Unleash the Power of Citizen Development: Leveraging Organizational Capabilities for Successful Low-Code Development Platform Adoption

Niculin Prinz
HTWG Konstanz
niculin.prinz@htwg-konstanz.de

Melanie Huber
BITCO³ GmbH
melanie.huber@bitco3.com

Justus Leonhardt
BITCO³ GmbH
justus.leonhardt@bitco3.com

Constanze Riedinger
HTWG Konstanz
constanze.riedinger@htwg-konstanz.de

Abstract

Given the increasing demand for application development and process automation, Low-Code Development Platforms (LCDPs) have become highly relevant in recent years. However, the lack of familiarity with the implementation and application of LCDPs in organizations poses a challenge. This publication therefore aims to shed light on the essential organizational capabilities that companies must master to overcome this obstacle. Using action design research, this study develops a model-based framework of 21 organizational capabilities for successful LCDP adoption. It underscores the importance of conceptual development as a prerequisite for effective management and long-term application of the technology. Furthermore, it emphasizes the importance of considering both technical and social aspects of the LCDP information system. The findings contribute to academia by providing a model-based capability framework, which serves as a structure for driving future research. Moreover, practitioners benefit from a practice-oriented and evaluated summary of initialization tasks and capabilities required for successful adoption.

Keywords: Low-Code Development Platforms, Citizen Development, Action Design Research, Organizational Capabilities, Socio-Technical Systems Model

1. Introduction

The COVID-19 pandemic has accelerated the adoption of new technologies in organizations undergoing digital transformation (Amankwah-Amoah et al., 2021) . These technologies often require innovative information technology (IT) solutions, leading to a constant increase in demand for corresponding software development and process automation (Hoogsteen & Borgman, 2022). However, organizations face two major challenges: First, there is a significant shortage of IT specialists, particularly in IT development (Binzer & Winkler, 2022). Second, digitization leads to business-specific demands that require a higher involvement of business end users in IT-related tasks (Urbach & Ahlemann, 2016). Referred to as citizen developers (CDs), these end users are anticipated to possess the ability to develop applications and workflows for themselves or others with no particular programming background and without the involvement of professional developers (Viljoen et al., 2023). The effective empowerment of these distinct personas leads to the democratization of software development, a phenomenon recognized to significantly enhance organizational innovation (Binzer & Winkler, 2022). Low-Code Development Platforms (LCDPs) - and No-Code Development Platforms as part of this market segment (Vincent et al., 2019) - present themselves as a suitable technological solution for establishing this concept of citizen development (CDE). They are defined as cloud-based platforms that allow business or IT users to develop applications or workflows with minimal or no programming code (Prinz et al., 2022). Market research institutes predict that by 2025, 70% of application development will be done using LCDPs (Wong et al., 2021). Despite their increasing popularity, research indicates that over 50% of organizations have yet to implement them (Beranic et al., 2020). A study involving over 1,000 leaders in low-code development reveals that the most significant barrier to adoption is a lack of familiarity with the technology and its usage (Creatio, 2021). As such, this paper aims to shed light on the essential capabilities that organizations must establish and master to successfully adopt LCDPs.

The topic of LCDPs has become increasingly important in scientific research, as evidenced by the significant rise in publications in recent years (e.g., Käss et al., 2022). Early literature focuses primarily on technology-related components of LCDPs (Prinz et al., 2021). Meanwhile, scientific publications also investigate the phenomena of LCDPs on a higher level: Some academics examine general drivers and

URI: https://hdl.handle.net/10125/106444
978-0-9981331-7-1

challenges of implementing and using LCDPs (Elshan et al., 2023; Käss et al., 2022; Prinz et al., 2022). Others explore specific challenges from the perspective of developers (e.g., Al Alamin et al., 2021). Research also evaluates the status quo of the adoption of LCDPs in the business environment of organizations (e.g., Beranic et al., 2020). Furthermore, Binzer and Winkler (2022) and Viljoen et al. (2023) both conduct multivocal literature reviews on CDE adoption from a procedural and thematical perspective and propose future research directions. However, we address two research gaps through this publication: First, recent LCDP research is primarily centered around exploratory studies that rely on expert interviews, surveys, or literature reviews, but lacks close involvement of practitioners in shaping research outcomes. Second, there remains a lack of overview of what organizations must be capable of during LCDP adoption, particularly from a managerial standpoint. Therefore, our objective is to address the following research question by employing practice-oriented action design research (ADR) grounded in the socio-technical systems (STS) model: *Which organizational capabilities do organizations require to master successful LCDP adoption?* By answering this research question, this study contributes to science as it provides a model-based framework of relevant capabilities to structure and drive future research. Moreover, practitioners gain an evaluated summary of capabilities to succeed in LCDP adoption.

In the upcoming chapter, we depict the current status quo of scientific research on LCDPs. Applying the ADR methodology, we subsequently outline the theoretical and organizational setup. Then, we elaborate on results and discussions, concluding with implications and directions for future research.

2. Status Quo

Forrester Research (Richardson & Rymer, 2014) first introduced the term LCDP in 2014. The authors define it as a platform that enables faster deployment of applications and workflows by reducing hand-coding. As time passed, further definitions add that LCDPs are mainly cloud-based and offer a Platform-as-a-Service (PaaS) model (Tisi et al., 2019) and extend it by explicitly mentioning that LCDPs target business but also professional IT developers (Prinz et al., 2022). With the term LCDPs gaining prominence in academic and organizational conversations, their practical significance has also grown. This is evident in the increased market potential over the years and future market forecasts: while low-code development technology revenue was at 9.2 billion USD in 2019 (Costello & Rimol, 2021), it increased to 22.5 billion USD in 2022 and is predicted to continue to grow exponentially (DeLisi & Howley, 2023). Scientists and practitioners mention several advantages of LCDPs like faster development, cost reduction, overcoming IT personnel shortage, and better cooperation between IT and business (e.g., Elshan et al., 2023; techconsult & smapOne, 2021). However, they also outline challenges like missing knowledge and trust in the technology, applications and data, security concerns, and insufficient governance (e.g., Prinz et al., 2022; techconsult & smapOne, 2021).

Having these challenges in mind, it is important for organizations to strategically set up prerequisites and develop approaches to overcome these barriers. Therefore, organizations require a holistic overview of all LCDP implementation and application tasks to be capable of successfully adopting LCDPs company-wide (Prinz et al., 2022). Scientists mention several potential tasks: First, they state that low-code application development and operation require different processes throughout the entire application lifecycle (Al Alamin et al., 2021). More specifically, organizations must define requirements for the respective application (e.g., Krejci et al., 2021) and, furthermore, develop (e.g., Henriques et al., 2018), review (e.g., Ragusa & Henriques, 2018), test (e.g., Al Alamin et al., 2021), deploy (e.g., Bock & Frank, 2021), and use (e.g., Silva et al., 2020) it. Additionally, organizations need to solve integration issues within the platform or with other systems (e.g., Al Alamin et al., 2021), regulate individual access-control (e.g., Brunschwig et al., 2020), and operate the platform (e.g., Overeem & Jansen, 2021). Furthermore, academics state that employees, in particular CDs, need to be trained (e.g., Tisi et al., 2019) and empowered (e.g., Hoogsteen & Borgman, 2022). For all of these described tasks, it is important to set up a respective governance (e.g., Heuer et al., 2022).

However, to our knowledge, no scientific publication has yet provided a comprehensive, theory-based, and practically evaluated overview of the necessary tasks to adopt LCDPs in an organization. Recognizing that a mere enumeration of tasks does not suffice for organizations to effectively navigate the process of adopting new technology on a management level, a suitable way to describe a coordinated set of tasks is by defining organizational capabilities (Helfat & Peteraf, 2003). Hence, defining organizational capabilities for LCDP adoption can direct academic exploration toward specific capabilities, enabling the creation of comprehensive results and artifacts to efficiently master these capabilities in real-world scenarios. Moreover, practitioners, especially managers, can derive significant value from these findings as they gain a comprehensive understanding

of what is necessary for the successful adoption of LCDPs within an organization.

3. Research Approach

This chapter explains how the ADR methodology relates to the research goal and outlines the theoretical foundation and organizational setting.

3.1 Action Design Research

LCDPs are a complex phenomenon and are still in the early stages of practical adoption. For this reason, we use ADR (Sein et al., 2011) to conduct practice-oriented research. This methodology helps to build and evaluate an IT artifact in a rapidly changing organizational setting (Heuer et al., 2022). ADR is useful for IS research that aims to develop a socio-technical design agenda that addresses a specific class of problems (Sein et al., 2011). It assumes that valid knowledge comes from various sources and the insights of researchers and practitioners are relevant to successful research (Moloney & Church, 2012). In general, ADR has found wide adoption in IS research (e.g., Schmager et al., 2023) and has also been used in the context of LCDP (Heuer et al., 2022). It follows four stages, each anchored by respective principles and tasks (Sein et al., 2011). We describe the first stage (1) *problem formulation* in this chapter. Then, we continue to summarize the results of the remaining three phases (2) *building, intervention, and evaluation (BIE)*, (3) *reflection and learning*, and (4) *formalization of learning* in chapter 4.

3.2 Stage 1: Problem Formulation

The subsequent paragraphs outline the chosen company, its identified LCDP-related issues, the theoretical basis, and the organizational setting and project schedule of the ADR team.

3.2.1. Company Introduction. ADR research often focuses on conducting a project in cooperation with one selected company as demonstrated by Sein et al. (2011). Drawing on the insights from the literature on case study research (Yin, 2013), we employed a systematic approach to select a participating organization that aligns with our research approach by defining three selection criteria: First, we prioritized companies with a positive financial performance, ensuring the inclusion of a well-established organization. Second, we considered organizations with a substantial number of business units and employees, enabling broader generalization of our findings to diverse companies. Third, we focused on organizations that exhibited a strategic commitment to accelerate their digital transformation, indicating their proactive stance towards and previous experiences in adopting new technologies. As a result, we chose a global engineering company operating in over 100 countries and employing more than 30,000 individuals. This organization demonstrated highly positive key performance indicators. Over the past years, they have also experienced significant acceleration in their digital transformation efforts. This growth involved digitizing their product range, adopting related technologies such as robotic process automation (RPA), and shifting to a more service-oriented business model with a strong digital focus.

The onset of the first ADR stage occurs when a problem is identified in practical situations and/ or when researchers anticipate its occurrence (Moloney & Church, 2012; Sein et al., 2011). We identified and conceptualized the research opportunity and formulated our research question based on both scientific and practical findings: Literature highlights the challenge of lack of knowledge and familiarity with this new technology and its usage (Creatio, 2021) and that research should provide a more holistic view of this research topic (Prinz et al., 2021). Additionally, we followed the approach of Sein et al. (2011) to identify the occurring practical problem by conducting semi-structured expert interviews, focus group discussions, and qualitative surveys with several stakeholders of the involved company: Their interest in adopting LCDPs stems from a rise in shadow IT due to their overcharged IT department. Therefore, they view LCDPs as a solution for non-IT users to create IT solutions independently, securely, and with simplified processes. They believe this adoption will curb shadow IT, drive digital transformation, and enhance digital skills. Consequently, they seek to devise a strategic overview of what to further develop and establish before, during, and after implementing LCDPs. Thus, we casted the described problem as an instance of a class of problems in which organizations encounter the obstacle of limited acquaintance with LCDP technology and its utilization. In response, we aimed to develop organizational capabilities to successfully adopt LCDPs.

3.2.2. Socio-Technical Systems Model as Foundation for Theory-Ingrained Artifact. As the outcome of an ADR project should be informed by theories or models (Sein et al., 2011), we identified the STS model (Bostrom & Heinen, 1977) as contributing theoretical basis and to be ingrained in the artifact by guiding its design. The STS model has found wide adoption in IS research (e.g., Hall & Haas, 2022) and

is also a common theoretical basis for conducting comprehensive LCDP research (e.g., Käss et al., 2023; Prinz et al., 2022). Therefore, we assume that it is also suitable for our research focus. Bostrom and Heinen (1977) assert that an information system involves the interaction of four interrelated variables in a technical and social system: More precisely, they state that *people* perform *tasks* with the help of a *technology* within an organizational *structure* (Bostrom & Heinen, 1977). Prinz et al. (2022) specified the STS model for LCDPs (Figure 1) and stated that respective people (*stakeholder-specific*) within the entire organization use the *platform-specific* technology of LCDPs to conduct *application-specific* development and operation tasks for the entire application lifecycle management (ALM). Therefore, companies must establish *organization-specific* structures such as low-code-related IT governance.

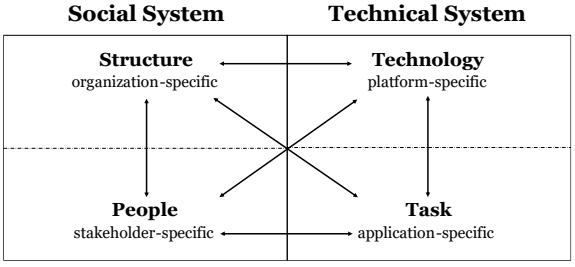

Figure 1. STS Model for LCDPs (Prinz et al., 2022)

In summary, organizations need to conduct specific tasks to fully master all four components of the STS model and overcome the barriers of adoption. Helfat and Peteraf (2003) state that "*a coordinated set of tasks, utilizing organizational resources, for the purpose of achieving a particular end result*" can be defined by an organizational capability. Capabilities enable organizations to compete and positively influence strategy and overall performance (O'Regan & Ghobadian, 2004). These described concepts serve as the theoretical basis for the subsequent phases of IT artifact development.

3.2.3 Organizational Setting for Practice-Inspired Research. Drawing from the theoretical foundations and identified problem, we delineated the scope, focus, mode of inquiry, as well as the roles and responsibilities in two four-hour workshop sessions. We secured a long-term commitment from the participating organization by setting up a project from September 2022 until the end of April 2023. Thereby, we identified different relevant topic areas and corresponding stakeholders: IT Governance, Enterprise and IT Architecture, Digital Workplace, and RPA. We asserted that establishing dedicated low-code governance is imperative for the successful adoption of LCDPs (e.g., Viljoen et al., 2023). Moreover, respective governance guidelines should be highly aligned with existing enterprise architecture and security guidelines (ISACA, 2018). Additionally, LCDPs foster the involvement of CDs which is why it is important to provide the required digital workplace but also involve them in the development of the IT artifact to gather requirements and decisions (Heuer et al., 2022). Lastly, we included internal RPA adepts with adoption experiences in the team as RPA is mentioned as quite similar technology (Bock & Frank, 2021). In an endeavor to reflect the pertinent subjects from a scientific standpoint, our research team comprised experts in LCDPs, IT Governance, Enterprise Architecture (EA), and RPA (Table 1).

Table 1. Organizational Setting of ADR team

Industry	Engineering
Staff	Around 30.000 Employees
Practitioners	- IT Governance (Head) - Enterprise & IT Architecture (Head & 2 employees) - Digital Workplace Platforms (Head & 2 employees) - RPA (Head & 1 employee) - Business (2 Potential CDs) - Project Manager
Researchers	- 1 Researcher (LCDPs) - 1 Researcher (IT Governance & EA) - 1 Researcher (RPA)

To accomplish the objective of our ADR project, we decided to differentiate between different phases of LCDP adoption as recent research implies: Wulf (2020) mentions that PaaS adoption can be divided into a (pre-)adoption and post-adoption phase. Furthermore, research by Viljoen et al. (2023) on scaling CDE identifies four adoption stages: They propose that effective adoption evolves from a lack of experience and knowledge (enablement), progresses to establishing the groundwork (foundation), advances to creating a suitable environment (growth), and finally culminates in implementation and value generation (saturation). Lastly, Prinz et al. (2022) state that LCDP adoption includes an implementation (initiation and technical realization) and application (utilization and management) phase. Based on these findings and the firm's prior learnings from the adoption of related technologies, we opted for a two-step approach: We decided to first define initialization tasks prior to the actual implementation, and second, derive organizational capabilities required for the management of LCDPs. Therefore, we organized workshops with the identified stakeholders to design and evaluate the artifact. In doing so, we adhered to the workshop guidelines and principles outlined by Thoring et al. (2020) that render them a suitable research methodology for ADR projects. For evaluation purposes, we predominantly employed the research methods of group discussion and artifact

analysis, which the authors regard as appropriate approaches for ensuring workshop outcome quality. Furthermore, we chose the IT-dominant BIE as the research design continuum. Sein et al. (2011) argue that this approach is suitable when creating an innovative design with the input of practitioners' first-hand experiences, focusing on a highly participatory process, and when early designs and alpha versions should be instantiated and evaluated in a rather limited organizational context. Therefore, we carried out the BIE stage of the alpha version within the organizational context of the ADR team, incorporating the diverse perspectives of all stakeholders. However, we took the beta version of the artifact into a wider organizational setting to comprehensively intervene and evaluate it based on the use context, perceptions, and experiences of the end users. We selected IT management as the respective end users for evaluating the beta version for its suitability and potential impact. This decision was motivated by the fact that organizational capabilities are well established in practice-oriented approaches for IT management like TOGAF (The Open Group, 2018) and should serve as a management guideline to adopt LCDPs in the entire organization. Table 2 summarizes the ADR project schedule and the respective methodology.

Table 2. ADR project

ADR Stage	Project Schedule & Methodology
Stage 1: Problem Formulation	**September 2022**: Problem Formulation 1.0 (semi-structured expert interviews, focus group discussions, and qualitative survey to identify and conceptualize the research opportunity and question) **October 2022:** Problem Formulation 2.0 (2x 4-hour workshops to define scope, focus, mode of inquiry, and roles and responsibilities)
Stage 2: BIE	**October – November 2022:** BIE 1.0 (8x 4-hour workshops to define initialization tasks for LCDP implementation) **December 2022:** 1x 4-hour ADR team-internal workshop to evaluate alpha version **January– March 2023**: BIE 2.0 (6x 4-hour workshops to derive organizational LCDP capabilities) **March 2023:** 1x 4-hour IT Management workshop to evaluate the beta version
Stage 3: Reflection and Learning	**September 2022 – March 2023:** Continuous reflection and learning by the ADR team
Stage 4: Formalization of Learning	**March – April 2023:** Formalization of learning by researchers and assessment with practitioners

4. Results

This chapter summarizes the outcomes of the ADR approach based on the earlier presented stages.

4.1. Stage 2: Building, Intervention, and Evaluation (BIE)

Table 3 provides a summary of the main tasks during LCDP implementation within the organization. The STS model and its four components serve as a basis for classification, guaranteeing a comprehensive analysis of the tasks for the entire information system. Certain tasks can be carried out simultaneously, while others need to be completed sequentially.

Table 3. Initialization Tasks for LCDP Implementation

STS Component	Task Description
Organization-specific	- Provide prerequisites for the use of cloud-based platforms for all employees - Analyze internal IT solution demand - Monitor external LCDP market - Set up and evaluate LCDP selection decision criteria - Decide for appropriate LCDP and licensing model - Set up connector catalogue - Set up governance roles, responsibilities & processes, aligned with technical LCDP functionality - Set up enterprise architecture management concept - Set up application portfolio management concept - Develop empowerment and implementation strategy
Platform-specific	- Set up LCDP environments - Grant user-specific access - Provide APIs to existing systems - Set up platform operation model
Application-specific	- Define application classes - Develop set of questions and response options for classification process - Define DevOps quality standards throughout entire application lifecycle for each application class.
Stakeholder-specific	- Develop training concept (including training methods & platform) - Develop consultancy concept

First, to implement LCDPs within an organization, the firm needs to provide the prerequisites for company-wide usage of all kinds of cloud-based platforms. The provision of appropriate technical infrastructure addresses two key aspects: On the one hand, LCDPs are predominantly cloud-based in today's context (Tisi et al., 2019), necessitating organizations to align themselves with the offerings available in the market. On the other hand, cloud-based LCDPs facilitate extensive low-code utilization across the entire global organization and reduce shadow IT (Binzer & Winkler, 2022) by fostering transparency through storing specific information in the cloud for all applications developed within the platform (Prinz et al., 2022). Consequently, the relevant stakeholders are required to analyze the internal IT solution demands while also keeping a close watch on the external LCDP market to ensure sustainable and well-informed decision-making for a suitable LCDP. Therefore, they need to develop and evaluate criteria for LCDP selection.

Following the selection of the LCDP(s), the organization can proceed with providing the platform(s). It primarily involves configuring various

environments and granting appropriate user access. Additionally, it is essential to ensure the availability of data by providing APIs to existing systems. Moreover, the responsible parties also need to establish a platform operation model, which involves defining processes for managing the LCDP vendor(s), handling potential updates, and preparing for future changes.

In addition to the technical provision, current LCDP research (Heuer et al., 2022) proposes a classification of applications into various types, a practice previously undertaken in research related to shadow IT (Zimmermann et al., 2016). This approach offers the advantage of having adaptive, yet distinct quality standards and responsibilities based on different application classes, thereby reducing the reliance on IT involvement for specific types of applications. Consequently, the organization needs to develop two corresponding concepts: First, they must establish a classification process that includes a set of questions and response options to assign potential applications to their respective class. Second, they need to define guardrails for low-code application development and operations (DevOps) across the entire application lifecycle for each class (Binzer & Winkler, 2022).

Moreover, the organization should provide training and consultation to CDs at various stages of utilizing the LCDP, despite having well-defined quality standards. This necessity arises for two reasons: First, business personnel might not fully grasp the opportunities that LCDPs offer, thereby impeding the effective encouragement and dissemination of knowledge regarding the employment of the LCDP(s) (Hoogsteen & Borgman, 2022). Second, the absence of adequate training could lead to employees struggling to comprehend and adhere to the mandatory DevOps quality standards, undermining both development efficiency and output quality (Viljoen et al., 2023). Consequently, the organization needs to develop training concepts and methods, in addition to devising an appropriate training platform. This endeavor aims to establish a clear learning path (Bernsteiner et al., 2022) that only developers possessing the requisite certification level are granted authorization to engage in DevOps within designated application classes. Furthermore, it is essential to develop a consultancy concept accompanied by a suitable organizational structure to effectively guide developers in implementing their acquired knowledge. Additionally, the organization should offer a connector catalog consisting of available and verified connectors to support developers in securely utilizing them in their development process.

The organization should also accord priority to governance-related prerequisites prior to the implementation phase (Viljoen et al., 2023). To establish an effective and comprehensive governance, that is aligned with centralized governance structures (Hoogsteen & Borgman, 2022), the delineation of roles, responsibilities, and relational mechanisms ensuring adherence to this framework becomes essential (Prinz et al., 2022). In relation to this, it is necessary to develop an enterprise architecture management (EAM) concept. This includes processes for aligning the demand for low-code applications (LCAs) with the strategic goals of the business units and facilitating knowledge exchange regarding potentially similar LCAs or components. Moreover, organizations require an overall transparency of the LCA portfolio. This can be achieved by managing two aspects: First, they should establish a concept and platform for monitoring of all developed LCAs. Second, they need to implement a process that ensures compliance with defined governance guidelines throughout the entire lifecycle, especially when facing changes or an increase in the number of users. Finally, the organization must establish an empowerment and implementation strategy, including, among others, the establishment of a community to raise awareness for the potential of the LCDP (Binzer & Winkler, 2022; Hoogsteen & Borgman, 2022) and foster a continuous engagement of business units in the entire adoption process (Heuer et al., 2022).

Table 3 provides a summary of the organizational capabilities essential for the adoption of LCDPs. These capabilities have been extracted from the initialization tasks and encompass the perspectives and experiences of the ADR team concerning technologies that are both related to LCDPs and have already been implemented and managed in the company (such as RPA).

An initial step for successful LCDP adoption is to *provide and manage the necessary technical infrastructure*. This implies providing the prerequisites for all employees to utilize cloud-based platforms and continuously managing the infrastructure to adapt to any potential technological changes. As an ongoing process, the organization must also effectively *manage the LCDP portfolio* by assessing whether there is a need for a new platform or if the existing LCDP(s) remain adequate.

Furthermore, the organization requires the capabilities to *provide* and then *operate the LCDP(s)*, as well as to *provide the necessary connectors and data*. As mentioned in the initialization tasks, certain prerequisites need to be fulfilled to initially provide and operate the platform and connectors. However, the management of these components is an ongoing

process in case the selected platform(s) and the organization's low-code usage mature.

Table 3. LCDP Organizational Capabilities

Organizational Capability	The organization should be able to…
Provide and Manage Technical Infrastructure	*…set up and manage the necessary company-wide infrastructure for general LCDP usage.*
Manage LCDP Portfolio	*…monitor the external LCDP market & internal IT solution demand, and drive and decide whether to implement a new LCDP.*
Provide Platform	*…set up the LCDP.*
Provide Connectors & Data	*…supply respective new connectors for data usage.*
Operate Platform	*…manage the LCDP vendors and update platform-specific changes and features.*
Classify Use Case	*…assign demanded low-code solutions to respective application classes.*
Define Requirements	*…specify functional and non-functional requirements for a desired LCA.*
Develop Low-Code Application	*…create LCA initially and throughout its life cycle.*
Test Low-Code Application	*…ensure that a LCA behaves in the intended way.*
Approve Low-Code Application	*…verify that all necessary DevOps quality standards for a LCA are met.*
Deploy Low-Code Application	*…migrate a developed LCA to a productive state.*
Operate Low-Code Application	*…manage a developed LCA.*
Use Low-Code Application	*…utilize a developed LCA.*
Support End User	*…assist end users in using a LCA.*
Provide Low-Code Training	*…train all potential developers regarding the specific scope of usage.*
Consult Citizen Developers	*…consult developers in terms of platform usage and DevOps.*
Manage Connectors & Access-Control	*…provide a connector catalogue of existing connectors and manage access for the respective stakeholders.*
Ensure Low-Code Governance	*… enforce the predefined governance requirements during the entire application lifecycle.*
Manage and Control LCA Demands	*…consult application owners in business fit (reusability of existing low-code solutions, strategic alignment, and platform selection).*
Manage LCA Portfolio	*…provide an overview of all developed LCAs.*
Empower Low-Code	*…encourage each employee for low-code adoption and usage.*

Moreover, eight organizational capabilities play a crucial role for stakeholders involved in the low-code application DevOps process, spanning from the conceptualization of the desired application to its actual usage. The capabilities involve that the respective stakeholders need to first *classify the use case* to determine the relevant quality standards and associated responsibilities for DevOps. Subsequently, they should *define requirements*, and proceed to *develop* and *test the low-code application*. Furthermore, they need to *approve and deploy the LCA*. Additionally, the organization must possess two further capabilities: *operate the application* through tasks such as change management and *support the end users* of the developed and operated LCA.

As mentioned earlier, various stakeholders play a critical role throughout the entire usage of the LCDP. Therefore, the organization needs to continuously *provide low-code training* and *consult citizen developers* throughout the entire ALM process; even after a company-wide roll-out of the platform. This is especially necessary because the functionality of LCDPs may evolve in the future which could inquire e.g., modification of the defined DevOps quality standards due to new features.

When the organization's low-code usage matures, another management priority is to continuously *manage connectors & access control*. This can be achieved by maintaining the initially developed connector catalog that encompasses existing solutions and adheres to general principles of data access. Moreover, the catalog should consider the appropriate level of certification required for connector usage.

Furthermore, the capability to *ensure low-code governance* necessitates a management process that closely aligns with the evolving functionality of the LCDP(s). This includes adapting responsibilities, processes, and quality standards in response to the maturation of low-code usage and the evolving LCDP features. In the medium and long term, the aspects of *EAM* and *application portfolio management* also continue to hold significance. This entails the ongoing *monitoring of demands* to assess the suitability of existing solutions and their alignment with the business strategy. Additionally, it involves the *control of developed LCAs* in accordance with the defined DevOps quality standards.

Ultimately, the organization must *empower low-code* and facilitate the widespread adoption, beginning during the set-up of the environment until saturation and value creation of the platform.

4.2. Stage 3: Reflection and Learning

Throughout the project, the ADR team maintained a consistent practice of reflection upon the design and outcomes of all workshops. Subsequently, we analyzed intervention results according to the stated goals following the ADR approach presented by Sein et al. (2011). By meticulously defining our scope, mode of inquiry, and roles and responsibilities during the two inaugural workshops, we managed to adhere to the initial agreements during the entire project. Nonetheless, especially the practical assessment of different LCDPs and the execution of the initialization tasks provided us with actionable insights.

During developing a proof-of-concept LCA for the maintenance department that involved the integration of multiple connectors with leading systems, we noticed a recurring pattern: First, the setup of non-standardized connectors to other systems mandates additional manual configuration and often requires the expertise of IT professionals and system owners. Drawing insights from these experiences, we

also discerned that the extent to which an LCDP provides technical support for governance assurance can fluctuate depending on the functionalities inherent in the chosen platform. Consequently, such variability can additionally influence the degree of IT involvement in "manual gate processes".

Second, we recognized the intricate interconnection between the capabilities of low-code DevOps and the ALM processes that pertain to traditional "high-code" software development already established within the organization. Nonetheless, we discerned notable disparities in the approval and deployment processes, particularly in the realm of CDE. The imperative of maintaining security amid the technical transition to a productive state prompted us to acknowledge a shared responsibility, often spanning both business and IT departments. This is a departure from the original strategy of the company that allocated all DevOps tasks to the respective business developers in the first instance. Therefore, the active involvement of IT experts also remains pivotal for the secure operation of the developed applications.

Third, our reflections led us to the conclusion that various empowerment strategies hold the potential to enhance the adoption of LCDPs across the entire organization. In addition to the previously mentioned establishment of a community, another effective approach is the identification of business employees who are already involved in shadow IT activities utilizing tools like *Microsoft Excel* and *Visual Basic for Applications* (VBA). Leveraging their existing skill sets and knowledge, these individuals can act as pilot users to test the entire approach and assess the clarity of guidelines, training concepts, and other associated aspects. Furthermore, we recognized that the integration of empowerment efforts into the demand management process or through facilitated dialogues between the IT department and business units, particularly as organizational maturity advances, holds promise for promoting collaboration and aligning strategic objectives. Additionally, the organization can incentivize professional IT developers to embrace LCDPs to expedite the development process.

Lastly, it is noteworthy that the maturity and specificity of each capability might be contingent upon the strategic orientation of the company's LCDPs vision and the organizational context: Firms aspiring to achieve a company-wide adoption of LCDPs, with a particular focus on fostering CDE, might exhibit higher levels of capability maturity and specification. In contrast, if organizations are primarily focused on accelerating the development of professional developers, the extent of maturity and specification in areas such as training, consulting, and certification processes, as well as empowerment methods, might not be as conspicuous. Nevertheless, it remains imperative to establish a foundational threshold of requisite specifications for each capability.

4.3. Stage 4: Formalization of Learning

The fourth stage of ADR involves formalizing learning by developing general solution concepts based on the principles of situated learning (Sein et al., 2011). In our research, we applied this stage to abstract the learnings obtained from the study and create solution concepts for the defined specific class of field problems. This process included documenting the achievements observed in terms of the IT artifact and organizational outcomes as well as using these findings to refine the theoretical model. As part of enhancing the theoretical STS model for LCDPs, the researchers categorized the 21 identified organizational capabilities into respective components of the model. Figure 2 provides a general overview of the LCDP organizational capability framework. By assigning the capabilities to all components of the STS model, we ensure a comprehensive analysis of the LCDP information system and provide an IT artifact for organizations to evaluate the specific areas of expertise required for successfully adopting LCDPs.

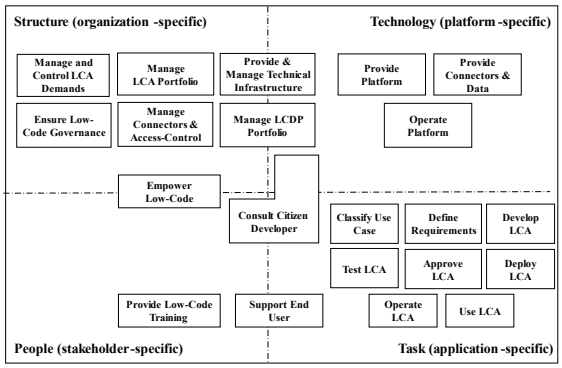

Figure 2. LCDP Organizational Capability Framework

Research states that challenges can arise across all four components of the STS model (Elshan et al., 2023; Käss et al., 2023; Prinz et al., 2022). Therefore, mastering all capabilities is essential for organizations to successfully adopt LCDPs. To ensure generalizability, we analyzed the key practical barriers of adoption identified in the three mentioned scientific studies and evaluated the framework by examining whether mastery of the developed capabilities would solve these challenges. Organizations can overcome the dare of resistance against change among employees through a sufficient empowerment strategy (*empower low-code)*, along with the capabilities to *provide training* and *consult citizen developers* at all

stages of low-code usage. Moreover, establishing an application classification model and defining appropriate DevOps quality standards and responsibilities to *ensure low-code governance* mitigates governance, security, compliance, privacy risks, and the potential increase of shadow IT. *Manage and control the LCA demand, approve the LCA,* and *manage the LCA portfolio* are also essential for a well-governed application landscape. Furthermore, organizations can address the challenge of vendor lock-in if they possess the capability to continuously *manage the LCDP portfolio* to evaluate the need for additional platforms. Therefore, we consider the developed framework as a solution for addressing the identified class of problems: shedding light on all organizational capabilities required to successfully adopt LCDPs.

5. Conclusion

With the rapid adoption of new technological advancements in organizations and the increasing demand for software development and process automation, LCDPs have gained significant relevance. However, there is still a lack of comprehensive understanding regarding best practices for implementing and applying this emerging technology in organizational contexts. This scientific publication aims to shed light on all required organizational capabilities to effectively overcome the existing challenges of LCDP adoption. Using the ADR methodology and the STS model, this study develops a theory-based framework of 21 organizational capabilities that companies must cultivate to facilitate the seamless and successful adoption of LCDPs. The findings emphasize the significance of conceptual development as a prerequisite before implementation. This preliminary step is essential for effectively managing LCDPs by establishing the necessary organizational capabilities for their long-term application. Moreover, this publication underscores the importance of considering both the technical and social aspects to comprehensively overcome the barriers associated with LCDP adoption.

The results contribute to academia by providing a scientifically developed model-based framework of organizational capabilities required for the successful implementation and application of LCDPs. The framework serves as a structured and comprehensive approach for facilitating future research in this research field. From a practical perspective, this publication presents insights for practitioners, especially managers, seeking to adopt LCDPs in their organizations. It offers a practice-related and evaluated summary of the capabilities that need to be mastered for successful adoption. Organizations can now strategically set up prerequisites and develop approaches to overcome barriers associated with LCDP adoption and comprehensively manage these platforms during application.

Although this publication contributes to the overall understanding of the organizational capabilities that are essential for the successful adoption of LCDPs, there are certain limitations: The study's contextual focus may limit the generalizability of the findings beyond the specific organizational setting and industry examined. Even though we chose the organization to the best of our knowledge and defined selection criteria as well as evaluated the capabilities with the key challenges for LCDP adoption during the formalization of the learning stage, replicating and validating the results in diverse contexts and with different methodologies could enhance their broader applicability. In addition, our endeavors have revolved around the development of organizational capabilities rooted in the theoretical framework of the STS model. Nevertheless, it is conceivable that extending this development from the vantage points of procedural and/or contingency perspectives could yield an enriched array of capabilities and empirical insights. Moreover, future research could consider investigating the long-term effects of LCDP adoption and explore how organizations increase their maturity in the post-implementation phase. Lastly, it is important to acknowledge that this framework serves as a foundation for the adoption of LCDPs. However, additional research is needed to provide a more detailed specification of each capability.

6. References

Al Alamin, M. A., Malakar, S., Uddin, G., Afroz, S., Bin Haider, T., Iqbal, A., Alamri, B., Javed, I. T., & Margaria, T. (2021). An Empirical Study of Developer Discussions on Low-Code Software Development Challenges. *IEEE/ACM International Conference on MSR*.

Amankwah-Amoah, J., Khan, Z., Wood, G., & Knight, G. (2021). Covid-19 and digitalization: The great acceleration. *Journal of Business Research*, *136*, 602–611.

Beranic, T., Rek, P., & Heričko, M. (2020). Adoption and Usability of Low-Code/No-Code Development Tools. *CECIIS 2020*.

Bernsteiner, R., Schlögl, S., Ploder, C., Dilger, T., & Brecher, F. (2022). Citizen vs. Professional Developers: Differences and Similarities of Skills and Training Requirements for Low Code Development Platforms. *ICERI 2022*.

Binzer, B., & Winkler, T. J. (2022). Democratizing Software Development: A Systematic Multivocal

Literature Review and Research Agenda on Citizen Development. In N. Caroll, A. Nguyen-Duc, X. Wang, & V. Stray (Eds.), *Software Business. ICSOB 2022* (pp. 244–259). Springer, Cham.

Bock, A. C., & Frank, U. (2021). Low-Code Platform. *Business & Information Systems Engineering*, *63*(6), 733–740.

Bostrom, R. P., & Heinen, J. S. (1977). MIS Problems and Failures: A Socio-Technical Perspective Part I: The Causes. *MIS Quarterly*, *1*(3), 17–32.

Brunschwig, L., Guerra, E., & de Lara, J. (2020). Towards access control for collaborative modelling apps. *MODELS '20*.

Costello, K., & Rimol, M. (2021). *Gartner Forecasts Worldwide Low-Code Development Technologies Market to Grow 23% in 2021*. www.gartner.com

Creatio. (2021). *The State of Low-Code/No-Code*. www.creatio.com

DeLisi, M. R., & Howley, C. (2023). *Gartner Forecasts Worldwide Low-Code Development Technologies Market to Grow 20% in 2023*. www.gartner.com

Elshan, E., Dickhaut, E., & Ebel, P. A. (2023). An Investigation of Why Low Code Platforms Provide Answers and New Challenges. *HICSS 2023*.

Hall, M., & Haas, C. (2022). A Research Agenda to Understand Drivers of Digital Gullibility. *ICIS 2022*.

Helfat, C. E., & Peteraf, M. A. (2003). The Dynamic Resource-Based View: Capability Lifecycles. *Strategic Management Journal*, *24*(10), 997–1010.

Henriques, H., Lourenço, H., Amaral, V., & Goulão, M. (2018). Improving the Developer Experience with a Low-Code Process Modelling Language. *MODELS '18*.

Heuer, M., Kurtz, C., & Böhmann, T. (2022). Towards a Governance of Low-Code Development Platforms Using the Example of Microsoft PowerPlatform in a Multinational Company. *HICSS 2022*.

Hoogsteen, D., & Borgman, H. (2022). Empower the Workforce, Empower the Company? Citizen Development Adoption. *HICSS 2022*.

ISACA. (2018). *Cobit 2019 Framework: Governance and Management Objectives.*

Käss, S., Strahringer, S., & Westner, M. (2022). Drivers and Inhibitors of Low Code Development Platform Adoption. *IEEE CBI 2022*.

Käss, S., Strahringer, S., & Westner, M. (2023). Practitioners' Perceptions on the Adoption of Low Code Development Platforms. *IEEE Access.*

Krejci, D., Iho, S., & Missonier, S. (2021). Innovating with employees: an exploratory study of idea development on low-code development platforms. *ECIS 2021*.

Moloney, M., & Church, L. (2012). *Engaged Scholarship: Action Design Research for New Software Product Development.*

O'Regan, N., & Ghobadian, A. (2004). The importance of capabilities for strategic direction and performance. *Management Decision*, *42*(2), 292–313.

The Open Group. (2018). *TOGAF Guide: Business Capabilities*. https://publications.opengroup.org/g189

Overeem, M., & Jansen, S. (2021). Proposing a Framework for Impact Analysis for Low-Code Development Platforms. *ACM/IEEE International Conference on MODELS-C*.

Prinz, N., Huber, M., Riedinger, C., & Rentrop, C. (2022). Two Perspectives of Low-Code Development Platform Challenges – An Exploratory Study. *PACIS 2022*.

Prinz, N., Rentrop, C., & Huber, M. (2021). Low-Code Development Platforms – A Literature Review. *AMCIS 2021*.

Ragusa, G., & Henriques, H. (2018). Code review tool for Visual Programming Languages. *IEEE Symposium on VL/HCC*.

Richardson, C., & Rymer, J. R. (2014). *New Development Platforms Emerge For Customer-Facing Applications.* Forrester. www.forrester.com

Schmager, S., Grøder, C. H., Parmiggiani, E., Pappas, I., & Vassilakopoulou, P. (2023). What Do Citizens Think of AI Adoption in Public Services? Exploratory Research on Citizen Attitudes through a Social Contract Lens. *HICSS 2023*.

Sein, M. K., Henfridsson, O., Purao, S., Rossi, M., & Lindgren, R. (2011). Action Design Research. *MIS Quarterly*, *35*(1), 37–56.

Silva, C., Vieira, J., Campos, J. C., Couto, R., & Ribeiro, A. N. (2020). Development and Validation of a Descriptive Cognitive Model for Predicting Usability Issues in a Low-Code Development Platform. *Human Factors*, *63*(6), 1–21.

techconsult & smapOne. (2021). *Low-Code-/No-Code-Development: Enabler der digitalen Transformation*. www.smapone.com

Thoring, K., Mueller, R., & Badke-Schaub, P. (2020). Workshops as a Research Method: Guidelines for Designing and Evaluating Artifacts Through Workshops. *HICSS 2020*.

Tisi, M., Mottu, J., Kolovos, D., de Lara, J., Gurerra, E., Di Ruscio, D., Pierantonio, A., & Wimmer, M. (2019). Lowcomote: Training the Next Generation of Experts in Scalable Low-Code Engineering Platforms. *STAF 2019*.

Urbach, N., & Ahlemann, F. (2016). *IT-Management im Zeitalter der Digitalisierung: Auf dem Weg zur IT-Organisation der Zukunft*. Springer.

Viljoen, A., Nguyen, J., Kauschinger, M., & Hein, A. (2023). Fostering Scalable Citizen Development in Organizations: Towards a Guiding Framework. *AMCIS 2023*.

Vincent, P., Iijima, K., Driver, M., Wong, J., & Natis, Y. (2019). *Magic Quadrant for Enterprise Low-Code Application Platforms*. www.gartner.com

Wong, J., Iijima, K., Leow, A., Jain, A., & Vincent, P. (2021). *Magic Quadrant for Enterprise Low-Code Application Platforms*. www.gartner.com

Wulf, F. (2020). Platform-As-A-Service Post-Adoption by Large-Scale Organizations: A Research Model Design. *ECIS 2020*.

Yin, R. K. (2013). *Case Study Research: Design and Methods* (5th ed.). Sage Publications, Inc.

Zimmermann, S., Rentrop, C., & Felden, C. (2016). Governing Identified Shadow IT by Allocating IT Task Responsibilities. *AMCIS 2016*.