

Critical Success Factors for Low-Code / No-Code Adoption in Organizations: Insights from the Literature

Paul Rozbitski
0009-0008-5413-0165
Technische Hochschule
Mittelhessen (THM) – University
of Applied Sciences
THM Business School
Giessen, Germany
Email: paul.rozbitski@w.thm.de

Moritz Fabius Zeitfuchs
Technische Hochschule
Mittelhessen (THM) – University
of Applied Sciences
THM Business School
Giessen, Germany
Email: moritz.fabius.zeitfuchs@w.thm.de

Christian Leyh
0000-0003-0535-0336
Technische Hochschule
Mittelhessen (THM) – University
of Applied Sciences
THM Business School
Giessen, Germany
Email: christian.leyh@w.thm.de

Abstract—Low-code and no-code (LCNC) platforms are gaining importance in organizations because they accelerate solution development, enable business unit involvement, and reduce pressure on IT departments. Despite relevance, research on critical success factors (CSFs) for LCNC adoption remains fragmented. This paper addresses this gap through a systematic literature review. Fifty-one peer-reviewed publications identified through a database-supported search were analyzed; from these, 145 raw CSFs were extracted. These CSFs were consolidated into 25 CSFs, grouped into eight thematic clusters. The results confirm that LCNC adoption is a multi-stage process shaped not only by technological factors but also by organizational, strategic, and operational conditions. Governance-related CSFs highlight the importance of balancing decentralized development freedom with requirements for security, quality, and compliance. Overall, the findings support a socio-technical understanding of LCNC adoption and provide a foundation for research and practice.

Index Terms—Low-Code, No-Code, Critical Success Factors, CSF, Literature Review.

I. INTRODUCTION AND MOTIVATION

DIGITAL transformation is fundamentally changing business models, processes, and value-creation logic [1]; consequently, organizational demand for software solutions is steadily increasing, driven by new digital products, services, and automation requirements. At the same time, organizations face the challenge of scarce IT resources and growing demand for skilled workers in the software development sector, which slows down the implementation of digitalization projects [2], [3].

These developments have resulted in the growing importance of low-code and no-code (LCNC) platforms. LCNC platforms are development environments that are aimed at simplifying software development through visual and configuration-based tools, thereby making development more accessible to stakeholders outside professional IT teams [4]. An international survey of 2,000 companies reveals that 81% of decision-makers already consider LCNC development to be strategically significant, and 31% state that

LCNC is a central component of their software-development strategy [5]. Market studies estimate that the global LCNC market will reach a volume of approximately \$50 billion by 2028, which underscores the growing strategic relevance of these technologies [6].

However, research on information systems (IS) implementation demonstrates that the success of technology implementation depends less on the technology itself than on organizational conditions and the joint optimization of the social and technical components of an organization [1], [7]. Moreover, studies show that successful technology integration does not automatically lead to broad organizational adoption: despite the availability of LCNC platforms, organizations frequently report low levels of usage, fragmented initiatives, or an inability to realize the expected benefits [8].

While individual studies on LCNC address aspects (e.g., governance, citizen development, or platform selection) in isolation, a comprehensive synthesis and structured consolidation of critical success factors (CSFs) is still lacking. Thus, organizations introducing or scaling LCNC platforms lack an evidence-based synthesis that goes beyond vendor-driven best-practice recommendations. In contrast, CSF research for established types of IS is methodologically well-advanced: Leyh et al. [9], for example, consolidate 33 success factors for Enterprise Resource Planning (ERP) projects, drawing on 272 publications.

Against this background, our paper addresses the following research question:

What critical success factors can be identified for the implementation and adoption of low-code and no-code in organizations?

To answer this research question, our paper proceeds as follows. After the introduction, Section II provides a short theoretical background related to LCNC, citizen development, governance, and the interplay of IT and business in LCNC

contexts. Additionally, the concept of ‘critical success factor’s and relevant theoretical perspectives on CSF research are introduced. Section III describes the methodology of the systematic literature review, including the search strategy, search strings, and selection criteria. Section IV presents the results of the descriptive analysis, including the identified CSFs and the structural patterns linking them. In Section V, we discuss the results of our analysis. Finally, in Section VI, the paper concludes with a summary and an outlook regarding potential further research steps.

II. THEORETICAL BACKGROUND

A. Low-Code and No-Code Development

Following [10], this paper uses the term “LCNC development”, which refers to a development practice characterized by minimal coding effort, drag-and-drop functionality, and visual programming, thereby facilitating access to application development for users with varying technical backgrounds [11], [12], [13]. No-code (NC) is technically distinguished from low-code (LC), wherein NC avoids programming as much as possible while LC allows limited coding to customize pre-built components [10]; in practice, however, the two approaches are often considered together [10].

The distinction between LCNC and traditional software development lies primarily in the abstraction of the development logic: technical details of conventional programming and development infrastructure are largely concealed, enabling business users to work with more accessible development interfaces [14]. While professional software development typically involves the use of multiple specialized development tools, LCNC platforms bundle development, deployment, lifecycle, and platform-management functions into a single environment and generate executable applications from conceptual descriptions [4], [14].

As LCNC platforms are integrated, traditional roles in software development shift: alongside classic IT functions, citizen developers (CDs) and business unit developers emerge. These individuals are embedded in business departments, possess domain knowledge, and typically have only limited software development experience [15], and they develop applications for their own work environment outside of traditional IT structures [16], [17], [18]. This shift is accompanied by a redistribution of tasks between IT and business departments; while CDs frequently take on process automation, application logic, and prototyping, the role of IT shifts increasingly toward architectural responsibility, platform operations, quality assurance, and governance [3].

The typical use cases for LCNC in organizations are associated with the development of digital solutions, the integration of existing systems, and the automation of manual tasks [19], and the fields of application range from web and mobile applications to enterprise services and business processes, as well as Internet of Things (IoT) applications [20]. This indicates that LCNC is used in various organizational and sector contexts as supporting opportunities for digitalization, process support, and system integration.

B. Application of the CSF Approach to LCNC Contexts

LCNC platforms are not used in isolation within organizations but instead are embedded in existing processes, structures, and IT landscapes. Therefore, as with the implementation of other IS, their implementation and adoption involve considerable organizational and technical coordination efforts. Studies on ERP implementation reveal that such projects are regularly complex, resource-intensive, and fraught with risk and that their success depends on overcoming specific conditions or critical problem areas [21]. In the systematic handling of these areas critical to success, the consideration of CSFs can offer a suitable analytical approach, despite originally being developed in a management context.

CSFs are defined as a limited number of key areas in which satisfactory results ensure successful competitive performance by a company, an organizational unit, or an individual [22], [23]. While goals describe specific target states or desired endpoints, CSFs identify the areas of action in which adequate performance is necessary to achieve these goals. Key performance indicators (KPIs) or metrics are used for measurement within a CSF area. Thus, CSFs are neither goals nor metrics, but rather critical areas of action for success [22], [23].

In the classic CSF approach, the identification of CSFs is based primarily on subjective perceptions of individual managers. Bullen and Rockart [22] emphasize that CSFs are closely tied to a manager’s role, objectives, and situational context and cannot be determined via a fixed, objective algorithm. However, Boynton and Zmud [24] highlight the methodological risks of this subjective, management-centered approach. Moreover, the strong focus on the past in CSF research is problematic, as empirically identified factors are primarily based on historical data, meaning their validity in dynamic environmental conditions remains uncertain [25].

The CSF approach is thus not a universal model of success but rather a context-dependent analytical and control tool. Accordingly, the originally management-oriented approach has frequently been applied in IS research on the implementation of IS. For example, Ngai et al. [21] show that, in the context of ERP implementation, CSFs serve to identify critical problem areas that significantly influence implementation success. Against this background, the CSF approach is utilized in this study as a conceptual framework for identifying areas of action relevant to the implementation and embedding of LCNC in organizations [22], [25].

C. Conceptual Framework for Analyzing LCNC CSFs

The integration of LCNC is not merely a technical decision but rather a socio-technical transformation process, with organizational and strategic implications [1], [8], [26]. Accordingly, research distinguishes between the formal implementation of an LCNC solution and its actual adoption within the organization. While ‘implementation’ refers to the technical and organizational setup required to make an LCNC platform available, ‘adoption’ refers to the sustained use, acceptance, and embedding of the platform in work practices, decision-

making processes, and organizational structures. This distinction is important because merely making an LCNC platform available—that is, completing the implementation step—does not necessarily translate into sustained adoption. Indeed, organizations may still face limited platform use, isolated initiatives, or difficulties in achieving the anticipated benefits [8]. This indicates that LCNC adoption challenges are not limited to technological characteristics but are also strongly shaped by organizational shortcomings, unclear responsibilities, limited internal communication, insufficient strategic integration, and limited management support [2] [8]. Therefore, LCNC use should be understood as a multi-stage process ranging from pre-adoption and decision-making through implementation and introduction/adoption to post-adoption and scaling. In addition, LCNC research may take a holistic or lifecycle-wide perspective covering several or all of these stages. Based on this processual understanding, we classified the analyzed publications according to the LCNC phase they primarily address: pre-adoption/decision, implementation, introduction/adoption, or post-adoption/scaling. Studies covering several or all phases were coded as holistic/lifecycle, while studies without a clearly identifiable phase were classified as unspecified.

For a broad analytical orientation, we refer to the Technology-Organization-Environment (TOE) framework, which distinguishes among technological, organizational, and environmental adoption factors. In the context of citizen development, Hoogsteen and Borgman [8] further emphasize inter-organizational aspects, particularly the role of external platform providers and specialized third parties. However, the TOE framework is too broad to directly capture the temporal phases and analytical dimensions in which LCNC-related CSFs occur. Therefore, we use TOE as a starting point that we complement with a phase-based and dimension-based coding scheme.

In addition to this temporal classification, we use a four-dimensional analytical lens to classify the identified CSFs. Inspired by enterprise architecture perspectives (which link strategy, business processes, and IT infrastructure [27]) and by current LCNC studies (emphasizing the multidimensional nature of citizen development, maturity, scalability, and organizational adoption [28], [29], [30]), we classify CSFs along strategic, organizational, operational, and technical dimensions. Strategic factors comprise management support, organizational goals, and alignment with digital and IT strategies; organizational factors concern governance structures, roles, responsibilities, and coordination between IT and business units; operational factors encompass training, support, development processes, and knowledge sharing; and technical factors involve platform characteristics, integration, data access, security, and maintainability. Together, the phase-based classification and the four-dimensional CSF lens provide the conceptual basis for the subsequent literature analysis.

III. METHODOLOGY – SYSTEMATIC LITERATURE REVIEW

To identify CSFs for LCNC adoption in organizations, our study used a systematic literature review (SLR) following the concept-centered approach of Webster and Watson [31]. The procedure comprised three steps: **(1) literature search**, **(2) extraction of CSFs**, and **(3) consolidation of the identified CSFs**. In the first step, relevant articles were identified through a structured search process; in the second step, raw CSFs were extracted from the identified publications; in the third step, the extracted CSFs were consolidated (in cases of duplication or semantic similarity) and grouped into higher-level clusters. The individual steps are illustrated in Figure 1. The following subsections describe the three methodological steps in greater detail.

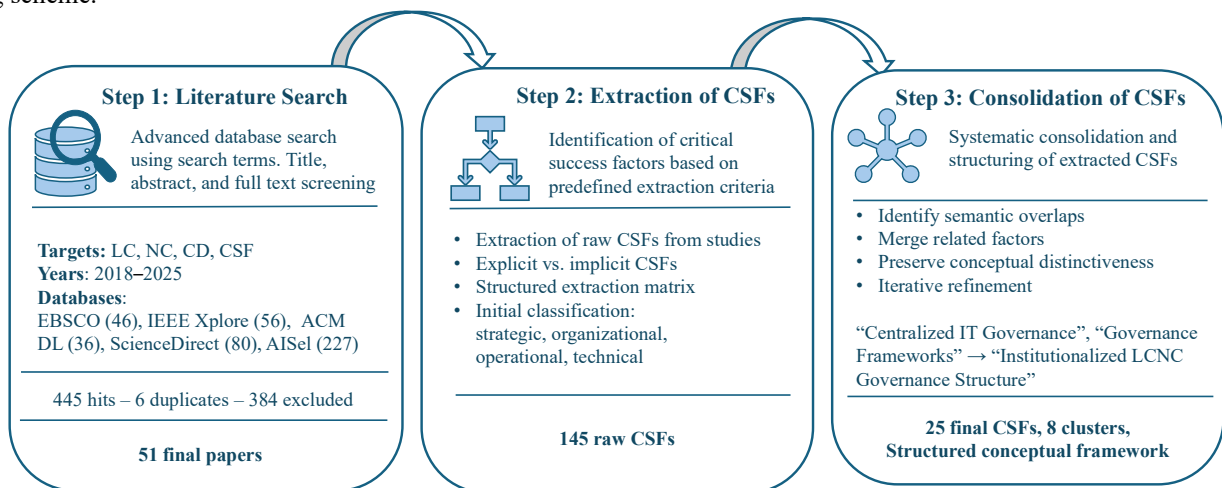


Fig. 1. Steps of literature review, from literature search to CSF extraction and consolidation (Step 1 abbreviations: LC = Low-code, NC = No-code, CD = Citizen Development, CSF = Critical Success Factor).

A. Literature Search

To find relevant publications, we searched in five databases: EBSCO, IEEE Xplore, ACM Digital Library (ACM DL), ScienceDirect, and AIS electronic Library (AISeL). These five databases were selected due to their subject-matter relevance to IS, technology, and management-science research. The search employed a standardized search string that combined terminological variants of LCNC as well as terms related to CSFs and implementation success. Minor adjustments to the search string due to database-specific characteristics were necessary; for example, in the ACM Digital Library, the term “Success” in connection with “Implementation” was removed because the combination resulted in a disproportionately high number of irrelevant hits, while in ScienceDirect, a shortened search string had to be used (see Table I).

TABLE I. DATABASES AND USED SEARCH STRINGS

Database(s)	Search Strings
EBSCO, IEEE Xplore, AISeL	(“Low Code” OR “Low-Code” OR “No Code” OR “No-Code” OR “LCNC” OR “Citizen Development” OR “Citizen Developer*”) AND (“Critical Success Factor*” OR “CSF*” OR “Success Factor*” OR “Implementation Success” OR “Adoption” OR “Enabler*” OR “Driver*” OR “Best Practice*”)
ACM DL	(“Low Code” OR “Low-Code” OR “No Code” OR “No-Code” OR “LCNC” OR “Citizen Development” OR “Citizen Developer*”) AND (“Critical Success Factor*” OR “CSF*” OR “Success Factor*” OR “Adoption” OR “Enabler*” OR “Driver*” OR “Best Practice*”)
ScienceDirect	(“Low Code” OR “No Code” OR “LCNC” OR “Citizen Develop”) AND (“CSF” OR “Success Factor” OR “Implementation” OR “Adoption”)

Although the term ‘low-code’ already appeared around 2014 in Gartner and Forrester reports, it took some time for the topic to become visible in academic publications. Accordingly, the starting point of 2018 was chosen, because it is only after this year that the academic discourse on LCNC-specific implementation issues shows a sufficient density of publications. Specifically, our literature review covered publications from 2018 through October 2025. Only German- and English-language academic publications with full texts were considered (due to the researchers’ background). A systematic backward and forward search was not conducted in this study.

The initial search yielded a total of 445 hits distributed across EBSCO (46), IEEE Xplore (56), ACM DL (36), ScienceDirect (80), and AISeL (227). After six duplicates were removed, a multi-stage analysis process was performed (see Fig. 1). First, a title screening was conducted, during which 278 publications were excluded. Thereafter, the abstracts of the remaining articles were examined; in this stage, an additional 106 publications were excluded because they had no discernible connection to CSFs in the context of LCNC projects. Finally, the remaining studies were subjected to a full-text review, during which four articles were excluded due to a lack of content relevance to the research question. At the

end of the selection process, a total of 51 scientific publications were included in the analysis, distributed across EBSCO (3), IEEE Xplore (9), ACM DL (4), ScienceDirect (2), and AISeL (33).

B. Extraction of CSFs

Relevant CSFs for LCNC adoption were identified inductively from the studies and subsequently amalgamated into overarching concepts, which resulted in a structured clustering of the identified CSFs. The identification of CSFs drew on several content-related criteria. These criteria were derived directly from the research question, which explicitly asks about factors that are relevant to and conditional for successful implementation and adoption at the organizational level. Accordingly, only statements with a direct connection to the success or failure of LCNC projects at the organizational level were considered as CSFs. Furthermore, an identified CSF had to represent an overarching condition for success; individual measures, specific tools, concrete implementation steps, or isolated technical functions were not classified as independent CSFs but rather interpreted as possible manifestations of overarching success mechanisms. During the identification process, care was taken to distinguish success-relevant conditions from statements merely describing outcomes or benefits; statements exclusively describing expected benefits or effects of LCNC technologies, without containing a corresponding logic of success, were not considered as CSFs.

The analysis also distinguished between explicit and implicit CSFs. Explicit CSFs are factors that the authors of the respective studies explicitly designate as success factors, drivers, prerequisites, or best practices; implicit CSFs emerge from statements in the studies that—in the context of the described results, discussions, or identified challenges—point to a condition for the success of LCNC projects, without it being explicitly designated as a success factor. Both CSF types were extracted and documented in research protocol.

Multiple statements in the same paper that were thematically related were combined into a paper-specific CSF, provided they addressed the same logic of success.

The identified raw CSFs were systematically documented in an extraction matrix; for each raw CSF, the designation, the corresponding text passage, the title, and the authors of the respective publication, as well as the data source, were recorded. In this stage, ‘raw CSFs’ refer to the success factors extracted directly from the analyzed studies, before any cross-paper consolidation was performed. An initial classification of these raw CSFs was performed across the four dimensions—strategic, organizational, operational, and technical—based on the theoretical lens outlined above.

C. Consolidation of CSFs

A total of 145 potential raw CSFs were derived from the literature in Step 2. In Step 3, these 145 raw CSFs were consolidated into 25 final CSFs (see Fig. 1, Step 3). The formation of the final CSFs was conducted inductively and gradually refined through an iterative comparison process. The 25

final CSFs were then grouped into eight higher-level clusters (see Table II).

The goals of Step 3 were to consolidate semantically similar raw CSFs and to transform the factors into a more concise and analytically meaningful structure. To this end, the raw CSFs were first compared to one another with an eye to their content, in order to identify overlaps and semantic similarities. Raw CSFs that describe the same organizational context, use synonymous terms, or address the same causal relationship were merged. Then, CSFs that target the same organizational control area (such as governance, qualification, or platform integration) were merged. In the third consolidation stage, we examined whether a merger would compromise the conceptual distinctiveness of the CSFs; in such cases, raw CSFs were kept separate.

This process can be illustrated through the example of the *Governance, Control, and Risk Management* cluster. The raw CSFs of “hybrid governance approach,” “collaborative and transparent governance structures,” “centralized IT governance,” “establishment of formal governance structures,” and “governance frameworks” address different facets of the same overarching governance need and were therefore merged into the final CSF “Institutionalized LCNC Governance Structure”. However, factors related to security and compliance requirements (“Security, Compliance, and Regulatory Management”) or to AI-specific quality control (“Responsible AI and

Quality Governance”) remained separate CSFs, as they addressed substantively distinct concerns that warranted independent treatment.

The 25 final CSFs that emerged after consolidation due to content similarities represent abstract categories bundling multiple success conditions described in the literature.

In the last stage of this step, the 25 final CSFs were grouped into eight higher-level clusters representing key areas of action in the context of LCNC implementation and adoption.

IV. RESULTS

This chapter presents the results of the systematic literature review. It first characterizes the identified studies, then outlines the distribution of CSFs across the clusters, and finally discusses the most frequently represented clusters in greater detail.

A. Characterization of the Identified Publications

Figure 2 summarizes the characteristics of the 51 articles included in our analysis. Their raw CSFs are concentrated primarily in the organizational and operational dimensions (32% each), followed by the strategic (19%) and then the technical (17%) dimensions. Methodologically, the field is dominated by qualitative research, although conceptual and mixed-methods studies are also represented; purely quantitative studies are rare.

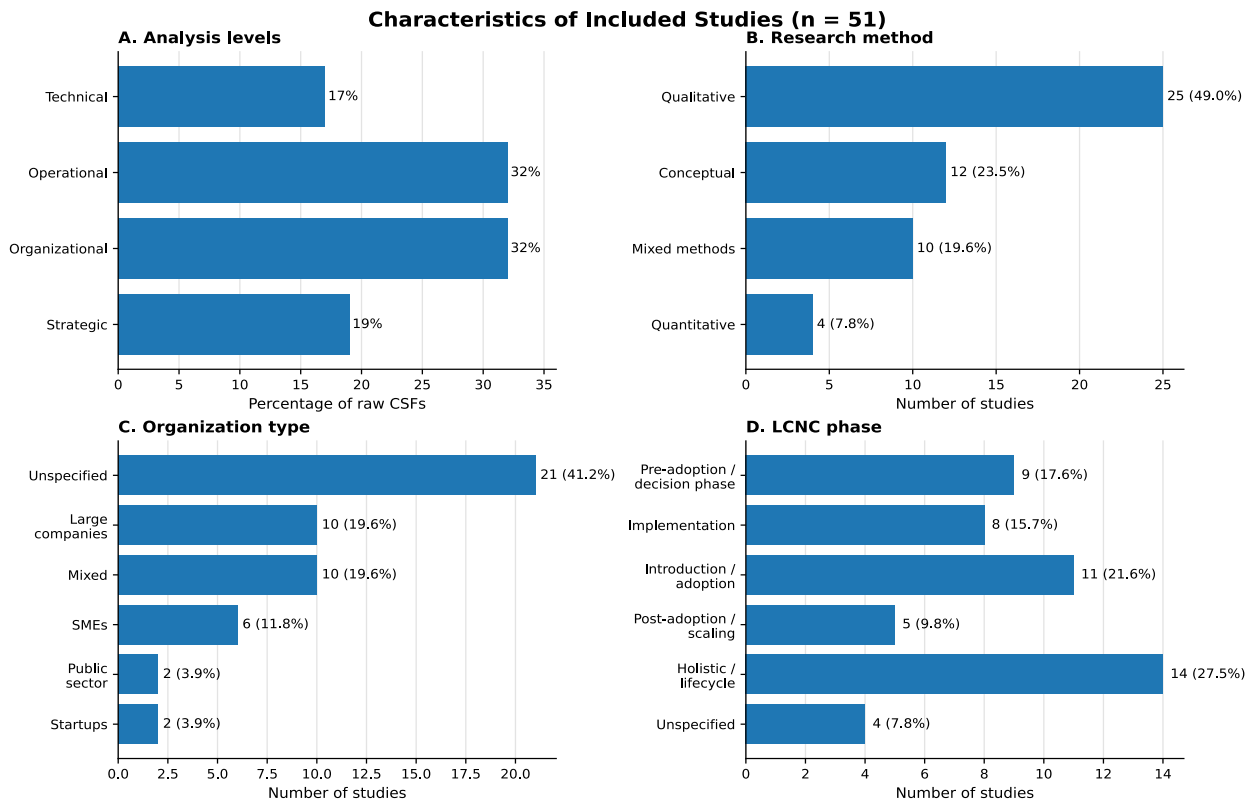


Fig. 2. Overview of the characteristics of the 51 included studies

Many studies do not specify a clear organizational context, and among the specified contexts, large companies and mixed settings are most frequently represented, followed by small and medium-sized enterprises (SMEs). In contrast, no study in the sample explicitly examines CSFs for LCNC adoption in public authorities (although two studies address LCNC in public-sector contexts, they do not focus on CSFs); thus, in general, the literature is clearly centered on private-sector settings. In terms of LCNC deployment phases, holistic or lifecycle-oriented perspectives form the largest category, followed by introduction/adoption, pre-adoption/decision, and implementation studies. Post-adoption and scaling are comparatively underrepresented.

B. Distribution of Raw CSFs and Cluster Frequencies

Table II provides an overview of the consolidated CSFs, their cluster assignments, and the number of underlying raw CSFs per final factor (as an indicator of empirical grounding). The CSFs differ in terms of their empirical support: those with a high number of raw CSFs, such as *Governance, Control, and Risk Management*, are supported by a broader literature base; those with a lower number of raw CSFs, such as *Roles, Responsibilities and Ownership*, should be understood as more exploratory success conditions whose empirical validation requires further research.

As shown in Table II, *Governance, Control, and Risk Management* is the most frequent cluster, with 31 raw CSFs. This is followed by *Human Factors, Competencies, and Enablement* and *Organizational Embedding and Scaling*, each with 22 raw CSFs, then by *Strategic Governance and Decision Logic*, with 21 raw CSFs. The clusters *Technical: Platform, Architecture, and Integration* (15), *Development Process, Methodologies, and Lifecycle* (14), and *Business-IT Collaboration* (13) arise less frequently. The least common cluster is *Roles, Responsibilities, and Ownership* (7).

Of the 145 identified raw CSFs, 106 (73%) were explicitly identified by the authors of the analyzed studies, while 39 (27%) were implicitly derived from the studies. Additionally, the results show a clear concentration in the organizational and operational dimensions: 47 factors fall into the operational category and 46 factors into the organizational category, while strategic factors (27) and technical factors (25) are seen less frequently. In other words, the literature primarily links LCNC success to organizational conditions, governance structures, and operational implementation practices.

Overall, governance-related and competency-oriented CSFs dominate the research field; this pattern is reflected in the thematic focus of the analyzed literature, which continues in the subsequent consolidation. The low representation of *Roles, Responsibilities, and Ownership* is nevertheless noteworthy, since role clarity is repeatedly emphasized in the literature as a prerequisite for other CSFs, such as governance and collaboration.

C. Detailed Description of the Most Frequent Clusters

This section provides a detailed description of all the identified clusters and their associated CSFs.

Cluster 1: Governance, Control & Risk Management

The CSFs of this cluster form a continuum ranging from institutional anchoring through risk-differentiated control mechanisms to the operational implementation of guidelines.

Institutionalized LCNC Governance Structure

‘Institutionalized LCNC Governance Structure’ is the most strongly grounded CSF in the analysis, and it is crucial to systematically balancing innovation and control. Successful LCNC implementations require both governance structures for risk mitigation [32] and the overcoming of cultural inhibitors, such as a lack of a development culture or resistance to change [33]. However, a purely centralized and restrictive governance model carries the risk of limiting innovation potential [34]. Instead, the studies recommend either a collaborative governance approach, which combines central control with the involvement of citizen and professional developers [35], or a hybrid approach, which combines top-down compliance with bottom-up user involvement [36]. Clear decision-making structures, portfolio management, and coordinated IT governance models promote transparency, risk minimization, and strategic coherence [8].

Operational Governance, Control, and Policy Mechanisms

‘Operational Governance, Control, and Policy Mechanisms’ ensures that LCNC initiatives do not lead to uncontrolled proliferation of applications or shadow IT but instead are managed in a structured, risk-appropriate manner. Formalized approval processes, such as use-case registrations and IT approvals, are required depending on the criticality of an application [37]. Similarly, the establishment of clear control structures is emphasized to avoid issues with integration and maintainability [38]. Therefore, role-based access rights and tiered permissions are necessary to enable innovation while simultaneously limiting risks [3].

Security, Compliance, and Regulatory Management

‘Security, Compliance, and Regulatory Management’ ensures that LCNC applications comply with legal, internal-corporate, and industry-specific requirements. Legal frameworks such as the General Data Protection Regulation (GDPR) must be taken into account as early as the platform selection phase [39]. The risks regarding data protection, access control, and the improper use of components are higher, as CDs often lack in-depth security knowledge [40]. LCNC platforms can contribute to risk minimization through platform-integrated governance functions, such as automated compliance checks and access-control mechanisms [36]. At the same time, robust security mechanisms, role-based access controls, and adherence to industry standards are identified as key requirements for LCNC platforms [41]. However, technical mechanisms alone are not sufficient: clear guidelines, supporting materials, and compliant user interfaces are necessary to support end users in developing and using applications in a manner that complies with the aforementioned requirements [42].

Risk-Based Management and Application Classification

Risk-based management of LCNC applications requires the systematic classification of applications by their potential impact on the organization, data, and decision-making processes. The literature recommends classifying LCNC applications according to risk, so that highly critical solutions—such as those affecting personnel, financial, or compliance decisions—are subject to explicit approval, while less critical applications may follow simplified control mechanisms while nevertheless complying with standardized accountability rules [36]. In the context of maturity-oriented governance models, an early classification of LCNC applications is recommended in order to systematically define responsibilities and quality standards before deployment into organizational practice [29]. An adaptive governance structure helps in establishing risk-appropriate control mechanisms and taking the organization's capacity to use LCNC applications into account.

Responsible AI and Quality Control

'Responsible AI and Quality Control' ensures that LCNC and AI-supported applications are operated reliably, transparently, and in a manner that is appropriate vis-à-vis the involved risks. Comprehensive testing and feedback loops are particularly necessary in business- or societally-critical contexts, as automated testing capabilities in LCNC platforms are sometimes limited and manual quality assurance can otherwise become a bottleneck [43]. With the use of generative AI, the need for systematic use-case categorization and risk-based assessment increases, particularly with regard to bias, quality of training data, and compliance requirements [35]. Human review remains essential, as AI-generated artifacts can be prone to errors, and so a close human-in-the-loop approach is required, particularly for complex use cases [35].

TABLE II. CONSOLIDATED CRITICAL SUCCESS FACTORS AND CLUSTER ASSIGNMENT

No.	Cluster	Cumulated raw CSFs	Final CSFs	# Raw CSFs
1	Governance, Control, and Risk Management	31	Institutionalized LCNC Governance Structure	10
			Operational Governance, Control, and Policy Mechanisms	9
			Security, Compliance, and Regulatory Management	6
			Risk-Based Management and Application Classification	3
			Responsible AI and Quality Control	3
2	Human Factors, Competencies, and Enablement	22	Systematic Qualification and Training Programs	7
			Knowledge Management and Structured Knowledge Transfer	7
			Motivation, Support, and User Acceptance	5
			Competency Mix and Availability of Qualified Expertise	5
3	Organizational Embedding and Scaling	22	Structured Adoption and Governance Model	8
			Institutionalization through Centers of Excellence and Central Structures	6
			Scalability and Performance Management	6
4	Strategic Governance and Decision-Logic	21	Systematic Use-Case and Technology Selection	10
			Top-Management Commitment and Resource Strategy	5
			Strategic Embedding of LCNC in Corporate and IT Strategy	4
			Strategic Portfolio Management and Maturity Governance	2
5	Technical: Platform, Architecture, and Integration	14	Technical Integration and Architectural Capability	8
			Technological Fit and Platform Robustness	6
6	Development Process, Methodologies, and Lifecycle	14	Platform-Aligned and Methodologically Embedded Development Processes	8
			Iterative, Feedback-Driven, and Standardized Development	6
7	Business-IT Collaboration	14	Structured Business-IT Collaboration	7
			Integration of Citizen and Professional Development	4
			Collaborative Organizational Culture	3
8	Roles, Responsibilities, and Ownership	7	Clear Roles and End-to-End Responsibilities	6
			Operational Management Involvement	1

Cluster 2: Human Factors, Competencies, and Enablement

This cluster links individual empowerment (qualification, competency mix) with organizational knowledge structures and motivational frameworks.

Systematic Qualification and Training Programs

‘Systematic Qualification and Training Programs’ for CDs are a central prerequisite for sustainable LCNC adoption. The maturity-model logic of [29] suggests that structured training concepts, defined learning paths, and continuous performance measurement are essential for moving from isolated initiatives to standardized and optimized capabilities. Since LCNC empowers non-technical users in particular, specific risks arise regarding AI ethics, data security, and compliance, all of which make mandatory training necessary [36]. In addition to formal training, communities and Centers of Excellence (CoE) promote knowledge transfer and support the development of competencies in practice [44].

Skill development and knowledge management are closely interrelated. Training programs build individual competencies, while structured knowledge transfer ensures their organization-wide availability and sustainability.

Knowledge Management and Structured Knowledge Transfer

LCNC initiatives often fail due not to the technology, but to insufficiently-documented and fragmented knowledge. The term “silent barrier” describes the difficulty of translating implicit domain knowledge into structured, automatable logic [45], and a lack of formalized knowledge-management mechanisms leads to inefficiencies, delays, and missteps. On this point, community-based learning formats, mentoring models, and dedicated knowledge databases promote the exchange of best practices and reduce redundant support requests. In addition, repositories containing documented solution cases, peer support, and structured training and exchange formats support sustainable competency development [3], [46]. Structured knowledge transfer mechanisms are particularly crucial in AI-supported LCNC approaches, to make expertise accessible and actionable across the organization [47].

Motivation, Support, and User Acceptance

The successful implementation of LCNC initiatives depends largely on the motivation and the acceptance of the involved employees. Beyond reduced technical barriers to entry, citizen development requires commitment and a willingness to learn, as CDs typically take on additional development responsibilities without a corresponding reduction in their workload [30]. To this end, in order to foster sustainable participation, organizations should adapt performance metrics, reward systems, and career paths that appropriately recognize the additional contributions of CDs [18]. Ongoing support, together with practical examples and training, is crucial for re-

ducing uncertainty, burnout, and technology-related anxieties, as well as for fostering positive attitudes toward LCNC [48].

Competency Mix and Availability of Qualified Expertise

While LC and AI platforms reduce technical barriers to entry, they do not replace the need for diverse types of expertise. Even with AI-supported LC development, the development of productive applications often requires a combination of technical know-how, AI-related know-how, and domain knowledge [49]. [50] show that especially in more complex use cases, both business expertise and IT competence are required to map processes in a manner that is functionally correct and technically viable. Cross-functional collaboration and realistic testing with intended users help in identifying the actual competency requirements [49]; nevertheless, professional IT expertise remains essential for maintenance, governance, and complex integration tasks, regardless of LCNC usage [50].

Cluster 3: Organizational Embedding and Scaling

This cluster describes the path from institutional embedding through structured adoption processes to sustainable scaling and performance measurement.

Structured Adoption and Governance Model

Structured adoption and a governance model are needed to strategically align LCNC initiatives and to scale them in a controlled manner. With regard to avoiding uncoordinated developments and the associated risks, [29] emphasize the need for a comprehensive maturity model that supports companies in systematically assessing their organizational capabilities, establishing adaptive governance, and deriving tailored adoption roadmaps. Key factors include early top-management sponsorship and targeted resource allocation [51]. At the same time, organization-wide advocacy efforts must be undertaken, and resistance from management, IT, and CDs must be addressed [38]. Case studies show that a phased rollout, co-design approaches, and iterative change management are crucial to sustainable acceptance [45]; additionally, guardrails, clearly defined roles, and business-led structures support a controlled yet flexible LCNC adoption [2].

Institutionalization through CoE and Central Structures

The institutionalization of LCNC through central structures serves to strategically anchor, coordinate, and control company-wide initiatives. The absence of central steering risks the uncoordinated development of isolated individual solutions, which is why a structured plan and standardized criteria for use-case selection are necessary [38]. A CoE or “Citizen Development Hub” serves as a central point of contact, formulates a vision, defines guidelines, offers training, and provides a technical knowledge base [2]. To this end, a dedicated LCNC unit pools expertise, clarifies responsibilities, and supports CDs through consulting and training [28]. The specific structure can be centralized, decentralized, or hybrid, but it

should include clearly defined roles, decision-making processes, and governance structures [29].

Scaling and Performance Management

‘Scaling and Performance Management’ encompasses the systematic management of the growth, performance, and cost-effectiveness of LCNC initiatives. Building on structured adoption processes, [32] propose and evaluate a maturity model with clearly defined target states, roadmaps, and platform-specific KPIs so as to continuously steer implementation and usage. As shown in [40], a lack of scalability can be attributed in particular to application complexity, the ability to support multiple developers, and increasing user and data volumes. As scaling increases, so do the demands on maintainability, performance, cost control, and technical robustness [41], [46]; therefore, transitions from experimental use to productive mass operation require professional structures, continuous evaluation, and (where necessary) greater integration of traditional IT architectures [15] [52].

Cluster 4 Strategic Governance and Decision-Logic

This cluster is concerned with how organizations strategically legitimize LCNC initiatives, prioritize them, and secure the necessary resources for their implementation.

Strategic Embedding of LCNC in Corporate and IT Strategy

The adoption of LCNC platforms requires clear strategic alignment between IT and business units; [8] point out that without such alignment, LCNC is often confined to the IT department or is perceived by business units as failing to deliver value. [32] describe a structured adoption process that comprises the assessment of existing capabilities, the definition of strategic goals, the derivation of concrete measures, and a management guided by governance and KPIs. [53] emphasize that a lack of strategic integration can lead to risks such as security issues, technical debt, or limited value contribution. Furthermore, successful implementation (particularly in SMEs) requires organizational clarity and a strategic process culture [54].

Systematic Use-Case and Technology Selection

The literature consistently indicates that LCNC should be employed only when it is justified by requirements, complexity, and integration needs. Drawing on expert interviews, [44] emphasize the importance of a structured assessment of use cases in terms of feasibility, governance requirements, user base, data criticality, and system integration. Technical criteria such as adaptability, integration capabilities, budget, and flexibility are also relevant to the decision-making process [11]. An insufficiently grounded decision to adopt LCNC entails the risk of a costly future migration to high-code, with the quality of the decision depending heavily on the experience of the experts involved [28], [55].

Strategic Portfolio Management and Maturity Governance

This CSF highlights the need to treat LCNC initiatives not as isolated or standalone projects, but as a manageable portfolio comprising platforms, use cases, and applications. This is particularly relevant to organizations deploying multiple LCNC platforms in parallel. Strategic portfolio management helps avoid redundancies, prioritize applications, assess risks, and align the LCNC portfolio with corporate and IT objectives. Maturity management complements this by assessing the current stage of development vis-à-vis a clear LCNC vision, defining target states, and deriving an implementation roadmap [29]. LCNC-specific KPIs, integration into enterprise architecture management, and regular audits support transparency, standardization, and risk mitigation within the portfolio [29]. Furthermore, a preceding assessment of risks, benefits, and the organization’s initial conditions provides the basis for selecting an appropriate LCNC implementation strategy [15].

Top-Management Commitment and Resource Strategy

Top management’s commitment and a clear resource strategy are key prerequisites for the sustainable adoption of LCNC. Studies show that strong support from top management significantly contributes to organization-wide adoption by initiating, communicating, and structurally embedding LCNC programs. Without such support, initiatives tend to be disjointed or emerge in an uncontrolled, bottom-up manner [8]. In addition to structural support, expected efficiency gains are a major driver of adoption decisions [51]. Successful LCNC initiatives require targeted resource allocation for, e.g., training, governance, and technical support. Particularly in SMEs, top management must recognize that CDs cannot operate entirely independently, and they may require dedicated support and additional capacity from the IT department when technical or development-related bottlenecks arise [45]. In the absence of strategic integration, management sponsorship, or structured resource planning, the potential of LCNC generally remains untapped.

Cluster 5: Technical: Platform, Architecture, and Integration

This cluster concerns the technical aspects of LCNC adoption, namely the platform’s suitability for the specific use case and the organization’s capacity to integrate it into existing IT architecture.

Technological Fit and Platform Robustness

‘Technological Fit and Platform Robustness’ describes the systematic assessment of whether an LCNC platform can meet the technical requirements of a specific use case over the long term. Prior to implementation, an organization must determine whether performance, infrastructure, and integration requirements (such as those for IoT or AI applications) can be met by LCNC alone or whether hybrid approaches are necessary [28], [49]. Practical examples show that robust architectures often combine LC with traditional code [56]. In addition to performance requirements, such risks as vendor lock-in, dependencies on third-party providers, infrastructure stability, and unexpected cost developments

must be actively managed [46]. Without strategic alignment, governance, and data-management mechanisms, quality issues and technical debt are likely to arise [53].

Technical Integration and Architectural Capability

‘Technical Integration and Architectural Capability’ refers to an organization’s ability to integrate LCNC platforms in a compatible form into existing IT landscapes. [40] identify the integration of LCNC solutions into existing systems as a potential barrier to adoption, particularly when no predefined integration components are available or when extensive legacy systems are involved. Consequently, studies emphasize the importance of systematically embedding LCNC technologies into enterprise architecture management to ensure transparency, reusability, and strategic alignment [29]. In addition to the initial integration, sustainable operation requires organizational capabilities for the deployment, management, and continuous development of the platforms, connectors, and data structures [29]. Thus, structured integration into existing architectures is a key prerequisite for making LCNC applications both scalable in the long term and usable across the entire organization.

Cluster 6: Development Process, Methodologies, and Lifecycle

The two CSFs in this cluster complement each other: the first concerns the methodological integration of LCNC into existing development processes, while the second focuses on the concrete design of iterative, feedback-driven development cycles.

Platform-Aligned and Methodologically Embedded Development Processes

Standardized, methodologically-embedded development processes are necessary to implement LCNC initiatives in a structured, quality-assured, and sustainable manner. This includes not only the delineation of internal development procedures but also the deliberate use of predefined platform processes. Companies must analyze and structure their processes and, if necessary, adapt them to the platform’s functional capabilities, so that organizational workflows are aligned with the standardized process logics and constraints of LCNC platforms [49]. [43] show that LCNC platforms can contribute to supporting agile software development as well as provide functions for key software engineering practices, including automated testing and documentation, which should be embedded into development routines. A realistic definition of requirements and an assessment of project scope and complexity are necessary to select suitable platform-supported development approaches [3]. Empirical findings also show that LCNC should be embedded in modular, iterative business process management models, rather than following strictly sequential stage-based models [54].

Iterative, Feedback-Driven, and Standardized Development

This CSF denotes the importance of iterative development cycles and continuous user feedback loops, along with usage of standardized components in the LCNC context. Studies

show that CDs develop applications in short iterations, conduct regular usability tests, and refine solutions together with stakeholders to ensure suitability and acceptance [44]. Particularly in the startup environment, LCNC platforms enable the rapid validation of a minimum viable product and continuous adaptation based on real user feedback [46]. Sustainable scaling also requires standardized, modular development approaches as well as central repositories for reusing components [57]. Case studies show that iterative prototyping not only supports requirement clarification but also can be translated into production-ready solutions [56]. Drawing on a phase model, [39] show that a structured evaluation and goal-definition phase lays the foundation for the subsequent iterative development and transition of prototypes into production-ready solutions.

Cluster 7: Business–IT Collaboration

This cluster considers various levels of collaboration, ranging from structured interaction between business units and IT, through the integration of professional and non-professional developers, to the cultural foundation of this cooperation.

Structured Business–IT Collaboration

‘Structured Business–IT Collaboration’ describes early-stage collaboration between business units and IT throughout the entire LCNC lifecycle. Because LCNC transforms the traditional development process, its adoption should be shaped jointly by business and IT, with the goal of avoiding resistance, role ambiguities, and conflicts of authority [51]. A hybrid collaboration model between CDs and IT teams is highlighted as a key success factor [58]. Moreover, the early and ongoing involvement of business units in defining requirements and in decision-making processes strengthens adaptability and acceptance [42]. Finally, visual process models and iterative coordination promote transparency and efficiency [43]. In this context, IT should primarily serve a supportive and empowering role, rather than acting primarily as a controlling force [15].

Integration of Citizen and Professional Development

The literature describes the integration of citizen and professional development as a crucial factor in combining the benefits of rapid LCNC development with technical quality assurance. Critical challenges in areas such as testing, performance, and data security require the involvement of professional IT expertise [50]; this is particularly true for integration, rollout, and sustainable professionalization in resource-constrained environments [45]. Software engineering expertise is also essential for architectural decisions and tool selection [59]. Furthermore, the use of generative AI increases the risk of technical-quality deficits if expert human oversight is not sufficiently integrated [35].

Collaborative Organizational Culture

A collaborative organizational culture forms the social foundation for successful LCNC adoption. LCNC can act as a mediating element and break down existing communication barriers; however, this requires a cultural shift, as the responsibility for development is increasingly shifted to line

departments [50]. [51] emphasize that technical and social aspects must be optimized together. An open culture featuring so-called LCNC champions facilitates knowledge transfer, lowers barriers to entry, and strengthens organization-wide acceptance of LC initiatives [51].

Cluster 8: Roles, Responsibilities, and Ownership

The CSFs in this cluster deal with the fundamental question of who in an organization is responsible for LCNC applications, from development to operation.

Clear Roles and End-to-End Responsibilities

[37] emphasize that clearly defining roles and responsibilities is crucial for avoiding such risks as shadow IT and quality issues. While platform responsibility often rests with central IT or a CoE, accountability for maintenance, support, and data classification varies across IT and CDs [37]. The “you build it, you run it” principle illustrates end-to-end ownership across the entire lifecycle [44]. A nuanced categorization of roles—e.g., utilizing LC champions as intermediaries between business and IT—supports structured governance and the sustainable allocation of responsibilities [3].

Operational Management Involvement

While clear roles and end-to-end responsibilities establish formal accountability, the active involvement of operational management ensures that LCNC initiatives are aligned with actual work practices. Operational managers act as a link between CDs, frontline staff, and IT by maintaining continuous dialogue on the requirements, usage contexts, emerging issues, and potential workarounds. This feedback enables early identification of problems and helps ensure that the developed applications are fit for purpose throughout their lifecycle without the creation of unintended risks or inefficiencies [52].

V. DISCUSSION

This discussion section examines the results of the SLR through the theoretical lenses adopted in Section II, and it further develops the CSF perspective for the LCNC context. The findings reveal that organizational and socio-technical success factors dominate the literature, indicating that LCNC adoption is not merely a technical implementation issue. Governance appears to be a central hybrid control mechanism, while roles and competence requirements are reshaped through the increasing importance of CDs and new business–IT constellations. Taken together, the results characterize LCNC adoption as a socio-technical transformation. This section concludes by outlining the theoretical and practical implications of the results.

A. Dominance of Organizational and Socio-technical Success Factors

Although LCNC platforms provide a technological foundation, only 17% of the identified CSFs relate to purely technical aspects. Instead, the highest concentration is found in

the *Governance, Control, and Risk Management* cluster, followed by the clusters *Human Factors, Competencies, and Empowerment* and *Organizational Embedding and Scaling*. In terms of the four dimensions of analysis (strategic, organizational, operational, and technical), organizational and operational factors clearly dominate.

This pattern indicates that LCNC adoption is shaped less by technical platform properties in themselves than by the organizational conditions in which these platforms are introduced and governed. While LCNC platforms lower technical barriers to entry, they simultaneously create new requirements regarding roles, competencies, coordination, control, and responsiveness to regulatory and sector-specific requirements. The findings thus support the relevance of the TOE perspective, particularly the importance of organizational and environmental conditions to LCNC adoption [7].

In addition, the present review confirms that public-sector contexts are comparatively underrepresented in the LCNC literature, with only two publications in the sample explicitly focusing on this setting. According to the TOE perspective, this gap is noteworthy, as differences in organizational and environmental conditions may limit the transferability of findings that are primarily derived from corporate settings.

B. Governance as a Central Hybrid Control Mechanism

A key finding is the prominent role of governance structures. LCNC expands the scope for decentralized development within business units but also creates such risks as shadow IT, security vulnerabilities, and unclear responsibilities. Therefore, governance functions not only as a control mechanism but also as an enabler that combines decentralized innovation with security, quality, and compliance.

Successful governance models are apparently adaptive and hybrid in nature, combining central guidelines and risk-based classification with decentralized development opportunities. Rather than following a one-size-fits-all logic, such governance mechanisms must be designed to be context-sensitive in order to steer LCNC platforms in a way that balances autonomy and control. The key challenge is to minimize risks without unnecessarily constraining innovation opportunities and local initiative [29], [37].

C. Transformation of Roles and Competency Requirements

LCNC brings about lasting changes to organizational role and competency structures. New profiles emerge, including CDs, LC champions, and hybrid business–IT interface roles. Citizen development does not replace professional IT expertise but complements it in a hybrid model, wherein business units increasingly take on automation and prototyping processes while IT remains responsible for architecture, operations, governance, and quality assurance [50].

Accordingly, training programs, knowledge management, and structured knowledge transfer emerge as central factors of success. LCNC requires not only technical platform skills but also interpersonal, domain-specific, and governance-related competencies [15]. At the same time, organizations need

mechanisms that make such knowledge available beyond individual users—e.g., through documentation, peer support, mentoring, and reusable solution patterns [3]. These mechanisms is particularly important because LCNC lowers barriers to development but does not eliminate the need for abstraction, coordination, and responsible use. These observations are consistent with prior work on CD competencies, which also emphasizes the continued relevance of technical, interpersonal, and domain-specific capabilities [60].

D. LCNC Adoption as a Socio-technical Transformation

The eight CSF clusters should not be understood as isolated domains, but rather as closely interrelated dimensions of LCNC adoption, whereby governance, qualifications, organizational embedding, and technical design mutually shape each other. Strategic steering influences how LCNC is governed and scaled; qualification enables the assumption of new roles; and technical decisions affect the scope, risks, and sustainability of operational use. Therefore, successful LCNC adoption depends on the alignment of organizational, technical, and human factors across multiple levels.

This interdependence underpins a socio-technical interpretation of LCNC adoption [61]. The identified clusters capture not only technological and organizational conditions but also the coordination mechanisms, role configurations, and operational practices through which LCNC is introduced, governed, and sustained. Because LCNC distributes development activities beyond IT to business units, governance and business–IT collaboration become important as the mediating mechanisms between central steering and decentralized flexibility. Consequently, LCNC should be understood not merely as a technology implementation, but as an organization-wide socio-technical transformation.

This interpretation is also reflected in the phase distribution of the reviewed studies: cross-lifecycle and holistic perspectives form the largest single category, and post-adoption and scaling phases appear less frequently as standalone research foci than the decision, adoption, and implementation phases (see Fig. 2). This suggests that the literature increasingly recognizes the breadth of the LCNC lifecycle but nevertheless devotes less explicit attention to the long-term organizational stabilization and scaling of LCNC initiatives.

E. Theoretical and Practical Implications

Organizations should place particular emphasis on organizational preparation, which includes establishing hybrid governance models, clearly defining roles and responsibilities, and implementing systematic qualification and upskilling programs. When accompanied by appropriate governance and capability development, LCNC can unlock considerable potential. Well-designed LCNC governance can help organizations more effectively balance innovation potential and risk exposure, enabling both flexibility and organizational control.

Moreover, the findings confirm the context-dependent nature of the CSF approach and underscore the need for dynamic, socio-technical models for LCNC adoption [7]. At the same time, the identified CSFs should be understood as an

analytically useful but still-provisional framework that requires further empirical validation across contexts, sectors, and adoption stages. Further empirical and quantitative studies—particularly on long-term scaling and on public-sector-specific challenges—are needed to test the validity and generalizability of the factors.

VI. CONCLUSIONS AND LIMITATIONS

In a systematic literature review, 51 academic publications were analyzed, from which 145 potential CSFs (called ‘raw’ CSFs in our analysis) were identified; these were then consolidated into 25 final CSFs, grouped into eight thematic clusters.

The identified CSFs show that successful LCNC adoption does not primarily depend on the choice of technical platforms but on the interplay of organizational, strategic, and operational conditions. Three main findings emerge from the analysis. First, governance-related success factors dominate the analysis: the *Governance, Control, and Risk Management* cluster, with 31 raw CSFs, is the most substantial. This illustrates that decentralized development freedom requires systematic control mechanisms. Second, LCNC adoption requires a renegotiation of roles and responsibilities between IT and business units, whereby citizen development should be understood not as a replacement of but a complement to professional IT expertise. Third, our analysis shows that the identified success factors do not operate in isolation but are linked by systematic interdependencies; governance requires clarity of roles, qualification necessitates institutional sponsorship, and scaling is only successful with sufficient strategic embedding. Therefore, LCNC platforms should be classified not as isolated development tools but as an organization-wide transformation phenomenon.

Nevertheless, the results must be interpreted in the context of methodological and content-related limitations. For instance, the sample is significantly influenced by the AISeL database: of the 51 publications ultimately included, 33 come from this database, resulting in a particularly strong representation of IS-oriented research discourse, whereas perspectives rooted in engineering may be underrepresented. With regard to the literature base, an additional limitation arises from the selection criteria applied; specifically, only peer-reviewed articles and conference papers were included, while gray literature and unpublished sources were excluded. However, these sources might contain relevant practical insights and complementary perspectives. Furthermore, the analyzed sample is dominated by qualitative and conceptual studies, with limited representation of quantitative studies.

In addition, the consolidation of the CSFs required interpretive decisions. We consolidated the 145 raw CSFs into 25 final CSFs using the criteria of content-related similarity and semantic similarity between factors; moreover, the assignment of these factors to higher-level clusters also involved subjective judgment. In addition, the extracted CSFs are based on the observations, assessments, and interpretations reported in the literature, and whether these factors actually

causally determine implementation success cannot be verified by the present SLR. This limitation not only applies to the present study but also points to a fundamental methodological constraint in both CSF research and SLRs.

Finally, the transferability of the results to different organizational contexts is limited. A large portion of the analyzed articles does not provide a clear statement regarding the specific type of organization under study. At the same time, studies on large companies dominate vis-à-vis studies on SMEs, the public sector, or startups. In this respect, future research should quantitatively confirm whether the identified factors can in fact be understood as success factors across different organizational settings.

REFERENCES

- [1] G. Vial, "Understanding digital transformation: A review and a research agenda," *Journal of Strategic Information Systems*, vol. 28, no. 2, pp. 118–144, 2019, doi: 10.1016/j.jsis.2019.01.003.
- [2] B. Binzer, E. Elshan, D. Fürstenau, and T. J. Winkler, "Establishing a Low-Code/No-Code-Enabled Citizen Development Strategy," *MIS Quarterly Executive*, vol. 23, no. 3, pp. 253–273, 2024, doi: 10.17705/2msqe.0009.
- [3] A. Viljoen, M. Radić, A. Hein, J. Nguyen, and H. Kremer, "Governing Citizen Development to Address Low-Code Platform Challenges," *MIS Quarterly Executive*, vol. 23, no. 3, pp. 305–324, 2024, doi: 10.17705/2msqe.00100.
- [4] K. Rokis and M. Kirikova, "Exploring Low-Code Development: A Comprehensive Literature Review," *Complex Systems Informatics and Modeling Quarterly*, vol. 36, pp. 68–86, 2023, doi: 10.7250/csimq.2023-36.04.
- [5] KPMG International, "Low-code adoption as a driver of digital transformation," KPMG International, 2024. [Online]. Available: <https://assets.kpmg.com/content/dam/kpmg/pt/pdf/pt-low-code-adoption-driver-digital-transformation.pdf>
- [6] J. Bratinčević, R. Taylor, and Z. Stone, "The Low-Code Market Could Approach \$50 Billion By 2028," Forrester. Accessed: Jan. 20, 2026. [Online]. Available: <https://www.forrester.com/blogs/the-low-code-market-could-approach-50-billion-by-2028/>
- [7] E. L. Trist, The evolution of socio-technical systems: a conceptual framework and an action research program. Toronto: Ontario Ministry of Labour, Ontario Quality of Working Life Centre, 1981.
- [8] D. Hoogsteen and H. Borgman, "Empower the Workforce, Empower the Company? Citizen Development Adoption," in *Proceedings of the 55th Hawaii International Conference on System Sciences (HICSS 2022)*, 2022.
- [9] C. Leyh, A. Lorenz, M. J. Faruga, and L. Koller, "ERP System Implementation Projects: An Update on Critical Success Factors," in *Information Technology for Management: Intelligent Alignment of IT with Business and Society*, Lecture Notes in Business Information Processing (LNBIP), vol. 555, E. Ziemba, W. Chmielarz, and J. Wątróbski, Eds., Springer Nature Switzerland, 2026, pp. 38–57, doi: 10.1007/978-3-031-93580-0_2.
- [10] A. Viljoen et al., "Navigating Flexibility and Standardisation in Low-Code/No-Code Development," *Information Systems Journal*, vol. 36, no. 1, pp. 95–109, 2025, doi: 10.1111/isj.70001.
- [11] Y. Luo, P. Liang, C. Wang, M. Shahin, and J. Zhan, "Characteristics and Challenges of Low-Code Development: The Practitioners' Perspective," in *Proceedings of the 15th ACM / IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM '21)*, 2021.
- [12] M. A. Al Alamin, S. Malakar, G. Uddin, S. Afroz, T. B. Haider, and A. Iqbal, "An Empirical Study of Developer Discussions on Low-Code Software Development Challenges," in *Proceedings of 18th IEEE/ACM International Conference on Mining Software Repositories (MSR)*, 2021, doi: 10.1109/MSR52588.2021.00018.
- [13] R. Waszkowski, "Low-code platform for automating business processes in manufacturing," IFAC-PapersOnLine, vol. 52, no. 10, pp. 376–381, 2019, doi: 10.1016/j.ifacol.2019.10.060.
- [14] A. C. Bock and U. Frank, "In Search of the Essence of Low-Code: An Exploratory Study of Seven Development Platforms," in *Proceedings of the ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)*, 2021, doi: 10.1109/MODELS-C53483.2021.00016.
- [15] O. Biedova, B. Ives, D. Male, and M. Moore, "Strategies for Managing Citizen Developers and No-Code Tools," *MIS Quarterly Executive*, vol. 23, no. 2, pp. 165–183, 2024, doi: 10.17705/2msqe.00093.
- [16] M. M. Li, C. Peters, M. Poser, K. Eilers, and E. Elshan, "ICT-enabled job crafting: How Business Unit Developers use Low-code Development Platforms to craft jobs," in *ICIS 2022 Proceedings*, 2022.
- [17] T. Davenport, "MISQE Insight: On the Inevitability of Citizen Development," *MIS Quarterly Executive*, vol. 22, no. 4, pp. vii–xi, 2023.
- [18] E. Elshan, B. Binzer, and T. J. Winkler, "From Software Users to Software Creators: An Exploration of the Core Characteristics of the Citizen Developer Role and the Related Re- and Upskilling Programs," *Business & Information Systems Engineering*, vol. 67, no. 1, pp. 31–53, 2025, doi: 10.1007/s12599-024-00915-1.
- [19] M. O. Ajimati, N. Carroll, and M. Maher, "Adoption of low-code and no-code development: A systematic literature review and future research agenda," *Journal of Systems and Software*, vol. 222, pp. 1–25, 2025, doi: 10.1016/j.jss.2024.112300.
- [20] A. Bucaioni, A. Cicchetti, and F. Ciccozzi, "Modelling in low-code development: a multi-vocal systematic review," *Softw Syst Model*, vol. 21, no. 5, pp. 1959–1981, Oct. 2022, doi: 10.1007/s10270-021-00964-0.
- [21] E. W. T. Ngai, C. C. H. Law, and F. K. T. Wat, "Examining the critical success factors in the adoption of enterprise resource planning," *Computers in Industry*, vol. 59, no. 6, pp. 548–564, Aug. 2008, doi: 10.1016/j.compind.2007.12.001.
- [22] C. V. Bullen and J. F. Rockart, "A primer on critical success factors," Center for Information Systems Research, MIT Sloan School of Management, 1981.
- [23] J. F. Rockart, "Chief executives define their own data needs," *Harvard Business Review*, vol. 57, no. 2, pp. 81–93, 1979.
- [24] A. C. Boynton and R. W. Zmud, "An Assessment of Critical Success Factors," *Sloan Management Review*, vol. 25, no. 4, pp. 17–27, 1984.
- [25] A. Nicolai and A. Kieser, "Trotz eklatanter Erfolglosigkeit: Die Erfolgsfaktorenforschung weiter auf Erfolgskurs," *Die Betriebswirtschaft : DBW*, vol. 62, no. 6, pp. 579–596, 2002.
- [26] U. Frank, P. Maier, and A. Bock, "Low Code Platforms: Promises, Concepts and Prospects: A Comparative Study of Ten Systems," ICB Research Reports, 2021, doi: 10.17185/DUEPUBLICO/75244.
- [27] J. W. Ross, P. Weill, and D. Robertson, *Enterprise Architecture As Strategy: Creating a Foundation for Business Execution*. Harvard Business Review Press, 2006.
- [28] A. Novales and R. Mancha, "Fueling Digital Transformation with Citizen Developers and Low-Code Development," *MIS Quarterly*

- Executive*, vol. 22, no. 3, pp. 221–234, 2023, doi: 10.17705/2msqe.00083.
- [29] N. Prinz and S. Zimmermann, “Towards a Maturity Model for Adopting Low-Code Development Platforms,” in *PACIS 2024 Proceedings*, 2024.
- [30] A. Viljoen, J. Nguyen, M. Kauschinger, and A. Hein, “Fostering Scalable Citizen Development in Organizations: Towards a Guiding Framework,” in *AMCIS 2023 Proceedings*, 2023.
- [31] J. Webster and R. T. Watson, “Analyzing the Past to Prepare for the Future: Writing a Literature Review,” *MIS Quarterly*, vol. 26, no. 2, pp. xiii–xxiii, 2002.
- [32] N. Prinz, C. Felden, M. Huber, C. Rentrop, C. Riedinger, and S. Zimmermann, “A Maturity Model for Managing Low-Code Development Platform Adoption,” in *PACIS 2025 Proceedings*, 2025.
- [33] S. Käss, S. Strahringer, and M. Westner, “Practitioners’ Perceptions on the Adoption of Low Code Development Platforms,” *IEEE Access*, vol. 11, pp. 29009–29034, 2023, doi: 10.1109/ACCESS.2023.3258539.
- [34] D. Fürstenau, M. Sandner, H. Rothe, and D. Anapliotis, “Shadow IT, risk, and shifting power relations in organizations,” in *AMCIS 2016 Proceedings*, 2016.
- [35] O. Bruhin, E. Dickhaut, E. Elshan, and M. M. Li, “The Rise of Generative AI in Low Code Development Platforms – An Analysis and Future Directions,” in *Proceedings of the 57th Hawaii International Conference on System Sciences*, 2024, doi: 10.24251/HICSS.2024.932.
- [36] P. Somer, “Algorithmic Accountability of Low-Code/No-Code Artificial Intelligence: A Literature Review,” in *AMCIS 2025 Proceedings*, 2025.
- [37] L. Stegmann, R. Berendorf, S. K. Zimmermann, and A. Weeger, “Balancing Risks and Opportunities in the Governance of Low Code Development Platforms,” in *ECIS 2024 Proceedings*, 2024.
- [38] E. Elshan, E. Dickhaut, and P. A. Ebel, “An Investigation of Why Low Code Platforms Provide Answers and New Challenges,” in *Proceedings of the 56th Hawaii International Conference on System Sciences*, 2023, doi: 10.24251/HICSS.2023.746.
- [39] F. Z. Cai, S. Y. Huang, T. S. Kessler, and F. J. Fottner, “A Case Study: Digitalization of Business Processes of SMEs with Low-Code Method,” *IFAC-PapersOnLine*, vol. 55, no. 10, pp. 1840–1845, 2022, doi: 10.1016/j.ifacol.2022.09.666.
- [40] S. Käss, S. Strahringer, and M. Westner, “Drivers and Inhibitors of Low Code Development Platform Adoption,” in *Proceedings of the 24th IEEE Conference on Business Informatics (CBI)*, 2022, doi: 10.1109/CBI54897.2022.00028.
- [41] A. Abendroth and B. Bender, “Bridging the Gap: Low-Code Platforms and the Future of ERP Customization,” in *AMCIS 2025 Proceedings*, 2025.
- [42] M. Heuer, C. Kurtz, and T. Böhm, “Towards a Governance of Low-Code Development Platforms Using the Example of Microsoft PowerPlatform in a Multinational Company,” in *Proceedings of the 55th Hawaii International Conference on System Sciences*, 2022, doi: 10.24251/HICSS.2022.831.
- [43] S. A. A. Naqvi, M. Zimmer, and P. Drews, “Empowering the Public Sector: Unveiling the Transformational Potential of Low-Code in Crisis Responses,” in *AMCIS 2024 Proceedings*, 2024.
- [44] J. M. Kraus and T. Hess, “IT for the People: Driving Digital Innovation with Citizen Development,” in *AMCIS 2025 Proceedings*, 2025.
- [45] M. M. Li, S. Hönigsberg, H. Wache, and B. Gläsel, “Digital Transformation in Small Businesses: Challenges of Citizen Developers and Low-Code Platforms,” in *AMCIS 2025 Proceedings*, 2025.
- [46] E. Laevskaya, S. A. A. Naqvi, and P. Drews, “No-Code Platforms in Startups: Explaining Decisions for Adoption and Abandonment,” in *AMCIS 2025 Proceedings*, 2025.
- [47] B. M. Shah, R. Bansal, and A. Ramaswamy, “Democratizing AI: Current Landscape and Framework Development for No-Code AI Adoption,” in *Proceedings of the International Conference on Artificial Intelligence and Digital Ethics (ICAIDE)*, 2025, doi: 10.1109/ICAIDE65466.2025.11189515.
- [48] S. McHugh, N. Carroll, and C. Connolly, “Low-Code and No-Code in Secondary Education—Empowering Teachers to Embed Citizen Development in Schools,” *Computers in the Schools*, vol. 41, no. 4, pp. 399–424, 2024, doi: 10.1080/07380569.2023.2256729.
- [49] M. Kandaurova, D. A. Skog, and P. M. Bosch-Sijtsema, “The Promise and Perils of Low-Code AI Platforms,” *MIS Quarterly Executive*, vol. 23, no. 3, pp. 275–289, 2024, doi: 10.17705/2msqe.00098.
- [50] L. Bies, M. Weber, T. Greff, and D. Werth, “A Mixed-Methods Study of Low-Code Development Platforms: Drivers of Digital Innovation in SMEs,” in *Proceedings of the International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)*, 2022, doi: 10.1109/ICECCME55909.2022.9987920.
- [51] S. Käss, S. Strahringer, and M. Westner, “A Multiple Mini Case Study on the Adoption of Low Code Development Platforms in Work Systems,” *IEEE Access*, vol. 11, pp. 118762–118786, 2023, doi: 10.1109/ACCESS.2023.3325092.
- [52] R. M. Davison, L. H. M. Wong, and S. Alter, “Combining Low-Code/No-Code with Noncompliant Workarounds to Overcome a Corporate System’s Limitations,” *MIS Quarterly Executive*, vol. 23, no. 3, pp. 291–304, 2024, doi: 10.17705/2msqe.00099.
- [53] N. Carroll, J. Holmström, and S. Matook, “Special Issue Editorial: Transforming Business with Low-Code and No-Code,” *MIS Quarterly Executive*, vol. 23, no. 3, pp. v–xiii, 2024.
- [54] P. Arno, “From Theory to Practice: Empirical Insights on BPM Lifecycle Adoption and Low-Code Utilization in SMEs,” *IEEE Access*, vol. 13, pp. 117385–117403, 2025, doi: 10.1109/ACCESS.2025.3585261.
- [55] A. Schaetzle, S. Jourdan, T. Sturm, and P. Buxmann, “Low-Code vs. High-Code: Decision Criteria for Developing Business Applications,” in *AMCIS 2025 Proceedings*, 2025.
- [56] O. Biedova, B. Ives, and I. Junglas, “Gnosis Freight: Harnessing Data and Low-Code to Shipping Container Visibility and Logistics,” *Communications of the Association for Information Systems*, vol. 52, pp. 538–551, 2023, doi: 10.17705/1CAIS.05222.
- [57] O. Bruhin, P. Ebel, and E. Elshan, “Exploring the Paradoxical Relationship between Citizen and Professional Developers in Low Code Environments,” in *ICIS 2024 Proceedings*, 2024.
- [58] M. Stoica and A. I. Nitu, “Agile Software Development in the Cloud Using Citizen Development,” *Informatica Economica*, vol. 29, no. 1, pp. 16–28, 2025, doi: 10.24818/issn14531305/29.1.2025.02.
- [59] A. P. Velásquez, C. Pérez-Salazar, L. A. Hernández-González, and Á. J. Sánchez-García, “Systematic Literature Review of Low-Code and Its Future Trends,” in *Proceedings of the 12th International Conference in Software Engineering Research and Innovation (CONISOFT)*, 2024, pp. 30–37, doi: 10.1109/CONISOFT63288.2024.00015.
- [60] P. Rozbitski, C. Leyh, and C.-H. Ip, “Addressing Skills Shortages through Low-Code/No-Code: Skill Profiles and Implementation Challenges,” in *Proceedings of the 20th Conference on Computer Science and Intelligence Systems (FedCSIS)*, 2025, doi: 10.15439/2025F3612.
- [61] P. Rozbitski, C. Leyh, and C.-H. Ip, “Toward a multidimensional view of Low-Code/No-Code competencies: An institutional and socio-technical perspective,” in *Information technology for management: AI-centered innovation from theory to practice*, Lecture notes in business information processing (LNBIP), vol. 589, E. Ziemba, W. Chmielarz, and J. Wątróbski, Eds., Cham: Springer, to be published 2026.