration platforms. Traditional BPM platforms are often characterised by their emphasis on formal modelling notations (e.g., BPMN), deep integration capabilities, and robust execution engines, typically requiring skilled IT professionals for implementation and maintenance (van der Aalst, 2013; Dumas et al., 2018).

In the industrial context, BPM has been successfully applied to model and improve complex maintenance and production processes (Jasiulewicz-Kaczmarek et al., 2018; Waszkowski et al., 2018). These studies confirm the enduring role of BPM as a methodological backbone for process optimisation, providing the foundation upon which modern automation and low-code solutions are now being built.

3.2. The rise of low-code development

The concept of taking away coding tasks has its roots in earlier rapid application development (RAD) methodologies. However, the modern low-code movement, as described by Forrester Research, presents a more generalised and widely accessible approach to software development. Forrester (2023), articulate the core foundations of low-code: visual development environments, model-driven development, and the ability to generate deployable applications with minimal manual coding. They highlight the primary drivers for low-code adoption, including the need to accelerate digital transformation, reduce IT backlogs, and empower business users (Forrester, 2023). Waszkowski (2019), in his article 'Low-code platform for automating business processes in manufacturing', provides an empirical perspective on the application of low-code specifically in the manufacturing sector, demonstrating its practical use for automating specific business processes. This work highlights how low-code can be leveraged to address specific industry needs, offering an example of its application beyond standard enterprise solutions (Waszkowski, 2019).

Recent analyses further highlight the growth of the low-code market. According to Gartner (2024a), the low-code application platforms market is forecast to reach USD 16.5 billion by 2027, with a CAGR of 16.3% for 2022–2027. A notable trend is the convergence of low-code and AI-driven no-code approaches, where generative AI supports automatic code generation, process modelling, and natural-language automation design (Forrester, 2024). This integration is redefining the boundaries of process automation, enabling even non-technical users to co-create intelligent digital solutions. These dynamics confirm the increasing strategic importance of low-code and its transformative potential across sectors.

3.3. Dependencies between BPM and low-code platforms

The overlap between business process management (BPM) and low-code technologies reflects a shift towards greater agility in business operations. Traditional BPM systems, while effective in modelling and enforcing structured processes, have often been seen as inflexible, particularly due to the level of IT involvement needed to implement even minor changes. Low-code platforms have emerged as a response to these limitations, offering a more user-friendly and iterative approach to process automation.

Still, bringing these two paradigms together has its challenges. Governance remains one of the critical challenges in low-code adoption, especially in enterprise environments where citizen development may create risks related to shadow IT, security, application lifecycle control and platform standardisation (Heuer et al., 2022; Viljoen et al., 2024). While low-code platforms can empower non-technical users and accelerate delivery, this shift also requires a rethinking of IT's role – from implementer to strategic enabler and overseer.

Recent research further refines the understanding of how low-code platforms operationalise BPM concepts. Waszkowski and Bocewicz (2022) proposed the *Visibility Matrix* model, which improves user interface modelling efficiency in low-code environments, while Waszkowski and Dybowski (2023) demonstrated techniques for generating complex, dynamic forms to enhance usability and flexibility in enterprise systems.

These contributions illustrate the practical maturation of low-code design patterns and their growing alignment with BPM principles. Low-code has also been analysed as an enabler of digital transformation in the manufacturing industry, particularly in relation to process automation and industrial digitalisation (Sanchis et al., 2020).

Moreover, market analyses underline the accelerating adoption of these technologies. Moreover, market analyses underline the accelerating adoption of these technologies (Gartner, 2024a). Simultaneously, Forrester (2023) and Forrester (2024) note a shift towards AI-augmented no-code platforms. Simultaneously, Forrester (2023) and Forrester (2024) note a shift towards *AI-augmented no-code* platforms that integrate generative design capabilities. This convergence of BPM, low-code, and AI represents a paradigm shift in enterprise automation, broadening participation in application development and fostering more adaptive, data-driven process ecosystems (Table 1).

Table 1. *Comparison of BPM and low-code platforms*

Aspect	Traditional BPM platforms	Low-code platforms
Primary focus	Process modelling, automation, and orchestration	Rapid application development with minimal hand-coding, not necessarily using a BPM engine
Target users	Business analysts, process architects, IT professionals	Business users, citizen developers, and professional developers
Development approach	Model-driven (BPMN, process flows)	Visual, component-based (drag-and-drop UI, data models, logic blocks)
Level of abstraction	Process-centric	Application-centric
Customisation capability	High, but often requires coding (Java, XML, scripting)	Moderate to high, with built-in extensions and optional code
Integration	Strong support for enterprise integration (ESB, SOA, connectors)	APIs, REST/SOAP connectors, easier plug-in of third-party services
Governance and compliance	Strong emphasis on process governance and auditing	Increasingly integrated governance models
Scalability and performance	Depend primarily on system architecture and underlying infrastructure; typically engineered for long-running workflows	Similarly dependent on infrastructure; modern cloud-native low-code platforms can achieve high scalability but vary by vendor
Time-to-market	Longer implementation cycles (but still shorter than traditional software development)	Shorter time – applications can be developed and deployed within days/weeks
Business agility	Moderate – changes often may require IT support	High – enables fast iterations and adjustments by non-technical users

Table 1 (cont.)

Entry threshold	Requires knowledge of BPMN notation and process modelling, often demanding IT expertise	Offers a lower entry barrier through visual design; however, a basic understanding of process modelling and application logic remains necessary
AI/ML integration	Limited, often with usage of external tools	Growing support for built-in AI features (e.g., predictions, automation)
Cost of ownership	High initial setup and maintenance costs	Lower upfront costs. Licensing may vary by vendor

Source: own analysis and adaptation based on market analysis and literature review (e.g., Bock & Frank, 2021; Dumas et al., 2018; Gartner, 2024b; Waszkowski & Bocewicz, 2022).

4. DISCUSSION

The evolution of BPM platforms towards low-code is a big change with implications for both businesses and software providers. This discussion elaborates on these opportunities and challenges, highlighting the critical factors for navigating this transition successfully.

4.1. Opportunities for businesses

The primary advantage of low-code platforms for businesses lies in accelerated digital transformation and enhanced agility.

- Increased agility and time to market – low-code platforms reduce the development time for new applications and process automations (Forrester, 2023). This allows businesses to prototype, test, and deploy solutions in response to evolving market conditions, customer demands, or regulatory changes. The visual nature of the development method means that changes can be implemented swiftly, shortening the feedback loop and enabling continuous adaptation, a key competitive advantage in today's fast-paced environment.
- Democratisation of development – by empowering 'citizen developers' – business users with domain expertise but limited coding and technical skills – low-code platforms bridge the traditional gap between business needs and IT delivery. This brings innovation, as individuals closest to the operational challenges can directly contribute to their digital solutions. As an example, Waszkowski (2019) demonstrated how low-code platforms can be successfully applied to automate specific business processes, even in complex manufacturing environments. This also helps in reducing the IT backlog, freeing up specialised IT resources for more complex, strategic initiatives.

- Cost reduction and ROI – while initial platform investments exist, low-code can significantly reduce the overall cost of software development and maintenance. Fewer highly specialised developers are needed for routine tasks, with the reusability of components. Predefined components are already a core of low-code platforms and reduce the initial cost of implementing basic application features (e.g. search engine, standard views, administrative panels, user management). Faster deployment also means a quicker return on investment from process automation projects (Forrester, 2023).
- Improved operational efficiency – by automating repetitive tasks, standardising workflows, and providing greater visibility into process execution, low-code platforms offer improvements in operational efficiency. This leads to reduced manual errors, improved data quality, and enhanced overall productivity across the organisation (Dumas et al., 2018).

4.2. Challenges for businesses

Despite the significant opportunities, adopting low-code has its implementation challenges.

- Governance and 'shadow IT' – as highlighted by Heuer et al. (2022); Viljoen et al. (2024), one of the most critical challenges is establishing effective governance. The ease of development can lead to uncontrolled implementation of applications ('shadow IT') outside of central IT oversight. This raises concerns about security vulnerabilities, data silos, compliance risks, and the potential for inconsistent user experiences. A governance framework, clear policies, and centralised IT involvement are essential to manage these risks (Heuer et al., 2022; Viljoen et al., 2024).
- Integration with legacy systems – while modern low-code platforms offer robust API capabilities, integrating new low-code applications with monolithic legacy systems can still be complex and resource-intensive, and sometimes even impossible. Orchestrating complex business processes that integrate both new and old systems requires careful planning and often custom integration work (Dumas et al., 2018).
- Scalability and performance concerns – while many enterprise-grade low-code platforms are designed for scalability, certain applications or extreme transaction volumes might still push the limits of some platforms. Businesses must carefully evaluate the platform's ability to handle workloads and complex business logic without compromising performance.
- Security and compliance – the simplified development environment must not compromise security. Ensuring that applications built by citizen developers meet the necessary security protocols, data privacy regulations (e.g., GDPR), and industry-specific compliance requirements is crucial. This necessitates built-in security features within the platform and diligent oversight (Heuer et al., 2022; Viljoen et al., 2024).
- Upskilling and change management – while low-code reduces coding requirements, it necessitates new skills in business process modelling, solution design, and understanding of platform capabilities. Organisations must invest in training programs for business users and redefine roles within IT to support citizen developers.

4.3. Opportunities for software providers

For software companies, the shift towards low-code from BPM presents a foundation for growth and innovation.

- Expanding market reach – low-code platforms enable providers to enter a much larger market segment, including organisations that previously lacked the resources or technical expertise for traditional BPM implementations. This includes small and medium-sized enterprises (SMEs) and specific business units within large corporations (Forrester, 2023).
- Increased demand for platform services – the need to quickly adapt to digital changes is creating a strong demand for scalable, user-friendly low-code platforms. Providers who can offer such solutions with strong integration capabilities, intuitive UIs, and powerful engines are well-positioned for market leadership (Forrester, 2023).
- New business models and revenue streams – new opportunities for professional services (consulting, implementation), training, and a marketplace for pre-built components and industry-specific solutions.
- Innovation in platform features – the competitive landscape drives innovation. Providers are constantly enhancing platforms with features like artificial intelligence/machine learning (AI/ML) integration for intelligent process automation (IPA), process mining capabilities for data-driven optimisation, and enhanced collaboration tools, pushing the boundaries of what low-code can achieve.

4.4. Challenges for software providers

Developing and maintaining low-code platforms introduces significant complexities for software providers.

- Balancing simplicity and power – the fundamental challenge is designing a platform that is intuitive and simple enough for citizen developers while simultaneously powerful and flexible enough to build complex, critical enterprise applications. This requires sophisticated architectural decisions regarding abstraction layers, modularity, and extensibility.
- Robust architecture and engine design – building a stable, scalable, and high-performance application engine that can execute diverse workflows and integrate with various systems is a monumental task. The engine must support concurrent processes, handle large data volumes, and provide reliable uptime, while remaining 'simple in use' for the end-user.
- Comprehensive integration capabilities – for a low-code platform to be truly effective, it must offer extensive and easy-to-use integration capabilities with a wide array of third-party applications, databases, and cloud services. This needs API management, connectors, and support for various integration patterns.
- Security by design – given the potential for widespread adoption by non-technical users, providers must embed security measures into the core of the platform's design. This includes robust authentication, authorisation, data encryption, vulnerability management, and audit trails (Heuer et al., 2022; Viljoen et al., 2024).

– Modularity and reusability – to maximise the benefits of low-code, platforms must facilitate the creation and reuse of components, templates, and business logic. This requires sophisticated design patterns and development tools that promote modularity and enable version control.
– Evolving technological landscape – the rapid pace of technological change, particularly in areas like artificial intelligence and machine learning, requires continuous innovation and integration into low-code platforms. Providers must invest heavily in R&D to remain competitive.

4.5 Case study – transformation from BPM to low-code at Tecna

In the case study of Tecna's platform transformation, the evolution from Aurea BPM to the Aurea Low-Code Platform serves as a practical example of the broader industry shift from traditional BPM architectures to more sophisticated, low-code solutions. The original Aurea BPM system (Fig. 1) was built around a modular but streamlined architecture, focused on essential BPM functionalities such as process modelling, task management, service orchestration, reporting, and registry management, underpinned by core components like the BPM engine and form/document services.

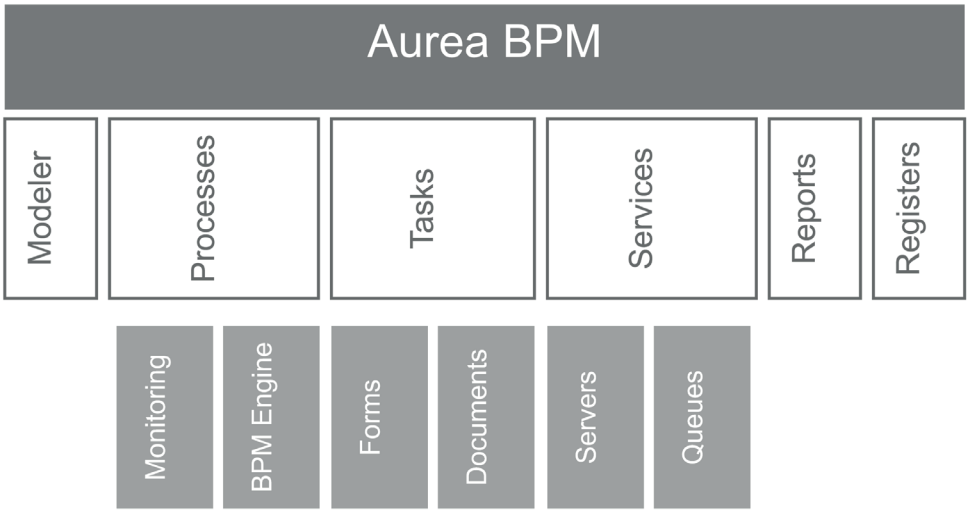

Fig. 1. *Architecture of the traditional Aurea BPM platform*

In contrast, the new Aurea Low-Code Platform (Fig. 2) demonstrates a significant increase in architectural complexity and technological depth. It incorporates microservices architecture, multiple frontends (including Vue.js microfrontends for runtime and modeller components), identity and access management via Keycloak, and a Spring Cloud Gateway-based API orchestration layer. It also integrates advanced services such as notification systems, form builders, runtime engines, and

enhanced dashboard modules. The platform supports mobile clients and multiple distributed databases, reflecting the scalability and extensibility required by low-code paradigms.

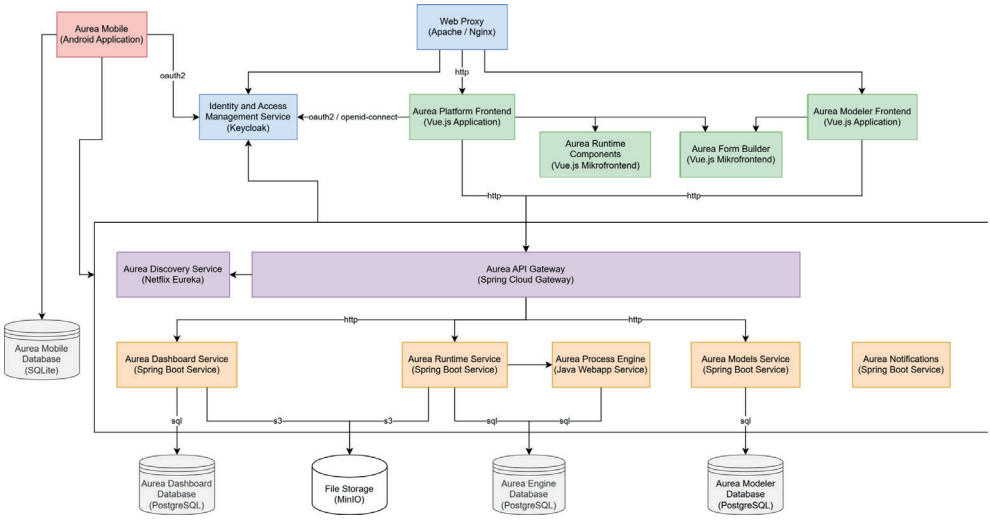

Fig. 2. *Architecture of the Aurea Low-Code Platform after transformation*

This transformation illustrates a key insight: while low-code platforms simplify application creation for end users and domain experts, they impose higher development and maintenance complexity for software providers. The migration required not only an architectural redesign but also a shift in engineering skills, deployment strategies, and DevOps tooling. It validates the findings in the literature that low-code development platforms introduce new technical and organisational challenges for vendors (Forrester, 2023; Sahay et al., 2020).

5. CONCLUSION

The journey of business process management (BPM) platforms towards low-code represents a pivotal moment in enterprise software. Low-code platforms offer significant opportunities for businesses to achieve faster time-to-market, reduce costs, foster innovation, and improve operational efficiency by democratising process automation.

However, this transformative shift is not without its challenges. For businesses, the adoption of low-code necessitates careful navigation of governance complexities to mitigate 'shadow IT', strategic planning for integration with existing legacy systems, and significant investment in upskilling employees. For software providers, the challenge lies in crafting platforms that balance user simplicity and enterprise-grade power, ensuring robust architecture, comprehensive security, and seamless integration capabilities in a rapidly evolving technological landscape.

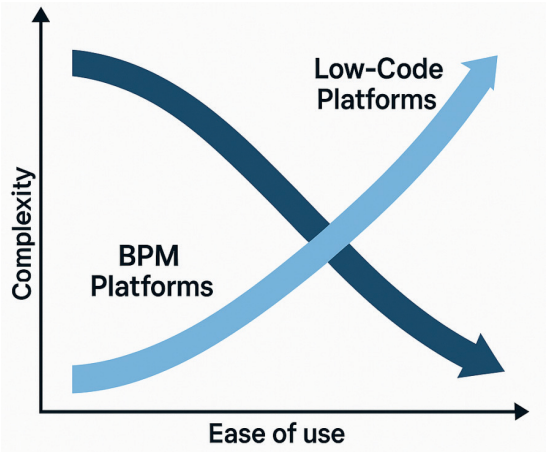

Fig. 3. *BPM and low-code comparison*

Figure 3 illustrates a two-dimensional comparison of BPM and low-code platforms along three axes: platform complexity, ease of use for end-users, and development difficulty for software providers. Low-code platforms are characterised by greater functional complexity and higher abstraction levels, which facilitate user-friendly interfaces and faster deployment for business users. However, this ease of use comes at the cost of increased architectural and technical challenges for software developers. In contrast, traditional BPM systems offer more straightforward, modular architectures that are easier to design and maintain from a development perspective but often require more technical involvement from end-users to configure and operate.

Ultimately, the successful implementation of low-code depends on several critical factors: establishing clear governance frameworks and procedures (Heuer et al., 2022; Viljoen et al., 2024), fostering collaboration between IT and business units, strategic integration planning, continuous skill development, and selecting platforms that are scalable, secure, and aligned with organisational needs.

REFERENCES

Aalst, van der, W.M.P. (2013). Business process management: A comprehensive survey. *ISRN Software Engineering*, **2013**, 507984, pp. 1–37. **DOI:** https://doi.org/10.1155/2013/507984.

Bock A.C. & Frank U. (2021). Low-code platform. *Business & Information Systems Engineering*, **63**(6), pp. 733–740. **DOI:** https://doi.org/10.1007/s12599-021-00726-8.

Di Ruscio D., Kolovos D., de Lara J., Pierantonio A., Tisi M. & Wimmer M. (2022). Low-code development and model-driven engineering: two sides of the same coin? *Software and Systems Modelling*, **21**, pp. 437–446. **DOI:** https://doi.org/10.1007/s10270-021-00970-2.

Dumas M., La Rosa M., Mendling J. & Reijers H.A. (2018). *Fundamentals of Business Process Management*. 2nd Edition. Cham: Springer. **DOI:** https://doi.org/10.1007/978-3-662-56509-4.

Forrester. (2023, June 20). *The Forrester Wave™: Low-Code Development Platforms for Professional Developers, Q2 2023*. Forrester Research. **URL:** https://www.forrester.com/report/the-forrester-wave-tm-low-code-development-platforms-for-professional-developers-q2-2023/RES178497 [18.09.2025].

Forrester. (2024, May 7). *The rise of application generation platforms.* Forrester Research. **URL:** https://www.forrester.com/report/the-rise-of-application-generation-platforms/RES180894 [18.09.2025].

Gartner. (2024a, May 24). *Risk and Opportunity Index: Low-Code Application Platforms* (ID 5459763). Gartner. **URL:** https://www.gartner.com/en/documents/5459763 [18.09.2025].

Gartner. (2024b, October 16). *Magic Quadrant for Enterprise Low-Code Application Platforms*. Gartner. **URL:** https://www.gartner.com/en/documents/5844247 [18.09.2025].

Heuer M., Kurtz C. & Böhmann T. (2022). Towards a governance of low-code development platforms using the example of Microsoft Power Platform in a multinational company. In: *Proceedings of the 55th Hawaii International Conference on System Sciences (HICSS)*, pp. 6881–6889. **DOI:** https://doi.org/10.24251/hicss.2022.831 [18.09.2025].

Jasiulewicz-Kaczmarek M., Waszkowski R., Piechowski M. & Wyczółkowski R. (2018). Implementing BPMN in maintenance process modeling. In: Świątek J., Borzemski L. & Wilimowska Z. (Eds.), *Information Systems Architecture and Technology: Proceedings of 38th International Conference on Information Systems Architecture and Technology – ISAT 2017*. Advances in Intelligent Systems and Computing, **656**, pp. 300–309. Cham: Springer. **DOI:** https://doi.org/10.1007/978-3-319-67229-8_27.

Pinho D., Aguiar A. & Amaral V. (2023). What about the usability in low-code platforms? A systematic literature review. *Journal of Computer Languages*, **74**, 101185. **DOI:** https://doi.org/10.1016/j.cola.2022.101185.

Sahay A., Indamutsa A., Di Ruscio D. & Pierantonio A. (2020). Supporting the understanding and comparison of low-code development platforms. In: *Proceedings of the 46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA 2020)*, pp. 171–178. IEEE. **DOI:** https://doi.org/10.1109/SEAA51224.2020.00036.

Sanchis R., García-Perales Ó., Fraile F. & Poler R. (2020). Low-code as enabler of digital transformation in manufacturing industry. *Applied Sciences*, **10**(1), 12. **DOI:** https://doi.org/10.3390/app10010012.

Viljoen A., Radić M., Hein A., Nguyen J. & Krcmar H. (2024). Governing citizen development to address low-code platform challenges. *MIS Quarterly Executive*, **23**(3), pp. 305–324. **DOI:** https://doi.org/10.17705/2msqe.00100.

Waszkowski R. (2019). Low-code platform for automating business processes in manufacturing. *IFAC-PapersOnLine*, **52**(10), pp. 376–381. **DOI:** https://doi.org/10.1016/j.ifacol.2019.10.060.

Waszkowski R. & Bocewicz G. (2022). Visibility matrix: efficient user interface modelling for low-code development platforms. *Sustainability*, **14**(13), 8103. **DOI:** https://doi.org/10.3390/su14138103.

Waszkowski R. & Dybowski M. (2023). Generating complex dynamic forms in low-code development platforms. In: Burduk A., Batako A., Machado J., Wyczółkowski R., Antosz K. & Gola A. (Eds.), *Advances in Production. Intelligent Systems in Production Engineering and Maintenance*. Lecture Notes in Networks and Systems, **790**, pp. 420–433. Cham: Springer. **DOI:** https://doi.org/10.1007/978-3-031-45021-1_32.

Waszkowski R., Nowicki T. & Worwa K. (2018). Corporate efficiency improvement with business process automation. *MATEC Web of Conferences*, **210**, 02012. **DOI:** https://doi.org/10.1051/matecconf/201821002012.

Yin R.K. (2018). *Case Study Research and Applications: Design and Methods*. 6th Edition. Thousand Oaks: SAGE Publications.