

Erfolgsfaktoren für die Implementierung von Low-Code/No-Code-Plattformen in Unternehmen

Low-Code/No-Code erfolgreich implementieren

J. Pydde, R. Ziegler, A. Mezger

ZUSAMMENFASSUNG Unternehmen müssen sich durch die digitale Transformation flexibel anpassen. Low Code/No Code (LCNC) bietet hierfür eine Lösung, indem es die Softwareentwicklung vereinfacht. Diese Arbeit identifiziert Erfolgsfaktoren für die Implementierung von LCNC-Lösungen in produzierenden Unternehmen. Die Literatur wird in Mensch, Technologie, Organisation und Aufgabe kategorisiert. Ein Prisma-basierter, systematischer Literaturreview dient der Analyse, um einen Leitfaden für die Implementierung von LCNC-Lösungen zu erstellen.

STICHWÖRTER

Industrie 4.0, Digitalisierung, Software

Successful implementation of low-code/no-code apps – Success factors for the implementation of low-code/no-code platforms in companies

ABSTRACT Companies need to be able to adapt flexibly in the context of digital transformation. Low Code/No Code (LCNC) offers a solution by simplifying software development. This work identifies success factors for implementing LCNC solutions in manufacturing companies. Literature is categorized into people, technology, organization, and tasks. A Prisma-based systematic literature review is used for analysis to create a guide for the implementation of LCNC solutions.

1 Einleitung

Eine der großen Herausforderungen der digitalen Transformation ist der hierzulande herrschende Fachkräftemangel. Laut einer Studie des Digitalverbands Bitkom von 2023 fehlen in Deutschland rund 149 000 IT-Fachkräfte. Dementsprechend fehlen auch wichtige IT-Kompetenzen, was die Innovationskraft und Wettbewerbsfähigkeit vieler Unternehmen erheblich beeinträchtigt [1]. Eine Möglichkeit, die Auswirkungen dieser Problematik einzudämmen besteht darin, die Komplexität des Entwicklungs- und Optimierungsprozesses von Software-Applikationen zu reduzieren, sodass auch Mitarbeitende ohne IT-Kenntnisse („Citizen Developer“) eigene Anwendungen programmieren können, um diese in die bestehende Infrastruktur zu implementieren [2].

Low-Code-/No-Code-Ansätze (LCNC) erlauben es, Softwareanwendungen zu entwickeln, ohne tiefgehende Programmierkenntnisse zu benötigen. Dies erfolgt durch visuelle Entwicklungsumgebungen (LCNC-Plattformen) und vorkonfigurierte Module (etwa Softwarebausteine und Konnektoren), die es auch Fachabteilungen und Nicht-Programmierern ermöglichen, maßgeschneiderte Lösungen zu erstellen [3].

Zwar bedeutet die Vereinfachung des gesamten Entwicklungsprozesses gewisse Einschränkungen bei den Freiheitsgraden solcher LCNC-Anwendungen, jedoch erlaubt diese Vereinfachung auch, das Domänenwissen der jeweiligen Prozessexperten direkt bei der Applikationsentwicklung zu berücksichtigen. Diese Vor- und Nachteile von LCNC-Plattformen und Anwendungen werden in der Literatur vielfach diskutiert [4, 5].

Deshalb widmet sich der vorliegende Beitrag der Frage, welche Aspekte und Bedingungen zur erfolgreichen Implementierung und Nutzung solcher LCNC-Plattformen im Unternehmen beitragen. Ergänzend wird dabei auch untersucht, welche vor- und nachgelagerten Schritte die Effizienz der Einführung einer LCNC-Plattform steigern können.

2 Verwandte Arbeiten und Terminologie

Die digitale Transformation und Industrie 4.0 sind zentrale Konzepte, die in der aktuellen Forschung häufig behandelt werden. Digitale Transformation beschreibt den umfassenden Prozess, digitale Technologien in alle Aspekte eines Unternehmens zu integrieren. Dies hat grundlegende Veränderungen in der Art und Weise der Tätigkeit von Unternehmen und der Schaffung von Werten zur Folge [6]. Der Ausdruck Industrie 4.0 repräsentiert die vierte industrielle Revolution. Dabei erfolgt die Vernetzung von Maschinen, Sensoren und Systemen in der Herstellung [7].

Low Code/No Code (LCNC) ist eine Technologie, die es ermöglicht, Softwareanwendungen mit minimalem oder keinem Programmieraufwand zu erstellen. Low-Code-Plattformen bieten visuelle Entwicklungsumgebungen, die Nutzern erlauben, durch Drag-and-Drop und vorkonfigurierte Komponenten komplexe Anwendungen zu erstellen [8]. No-Code-Plattformen hingegen sind vollständig darauf ausgelegt, dass Anwender ohne Programmierkenntnisse Software entwickeln können. Diese Ansätze fördern die Effizienz in der Softwareentwicklung, indem sie die

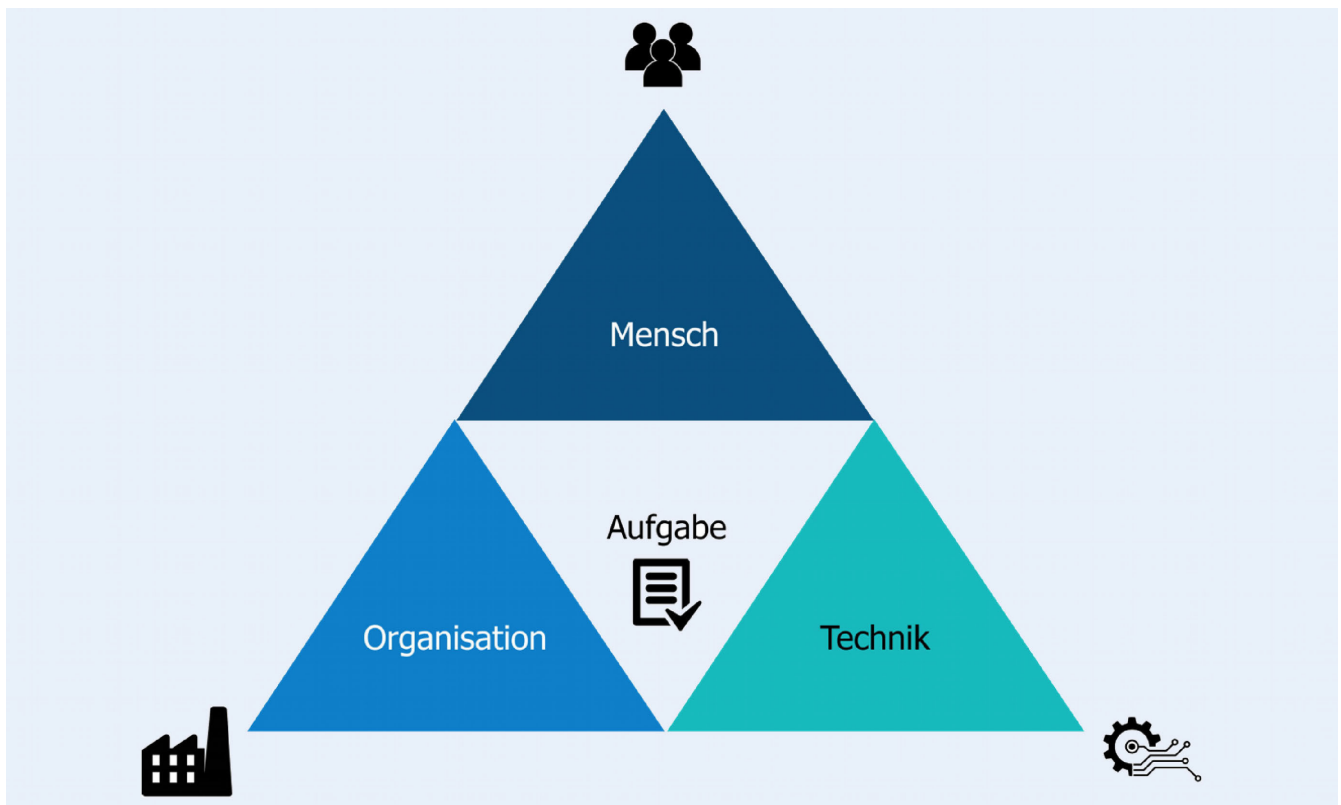


Bild 1. Erweiterte MTO (Mensch, Technologie, Organisation)-Darstellung. Grafik: in Anlehnung an [10]

Eintrittsbarrieren senken und die Produktivität sowohl von Entwicklern als auch von Fachabteilungen steigern [9].

Das MTO-Modell (Mensch, Technologie, Organisation) ist ein bewährtes Rahmenwerk zur Analyse und Gestaltung von Arbeitsprozessen in der Industrie [10]. In Zeiten der digitalen Transformation und Industrie 4.0 ist dieses Modell besonders relevant, da es die Wechselwirkungen zwischen den drei Dimensionen berücksichtigt, welche für den Erfolg von Technologien wie LCNC entscheidend sind:

- **Mensch:** Die Einbeziehung des menschlichen Faktors ist grundlegend, da Akzeptanz und Engagement der Mitarbeitenden maßgeblich für die erfolgreiche Implementierung von LCNC-Lösungen sind. Im Kontext von LCNC verändert sich die Rolle des Citizen Developer. Er ist nicht nur Nutzer, sondern wird auch zum Entwickler von einfachen Softwareanwendungen [11].
- **Technologie:** Der technologische Aspekt ist wichtig, da die Verfügbarkeit und Benutzerfreundlichkeit der LCNC-Plattformen entscheidend für deren Akzeptanz und Effektivität sind [12]. Die Technik muss so gestaltet sein, dass sie die Benutzer unterstützt und nicht überfordert.
- **Organisation:** Die organisatorischen Rahmenbedingungen beeinflussen maßgeblich, wie Technologien implementiert und genutzt werden. Eine agile Organisation, die schnelle Entscheidungen und Anpassungen ermöglicht, ist unerlässlich für den Erfolg von LCNC-Initiativen [13].

Das MTO-Modell wurde aufgrund der Notwendigkeit, den spezifischen Anforderungen und Dynamiken der Industrie 4.0 gerecht zu werden, um die Kategorie „Aufgabe“ ergänzt, siehe **Bild 1**.

In diesem Kontext spielen die Aufgaben, die von den Mitarbeitenden zu erfüllen sind, zunehmend eine zentrale Rolle [14].

- **Aufgabe:** Definition und Strukturierung von Aufgaben sind entscheidend, um Effizienz und Effektivität bei der Nutzung von LCNC-Lösungen zu gewährleisten. Die Automatisierung von Routineaufgaben und die Übertragung von Verantwortung an Mitarbeitende, die in der Lage sind, diese Technologien zu nutzen, erfordern eine klare Aufgabenverteilung und Aufgabenbeschreibung [14].

In der aktuellen Forschung zu LCNC-Plattformen gibt es eine Vielzahl von Studien, die sich mit verschiedenen Aspekten dieser Technologie befassen. Viele Arbeiten konzentrieren sich auf die Vorteile und Herausforderungen der Implementierung von LCNC-Lösungen in Unternehmen [15]. Eine häufig behandelte Thematik ist die Senkung der Eintrittsbarrieren für die Citizen Developer, also nicht-technische Benutzer, was die Möglichkeit der Entwicklung von Softwareanwendungen fördert [8].

Zusätzlich wird in der Literatur die Rolle von Schulungen und einer begleitenden Unterstützung bei der Einführung solcher Systeme hervorgehoben, um die Akzeptanz für LCNC-Lösungen bei den Mitarbeitenden zu erhöhen [16]. Studien betonen auch die Notwendigkeit, bestehende Geschäftsprozesse anzupassen, um die vollen Vorteile von LCNC-Plattformen auszuschöpfen [17].

Jedoch gibt es in der bestehenden Literatur eine erkennbare Forschungslücke hinsichtlich der systematischen Identifikation von Erfolgsfaktoren bei der Implementierung von LCNC in produzierenden Unternehmen. Während einige Arbeiten spezifische Bereiche wie Usability oder technische Anforderungen abdecken, fehlt eine umfassende Analyse, die diese Faktoren in einem strukturierten Rahmen zusammenführt und Handlungsempfehlungen für Unternehmen abgibt [14].

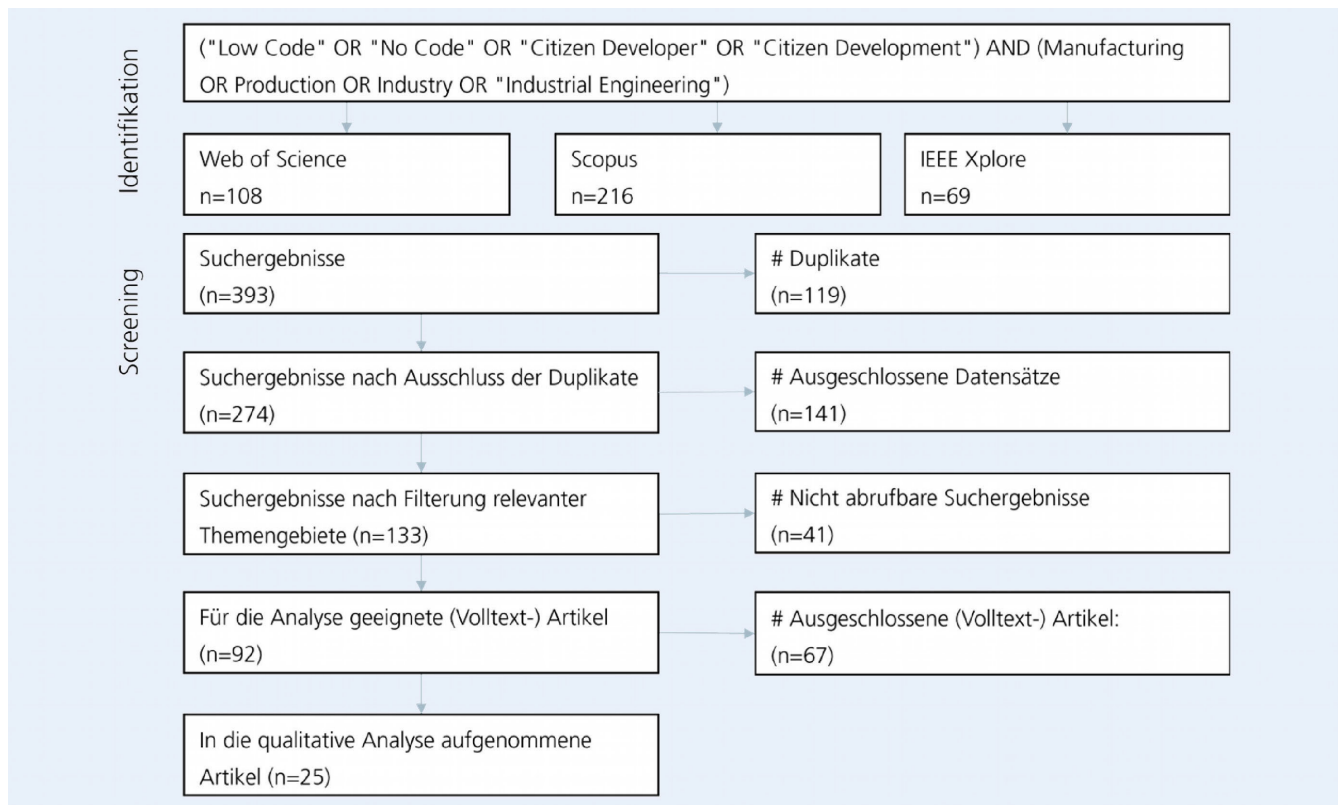


Bild 2. PRISMA-Flow-Diagramm zum Thema LCNC (Low Code/No Code). Grafik: in Anlehnung an [18]

Zusammenfassend ist festzustellen, dass die Forschung zu LCNC-Plattformen zwar bereits wichtige Aspekte behandelt, jedoch ist eine umfassende Analyse der Erfolgsfaktoren und deren praktische Anwendung in der Industrie nach wie vor erforderlich. Diese Erkenntnisse bilden die Grundlage für den im nächsten Kapitel entwickelten Forschungsansatz.

3 Forschungsfrage und Forschungsvorgehen

Für die Identifizierung relevanter Literatur zu den Erfolgsfaktoren und Umsetzungsbarrieren von LCNC-Lösungen wurde ein systematisches Literaturreview nach dem PRISMA-Protokoll durchgeführt. Ziel dieser Analyse war es, sowohl die Anwendung von LCNC-Technologien in industriellen Kontexten als auch deren potenzielle Erfolgsfaktoren und Herausforderungen zu beleuchten [18].

Die Literaturrecherche wurde in den drei Meta-Datenbanken „Web of Science“, „Scopus“ und „IEEE Xplore“ durchgeführt, um eine möglichst umfassende Abdeckung des Themenfeldes zu gewährleisten. Der verwendete Suchstring wurde bewusst breit formuliert (zum Beispiel „Low Code“ OR „No Code“ OR „Citizen Developer“ OR „Citizen Development“) und mit Begriffen aus dem industriellen Umfeld kombiniert („Manufacturing“, „Production“, „Industry“ und „Industrial Engineering“). Diese generische Formulierung des Suchstrings ermöglichte es, ein breites Spektrum an potenziell relevanten Arbeiten zu berücksichtigen. Das ist im Kontext dieser Forschungsarbeit insofern relevant, da sich LCNC-Anwendungen über verschiedene Branchen und Disziplinen hinweg entwickeln und Informationen über konkrete Erfolgsfaktoren zur Implementierung von LCNC-Plattformen oft

aus dem jeweiligen Anwendungskontext erschlossen werden müssen. Nach Anwendung der Kriterien wurden 25 von insgesamt 393 Veröffentlichungen als relevant identifiziert, wie in **Bild 2** dargestellt. Die Untersuchung betrachtet den Zeitraum 2000 bis 09. Februar 2024.

Das Ergebnis der Literaturrecherche ist eine Auflistung der unterschiedlichen Erfolgsfaktoren. Unter dem Begriff „Erfolgsfaktor“ wird in dieser Literaturrecherche folgendes verstanden: Ein Erfolgsfaktor beschreibt eine Bedingung, Eigenschaft oder Ressource, welche erfüllt sein muss oder maßgeblich dazu beiträgt, eine erfolgreiche Einführung und Nutzung der LCNC-Plattformen im Unternehmensbetrieb zu gewährleisten. Diese Erfolgsfaktoren können dabei helfen, Hindernisse in der Einführungsphase der LCNC-Technologie zu überwinden. Die Berücksichtigung der im Rahmen dieses Beitrags identifizierten Erfolgsfaktoren trägt dazu bei, den Implementierungsprozess der Plattformen erfolgreich und zudem insgesamt effizienter und einfacher zu gestalten. Die Vorgehensweise zur Identifizierung dieser Erfolgsfaktoren umfasste die folgenden Schritte, siehe **Bild 3**:

- Schritt 1: Erfassung der Daten: Die relevante Literatur wurde in Hinblick auf theoretische und praktisch erprobte Erfolgsfaktoren zur Implementierung von LCNC-Lösungen analysiert. Dafür wurden entsprechende Textsegmente farblich markiert.
- Schritt 2: Strukturierung der Daten: Die einzelnen Erfolgsfaktoren wurden wörtlich in eine Tabelle übernommen und dabei inhaltlich einer der vier definierten Oberkategorien „Mensch“, „Technik“, „Organisation“ oder „Aufgabe“ zugewiesen.
- Schritt 3: Subkategorisierung am Conceptboard: Für thematisch ähnliche Erfolgsfaktoren wurden entsprechend ihrer in Schritt 2 definierten Oberkategorien nach dem erweiterten

Schritt 1: Farbliche Codierung relevanter Textstellen in der gesichteten Literatur nach dem erweiterten MTO-Modell.



Erfolgsfaktor:	Mensch:	Organisation:	Technik:	Aufgabe
Quelle:				
1				
(...)				
25				

Schritt 2: Zusammenfassung und Formulierung der codierten Textstellen in die einzelnen Erfolgsfaktoren.

Schritt 3: Gruppierung und entsprechende Unterkategorisierung der identifizierten Erfolgsfaktoren am Conceptboard.

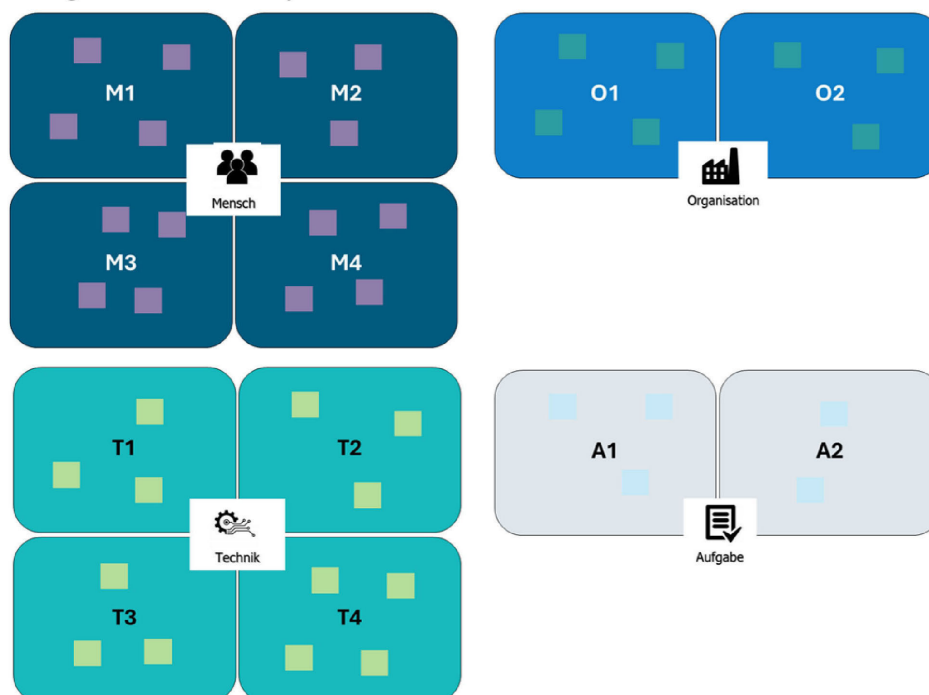


Bild 3. Vorgehensweise zur Ermittlung der Erfolgsfaktoren. Grafik: eigene Darstellung

MTO-Modell weitere Unterkategorien abgeleitet.

Das Conceptboard bot eine visuelle Plattform, um die Beziehungen zwischen den Faktoren zu visualisieren und die Ergebnisse übersichtlich darzustellen. Farbcodierungen und Diagramme halfen, die verschiedenen Kategorien und deren Interaktionen klar zu differenzieren.

Es ist zu erwähnen, dass der Bezug zu den Erfolgsfaktoren von LCNC-Lösungen nicht immer direkt aus der gesichteten Literatur abgeleitet werden konnte. In diesen Fällen wurden mithilfe manueller Ergänzungen verwandte Forschungsergebnisse und Praxisberichte zu ähnlichen digitalen Technologien analysiert und

kontextualisiert. Dieses Vorgehen stellte sicher, dass die theoretischen und praktischen Erfolgsfaktoren trotz der bislang begrenzten Forschung zu LCNC-Implementierungen identifiziert und einbezogen werden konnten.

Insgesamt erlaubte die methodische Herangehensweise eine umfassende und strukturierte Analyse der gesichteten Literatur.

4 Ergebnisse der Untersuchung

In diesem Kapitel werden die identifizierten Erfolgsfaktoren für die Implementierung von LCNC-Lösungen in produzierenden Unternehmen vorgestellt.

4.1 Erfolgsfaktoren bei der Implementierung von LCNC im Bereich Mensch

M1. Erfolgsfaktor: Einfache Erlernbarkeit

Die Fähigkeit, LCNC-Plattformen leicht erlernen und anwenden zu können, ist ein maßgeblicher Erfolgsfaktor für die erfolgreiche Implementierung. Hohe Lernhürden, wie sie etwa bei traditionellen Programmiersprachen bestehen, können durch die Verwendung einer leicht verständlichen Sprache mit einer begrenzten Anzahl an Schlüsselwörtern, Operatoren und Syntaxregeln überwunden werden [19]. Eine benutzerfreundliche Gestaltung und Lesbarkeit, etwa durch Anlehnung an die natürliche Sprache, ermöglichen Citizen Developer die aktive Mitwirkung an der Entwicklung von Anwendungen. Die Möglichkeit mit der eigenen gewohnten Ausdrucksweise zu programmieren, trägt wesentlich zur Akzeptanz der LCNC-Plattform bei [19]. Die Entwicklung von Anwendungen erfolgt meist intern. Die Entwicklung durch eigene Citizen Developer steigert die Vertraulichkeit der Anwendung und wirkt sich damit positiv auf die Implementierung und Motivation der Mitarbeitenden aus [20].

Paradoxerweise begünstigt eine geringe interne IT-Kapazität die Implementierung von LCNC-Plattformen, da IT-ferne Abteilungen verstärkt versuchen, ihre Anwendungen eigenständig mit LCNC-Lösungen zu entwickeln, anstatt auf die bereits ausgelastete IT-Abteilung zu warten [21].

Aufgrund der einfachen Erlernbarkeit von LCNC, kann dem IT-Fachkräftemangel so entgegengewirkt werden [22–25].

M2. Erfolgsfaktor: Schulungen der Mitarbeitenden

Ein wichtiger Erfolgsfaktor für die Implementierung von LCNC-Plattformen ist die einfache und schnelle Schulung der Mitarbeitenden. Schulungen sind besonders effektiv, da LCNC-Lösungen so intuitiv gestaltet sind, dass Fachkräfte aus den jeweiligen Domänen schnell im Umgang mit ihnen geschult werden können. Diese intuitive Bedienbarkeit erlaubt es den Mitarbeitenden, ihre Expertise rasch in den Entwicklungsprozess einzubringen und die Plattformen effizient zu nutzen [26]. In den frühen Phasen der Einführung sind unkomplizierte Schulungen und Mentoring besonders entscheidend, um den Mitarbeitenden das nötige Wissen und die Fähigkeiten zu vermitteln, die für den erfolgreichen Einsatz von LCNC-Plattformen erforderlich sind [19].

M3. Erfolgsfaktor:

Einfache und effiziente Anwendung von LCNC-Lösungen

Eine endnutzerzentrierte Prozessgestaltung ist entscheidend, um intuitive und benutzerfreundliche Anwendungen zu schaffen, da diese die Akzeptanz innerhalb des Unternehmens und damit die Erfolgchancen der Implementierung erhöhen [27]. Auch die Aussicht auf einen signifikanten Effizienzgewinn kann für viele Unternehmen motivierend wirken und somit die Chancen einer erfolgreichen Einführung von LCNC-Technologien steigern [28].

Des Weiteren erhöht sich die Produktivität der Entwickler in dynamischen und interaktiven Programmierumgebungen (auch bekannt unter dem Begriff „Live Programming“), wodurch sich

die Wahrscheinlichkeit einer erfolgreichen Implementierung erhöht [19]. Das Programm sollte außerdem jederzeit ablaufbereit erscheinen, da sich die Entwickler ohne IT-Kenntnisse beispielsweise nicht immer über Zwischenschritte wie etwa Kompilierungsprozesse bewusst sind [22].

M4. Erfolgsfaktor: Schaffung der richtigen Voraussetzungen

Um die Implementierung von LCNC-Plattformen in Unternehmen erfolgreich zu gestalten, ist es entscheidend, die richtigen Voraussetzungen zu schaffen und den Nutzern geeignete Werkzeuge an die Hand zu geben, um ihre Effizienz und Effektivität im Umgang mit den LCNC-Plattformen zu steigern [29]. Dies umfasst zum Beispiel die Ausstattung der Nutzer mit Modeling Constructs, um es ihnen zu ermöglichen, die Ziele eines Workflows auf höherem Abstraktionsniveau zu spezifizieren [30].

Zudem sollten Mitarbeitende aus technischen Bereichen mit datengestützten Lösungen und den passenden Tools ausgestattet werden, um fundierte Entscheidungen treffen zu können [29]. Diese Maßnahmen tragen dazu bei, eine solide Grundlage für die erfolgreiche Implementierung von LCNC-Plattformen zu schaffen [29, 31].

4.2 Erfolgsfaktoren bei der Implementierung von LCNC im Bereich Technologie

T1. Erfolgsfaktor: Einfaches Programmieren (Entwicklung)

Technologische Aspekte sind entscheidend für den Erfolg und die Akzeptanz von LCNC-Plattformen in Unternehmen. Ein wesentlicher Faktor ist die Ausdrucksstärke der Plattformen, die Entwicklern ermöglicht, komplexe Ideen durch bereitgestellte Funktionen umzusetzen [19]. LCNC-Plattformen bieten flexible Anpassungsmöglichkeiten, um Lösungen auf spezifische Unternehmensanforderungen zuzuschneiden, was in der Vergangenheit oft umfangreiche Programmierkenntnisse und lange Entwicklungszeiten erforderte [29]. Ein gut definiertes Technologie-Metamodell bildet das Fundament für die Entwicklung und ermöglicht es auch Citizen Developern, Anwendungen zu erstellen [32]. Im traditionellen Programmieransatz sind Entwickler häufig isoliert und müssen manuell aufwendige Codes schreiben. Im Gegensatz dazu bieten LCNC-Plattformen Nachvollziehbarkeit und automatisierte Testprozesse, die die Qualität der Anwendungen verbessern [33]. Die Bereitstellung teil- und wiederverwendbarer Bibliotheken erhöht die Wahrscheinlichkeit einer erfolgreichen Implementierung und fördert die Zusammenarbeit zwischen Fachanwendern und Entwicklern. Die Wiederverwendbarkeit von Modulen über verschiedene Workflows steigert die Effizienz und beschleunigt die Entwicklung [29].

Automatisierte Datenverarbeitungsprozesse reduzieren den manuellen Aufwand, indem Daten mit minimalem Benutzereingriff verarbeitet werden [34]. Semantische Applikationsvorlagen erleichtern die Prototypenerstellung, während die automatische Codegenerierung den Bedarf an manueller Programmierung verringert und eine intuitive Benutzeroberfläche mit funktionalen Bausteinen es Citizen Developern erlaubt, effektive Anwendungen selbstständig zu entwickeln [35, 36].

Visuelle Schnittstellen und Drag-and-Drop-Funktionen minimieren den Entwicklungsaufwand und steigern die Produktivität. Indem die Programmierkomplexität durch vorgefertigte Komponenten deutlich reduziert wird, können sich Entwickler stärker auf die spezifische Anpassung der LCNC-Anwendung konzentrieren [20, 22]. Die Bereitstellung einer Cloud-Programmierungsfunkti-

on durch LCNC-Plattformen ermöglicht eine verbesserte Teamzusammenarbeit und flexibles Arbeiten, wodurch der Programmieraufwand reduziert wird. Eine benutzerfreundliche Debugging-Umgebung erleichtert die Fehlerbehebung [19]. Insgesamt macht LCNC die Softwareentwicklung zugänglicher, indem es die Barrieren des traditionellen Programmierens senkt [25, 34, 37].

T2. Erfolgsfaktor:

Kompatibilität mit anderen Tools/Technologien

Die Kompatibilität von LCNC-Plattformen mit bestehenden Tools und Technologien ist ein entscheidender technologischer Erfolgsfaktor bei der Implementierung in Unternehmen [25]. Eine hohe Interoperabilität und die Fähigkeit, komplexe Datenformate zu unterstützen, erlauben eine nahtlose Integration in die vorhandene IT-Infrastruktur [21]. Ein erweiterbares Ökosystem, das die Einbindung neuer Tools, Dienste und Plattformen erlaubt, fördert die Flexibilität und Skalierbarkeit der Anwendungen [38].

Ein entscheidender Faktor ist außerdem die Implementierung standardisierter Schnittstellen, um neue Datenquellen bereits während der Entwurfsphase effektiv einbinden zu können [34]. Die Modularität von Sprachen und Modellierungswerkzeugen fördert die Kombination einzelner Bausteine und die Etablierung von Sprachhierarchien. Dies ist vor allem in Anwendungsbereichen wie Smart-Manufacturing relevant, da hier für verschiedene Szenarien unterschiedliche Tiefen von Ereignis- oder Logikmodellen erforderlich sind [39]. Gleichzeitig müssen potenzielle Einschränkungen berücksichtigt werden, wie etwa Plattformabhängigkeit, begrenzte Funktionalität bei der Migration zu anderen Plattformen oder Sicherheitsrisiken [40]. Die Fragmentierung verschiedener Low-Code-Entwicklungsparadigmen nach Anbieter und deren spezifischem Programmiermodell kann zusätzliche Kompatibilitätsprobleme verursachen [20].

Daher ist es essenziell, diese Aspekte bei der Auswahl und Implementierung von LCNC-Plattformen zu beachten, um langfristig eine effiziente, flexible und sichere Entwicklungsumgebung zu gewährleisten [34, 38, 40].

T3. Erfolgsfaktor: Einfache Wartung

Ein anderer zentraler technologischer Erfolgsfaktor bei der Implementierung von LCNC-Plattformen in Unternehmen ist die einfache Wartung [20]. Der Einsatz der von LCNC-Plattformen bereitgestellten Tools zur Code-Wartung und kontinuierlichen Softwareverbesserung vereinfacht die Anpassung bestehender Anwendungen und erlaubt eine schnelle Reaktion auf sich ändernde Geschäftsanforderungen [41]. Zudem zeichnen sich erfolgreich implementierte LCNC-Plattformen oft durch einfache Versionsverwaltungssysteme aus, die eine effiziente Nachverfolgung und Verwaltung von Änderungen durch benutzerfreundliche Kennzeichnungsmethoden erlauben [19].

Ein weiterer Vorteil für eine erfolgreiche Implementierung ist die Integrierbarkeit vorhandener Tools zur Erstellung und Verwaltung standardisierter Ausführungsumgebungen [42]. Durch das sogenannte „Hot Plugging“ können Softwarekomponenten ohne Unterbrechung hinzugefügt oder ausgetauscht werden, was die Flexibilität und Wartungsfreundlichkeit erhöht [19]. Aufgrund der Tatsache, dass weniger Code geschrieben wird, muss auch weniger Code gewartet werden, was den Wartungsaufwand reduziert, die Effizienz steigert und zu niedrigeren IT-Kosten führt [19, 22, 29].

T4. Erfolgsfaktor:

Vereinfachte Analysemethoden von LCNC-Lösungen

Die Vereinfachung von Analysemethoden innerhalb von LCNC-Lösungen ist ebenfalls ein entscheidender technologischer Erfolgsfaktor für deren Implementierung in Unternehmen. LCNC-Plattformen bieten automatisierte Benchmarking-Tools, die Benutzern Leistungskennzahlen über laufende Prozesse und Pipelines zur Verfügung stellen. Diese ermöglichen eine effektive Überwachung und Optimierung der Arbeitsabläufe [42]. Individuell anpassbare Echtzeit-Dashboards erlauben es Unternehmen, wichtige Kennzahlen visuell zu analysieren [43]. Dabei sollten zwei Echtzeit-Betriebsmodi zur Verfügung stehen: einer für die Entwicklungsphase und einer für den regulären Betrieb. Die Modi erlauben eine kontinuierliche Beobachtung sowohl während der Entwicklungsphase als auch im aktiven Betrieb [44].

Im laufenden Betrieb ist eine intuitive Analysierbarkeit von Prozessdaten (Process Data Analytics, PDA), die von der LCNC-Plattform als Analysetools für die Daten der Geschäftsabläufe bereitgestellt wird, von großer Bedeutung [24]. Diese Tools erlauben eine detaillierte Untersuchung und Optimierung von Unternehmensprozessen basierend auf den vorhandenen Daten. Durch den Einsatz von Machine-Learning-Algorithmen und adaptiver Datenanalyse in LCNC-Plattformen können Muster in den Geschäftsprozessdaten identifiziert und prognostische Analysen durchgeführt werden [43]. Die gewonnenen Erkenntnisse tragen dazu bei, fundierte Entscheidungen zu treffen und Unternehmensprozesse dynamisch anzupassen. Insgesamt erleichtern diese technologischen Ansätze die Analyse und Auswertung von Daten innerhalb von LCNC-Lösungen ungemein, was zu effizienteren Arbeitsabläufen und einer gesteigerten Unternehmensleistung führt [43, 44].

4.3 Erfolgsfaktoren bei der Implementierung von LCNC im Bereich Organisation

O1. Erfolgsfaktor: Organisatorische Datenstrategie

LCNC-Plattformen optimieren die Art und Weise, wie Unternehmen ihre Datenstrategien gestalten. Eine effektive organisatorische Datenstrategie muss in der Lage sein, große Datenmengen verfügbar zu machen, vor allem wenn es um „Operational Legacy Data“ geht. Durch den Einsatz von LCNC können Unternehmen die Verfügbarkeit, Nachvollziehbarkeit, Analysierbarkeit und Nutzung von Daten verbessern. Ein entscheidender Faktor von LCNC ist die Möglichkeit zur vorausschauenden Analyse, die es Unternehmen erlaubt, fundierte Entscheidungen zu treffen und zukünftige Trends vorherzusagen. Durch die intuitive Nutzung von LCNC-Plattformen können Citizen Developer komplexe Analysen durchführen, was die Agilität und Reaktionsfähigkeit des Unternehmens erhöht [27].

Ein positiver Faktor für die Implementierung von LCNC ist, dass einige zu entwickelnde Anwendungen bereits in früheren Fallstudien evaluiert wurden. Zudem stehen sie oft durch Open-Source-Quellen kostenlos zur Verfügung und werden bereits in ähnlichen Industriesektoren erfolgreich eingesetzt. Dies verringert das Risiko und erhöht die Erfolgchancen des Rollouts, da bewährte Praktiken sowie branchenspezifische Erfahrungen einfließen können [24].

Insgesamt trägt eine gut durchdachte Datenstrategie unter Verwendung von LCNC-Technologien dazu bei, die Effizienz und Agilität von Unternehmen zu steigern.

O2. Erfolgsfaktor: Effiziente Planung der IT-Kapazitäten

Ressourcensparende Anwendungen sind von besonderer Bedeutung für Organisationen mit begrenzten IT-Kapazitäten. LCNC-Plattformen ermöglichen es, IT-Kosten zu senken, da weniger externe Ressourcen benötigt werden und Fachabteilungen direkt in die Entwicklung einbezogen werden [9, 22].

Durch ressourcensparende Anwendungen können Unternehmen ihre Betriebskosten reduzieren und gleichzeitig die Effizienz erhöhen. Mitarbeitende aus Fachabteilungen können eigenständig Lösungen entwickeln, was die Abhängigkeit von IT-Abteilungen verringert. Dies fördert eine agile Unternehmenskultur, in der Entscheidungen schneller getroffen und umgesetzt werden können [13, 14].

4.4 Erfolgsfaktoren bei der Implementierung von LCNC im Bereich Aufgabe

A1. Erfolgsfaktor: Anpassbare, ready-to-use Komponenten

Anpassbare, sofort einsatzbereite Komponenten sind entscheidend, um Aufgaben im Betriebsalltag effizienter zu bewältigen. Diese Komponenten ermöglichen es den Citizen Developern, schnell maßgeschneiderte Lösungen für alltägliche Herausforderungen zu erstellen, ohne sich mit komplexen Programmierdetails auseinandersetzen zu müssen [22]. LCNC-Plattformen bieten hierbei dem Anwender intuitive, vorgefertigte Bausteine, die an spezifische Aufgaben angepasst werden können. Das erleichtert die direkte Integration in Arbeitsprozesse und reduziert zeitaufwendige Zwischenschritte. Mitarbeitende können sich so auf ihre Kernaufgaben konzentrieren, während administrative oder repetitive Aufgaben durch LCNC-gestützte Automatisierungen vereinfacht werden [19, 20].

A2. Erfolgsfaktor: Sinnvolle Abgrenzung und Dokumentation der Entwicklungsarbeit

Eine sinnvolle Abgrenzung von Entwicklungsaufgaben sowie die anschließende umfassende Dokumentation der jeweiligen Entwicklungsarbeit einzelner Citizen Developer sind essenziell, um die Nachvollziehbarkeit, Wiederverwendbarkeit und effiziente Umsetzung von LCNC-Lösungen im Betriebsalltag sicherzustellen. LCNC-Plattformen unterstützen eine sinnvolle Abgrenzung, indem sie die Aufgabenteilung durch Workflows ermöglichen, die es den Mitarbeitenden gestatten, sich auf individuelle Verantwortungsbereiche zu konzentrieren [19, 25].

Dies erlaubt eine schrittweise und effiziente Umsetzung von Projekten, ohne wichtige Aufgaben zu vernachlässigen, da jeder Mitarbeitende im Team weiß, welche Aufgaben im Rahmen des Betriebsalltags seiner Verantwortung unterliegen. Dies reduziert Überforderung und fördert die Zusammenarbeit im täglichen Arbeitsprozess. Eine umfassende Dokumentation dieser Entwicklungsarbeit ist entscheidend für die erfolgreiche Implementierung von LCNC-Plattformen [19].

Automatische Dokumentationsfunktionen können diese Prozesse unterstützen, indem sie die Nachvollziehbarkeit und Wiederverwendbarkeit der entwickelten Lösungen gewährleisten, da die strukturierte Erfassung von Arbeitsergebnissen und Prozessen es erleichtert, erfolgreiche Ansätze für ähnliche Aufgaben erneut zu nutzen oder weiterzuentwickeln [24]. Eine klare Dokumentation reduziert die Wahrscheinlichkeit von Fehlern, erleichtert die Einarbeitung neuer Mitarbeitende und ermöglicht, dass Lösungen reibungslos in die alltäglichen Arbeitsabläufe integriert bleiben [19].

5 Limitationen

Eine potenzielle Einschränkung dieser Arbeit besteht in der durchgeführten Literaturrecherche, da nicht garantiert werden kann, dass sämtliche relevanten Publikationen vollständig erfasst wurden. Obwohl der systematische Literaturreview gemäß den Richtlinien des PRISMA-Protokolls als methodisch fundierter Ansatz gilt, können dennoch einzelne Quellen durch das Auswahlraster fallen.

Zusätzlich kann die Zuordnung der Erfolgsfaktoren zu den Kategorien Mensch, Technologie, Organisation und Aufgabe Überschneidungen aufweisen, was auf die inhaltliche Komplexität der Themen zurückzuführen ist. Diese Herausforderung wurde aber durch eine qualitativ bewertete Schwerpunktsetzung weitestgehend adressiert.

In diesem Beitrag wurden keine expliziten Untersuchungen der Hindernisse, Nachteile und Folgen der Implementierung von LCNC-Lösungen angestellt. Eine eingehende Auseinandersetzung mit diesen Aspekten wäre jedoch ein wichtiger Baustein zur Entwicklung einer umfassenden Implementierungsroadmap für LCNC-Lösungen in Unternehmen.

6 Ausblick und Fazit

Der vorliegende Beitrag kategorisiert, welche Faktoren zu einer erfolgreichen LCNC-Plattform-Implementierung beitragen können. Basierend auf diesen Erkenntnissen ergeben sich vielversprechende Ansätze für zukünftige Untersuchungen und Weiterentwicklungen. So könnte zum Beispiel die Relevanz unterschiedlicher Erfolgsfaktoren über den Anwendungs- oder Implementierungsprozess hinweg (also vor, während oder nach der Implementierung) untersucht werden.

Eine weitere Möglichkeit zur Weiterentwicklung bietet die Ausarbeitung eines detaillierten Leitfadens für Unternehmen, der diese bei der Umsetzung der Implementierung begleitet und eingehend unterstützt. Ein solcher Leitfaden könnte Orientierungshilfe in Form von konkreten Handlungsempfehlungen liefern, um die praktische Umsetzung relevanter Erfolgsfaktoren im eigenen Unternehmensalltag zu erleichtern. Auch die Validierung der gewonnenen Forschungsergebnisse in einem Unternehmen bietet Potenzial für weiterführende Arbeiten. Diese Forschungsansätze weiter zu vertiefen könnte neue Perspektiven für die erfolgreiche Implementierung von LCNC-Plattformen eröffnen. Zumal hierbei auch vielversprechende technologische Fortschritte wie in der generativen KI eine immer größere Rolle spielen könnten.

Die Untersuchung bietet eine Übersicht über den Erkenntnisstand im dynamischen Forschungsfeld der LCNC-Lösungen. Durch die systematische Analyse und Strukturierung der Erfolgsfaktoren entlang der Dimensionen Mensch, Technologie, Organisation und Aufgabe werden Grundlagen für weiterführende wissenschaftliche Arbeiten geschaffen.

Unternehmen können aus den identifizierten Erfolgsfaktoren eigene Handlungsempfehlungen zur erfolgreichen Implementierung von LCNC-Plattformen ableiten. Aus diesen Erfolgsfaktoren geht hervor, wie LCNC-Plattformen effektiv eingeführt und genutzt werden können, um betriebliche Herausforderungen wie den IT-Fachkräftemangel zu bewältigen. Damit liefert die Studie eine Orientierungshilfe für die praktische Anwendung von LCNC-Technologien im Unternehmenskontext.

Literatur

- [1] Bitkom Research: Rekord-Fachkräftemangel: In Deutschland sind 149.000 IT-Jobs unbesetzt. Stand: 2023. Internet: www.bitkom.org/Presse/Presseinformation/Rekord-Fachkraeftemangel-Deutschland-IT-Jobs-unbesetzt/. Zugriff am 14.03.2024
- [2] Mueller, K.: Komplexität beherrschen: Modellbasierte Softwareentwicklung von Enterprise Software. mgm technology partners GmbH. Stand: 2023. Internet: insights.mgm-tp.com/de/komplexitaet-beherrschen-modellbasierte-softwareentwicklung-von-enterprise-software/. Zugriff am 14.03.2024
- [3] IBM: Was ist Low-Code? Stand: 2024. Internet: www.ibm.com/de-de/topics/low-code. Zugriff am 14.03.2024
- [4] Adrian, B.; Hinrichsen, S.; Schulz, A. et al.: Low-Code-Programmierung als Ansatz zur Gestaltung bedarfsgerechter informatorischer Assistenzsysteme – eine Fallstudie. In: Bornewasser, M.; Hinrichsen, S. (Hrsg.): Informatorische Assistenzsysteme in der variantenreichen Montage. Heidelberg: Springer Vieweg, 2020, S. 173–186
- [5] Sauer, S.; Weidmann, N.; Kirchhoff, J.: Merkmale und Entwicklungslinien der Low-Code-Programmierung. In: Hinrichsen, S.; Sauer, S.; Schröder, K. (Hrsg.): Prozesse in Industriebetrieben mittels Low-Code-Software digitalisieren. Heidelberg: Springer Vieweg 2023, S. 17–29
- [6] Vial, G.: Understanding digital transformation: A review and a research agenda. *The Journal of Strategic Information Systems* 28 (2019) 2, pp. 118–144
- [7] Kagermann, H.; Wahlster, W.; Helbig, J.: Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0. Stand: 2013. Internet: www.acatech.de/publikation/umsetzungsempfehlungen-fuer-das-zukunftsprojekt-industrie-4-0-abschlussbericht-des-arbeitskreises-industrie-4-0/download-pdf?lang=de. Zugriff am 14.03.2024
- [8] Gartner: Forecast Analysis: Low-Code Development Technologies. Stand: 2021. Internet: www.gartner.com/en/documents/3995846. Zugriff am 14.03.2024
- [9] Fuerst, S.; Sanchez-Dominguez, O.; Rodriguez-Montes, M. A.: The Role of Digital Technology within the Business Model of Sustainable Entrepreneurship. *Sustainability* 15 (2023) 14, #10923
- [10] Ulich, E.: Arbeitssysteme als Soziotechnische Systeme – eine Erinnerung. *Journal Psychologie des Alltagshandelns* 6 (2013) 1, S. 4–12
- [11] Hirsch-Kreinsen, H.; Ittermann, P.: Digitalization of Work Processes: A Framework for Human-Oriented Work Design. In: McMurray, A.; Muenjohn, N.; Weerakoon, C. (Hrsg.): *The Palgrave Handbook of Workplace Innovation*. Cham: Palgrave Macmillan, 2021, pp. 273–293
- [12] Hanelt, A.; Bohnsack, R.; Marz, D. et al.: A Systematic Review of the Literature on Digital Transformation: Insights and Implications for Strategy and Organizational Change. *Journal of Management Studies* 58 (2021) 5, pp. 1159–1197
- [13] Bresciani, S.; Ferraris, A.; Romano, M. et al.: Digital Transformation Management for Agile Organizations: A Compass to Sail the Digital World. Leeds: Emerald Publishing Limited 2021
- [14] Kraus, S.; Ferraris, A.; Bertello, A.: The future of work: How innovation and digitalization re-shape the workplace. *Journal of Innovation & Knowledge* 8 (2023) 4, #100438
- [15] Myers; Klein: A Set of Principles for Conducting Critical Research in Information Systems. *MIS Quarterly* 35 (2011) 1, pp. 17–36
- [16] Ullrich, A.; Reißig, M.; Niehoff, S. et al.: Employee involvement and participation in digital transformation: a combined analysis of literature and practitioners' expertise. *Journal of Organizational Change Management* 36 (2023) 8, pp. 29–48
- [17] Calderon-Monge, E.; Ribeiro-Soriano, D.: The role of digitalization in business and management: a systematic literature review. *Review of Managerial Science* 18 (2024) 2, pp. 449–491
- [18] PRISMA: PRISMA statement. Stand: 2024. Internet: www.prisma-statement.org/. Zugriff am 14.03.2024
- [19] Avishahar-Zeira, A.; Lorenz, D. H.: Could No-Code Be Code? Toward a No-Code Programming Language for Citizen Developers. Onward! 2023 ACM SIGPLAN International Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software, Cascais, Portugal, 2023, pp. 103–119
- [20] Sanchis, R.; Garcia-Perales, Ó.; Fraile, F. et al.: Low-Code as Enabler of Digital Transformation in Manufacturing Industry. *Applied Sciences* 10 (2020) 1, #12
- [21] Guevara, I.; Chaudhary, H. A. A.; Margaria, T.: Model-Driven Edge Analytics: Practical Use Cases in Smart Manufacturing. *International Symposium on Leveraging Applications of Formal Methods*, 2022, pp. 406–421
- [22] Gomes, P. M.; Brito, M. A.: Low-Code Development Platforms: A Descriptive Study. 17th Iberian Conference on Information Systems and Technologies (CISTI), Madrid, Spain, 2022, pp. 1–4
- [23] Sahay, A.; Di Ruscio, D.; Pierantonio, A.: Understanding the role of model transformation compositions in low-code development platforms. *MODELS '20: ACM/IEEE 23rd International Conference on Model Driven Engineering Languages and Systems*, Virtual Event Canada, 2020, pp. 1–5
- [24] Schneid, K.; Stapper, L.; Thone, S. et al.: Automated Regression Tests: A No-Code Approach for BPMN-based Process-Driven Applications. 2021 IEEE 25th International Enterprise Distributed Object Computing Conference (EDOC), Gold Coast, Australia, 2021, pp. 31–40
- [25] Käss, S.; Strahringer, S.; Westner, M.: A Multiple Mini Case Study on the Adoption of Low Code Development Platforms in Work Systems. *IEEE Access* 11 (2023), pp. 118762–118786
- [26] Wang, Y.; Feng, Y.; Zhang, M. et al.: The Necessity of Low-code Engineering for Industrial Software Development: A Case Study and Reflections. 2021 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW), Wuhan, China, 2021, pp. 415–420
- [27] Redchuk, A.; Walas Mateo, F.: New Business Models on Artificial Intelligence—The Case of the Optimization of a Blast Furnace in the Steel Industry by a Machine Learning Solution. *Applied System Innovation* 5 (2022) 1, p. 6
- [28] Lin, C.; Jia, C.; Han, G. et al.: A Low-Code Edge Computing-Based Predicting Scheme for Aeroengine Components to Enable Intelligent Aeronautical Manufacture. *IEEE Journal of Emerging and Selected Topics in Industrial Electronics* 5 (2024) 2, pp. 745–752
- [29] El Kamouchi, H.; Kissi, M.; El Beggar, O.: Low-code/No-code Development: A systematic literature review. 14th International Conference on Intelligent Systems: Theories and Applications (SITA), Casablanca, Morocco, 2023, pp. 1–8
- [30] Redchuk, A.; Walas Mateo, F.; Pascal, G. et al.: Adoption Case of IIoT and Machine Learning to Improve Energy Consumption at a Process Manufacturing Firm, under Industry 5.0 Model. *Big Data and Cognitive Computing* 7 (2023) 1, #42
- [31] Horváth, B.; Horváth, Á.; Wimmer, M.: Towards the next generation of reactive model transformations on low-code platforms. *MODELS '20: ACM/IEEE 23rd International Conference on Model Driven Engineering Languages and Systems*, Virtual Event Canada, 2020, pp. 1–10
- [32] Bexiga, M.; Garbatov, S.; Seco, J. C.: Closing the gap between designers and developers in a low code ecosystem. *MODELS '20: ACM/IEEE 23rd International Conference on Model Driven Engineering Languages and Systems*, Virtual Event Canada, 2020, pp. 1–10
- [33] Qu, D.; Zhang, Y.; Hu, X. et al.: Contract-Based Design for Low-Code Development in Industrial Edge Applications. 2023 IEEE 32nd International Symposium on Industrial Electronics (ISIE), Helsinki, Finland, 2023, pp. 1–6
- [34] Das, P.; Srivastava, S.; Moskovich, V. et al.: CDI-E. *Proceedings of the VLDB Endowment* 15 (2022) 12, pp. 3319–3331
- [35] Ren, H.; Dorofeev, K.; Anicic, D. et al.: SeLoC-ML: Semantic Low-Code Engineering for Machine Learning Applications in Industrial IIoT. In: Sattler, U.; Hogan, A.; Keet, M. et al. (Hrsg.): *The Semantic Web – ISWC 2022*. Cham: Springer International Publishing 2022, pp. 845–862
- [36] Halim, J.; Eichler, P.; Krusche, S. et al.: No-code robotic programming for agile production: A new markerless-approach for multimodal natural interaction in a human-robot collaboration context. *Frontiers in robotics and AI* 9 (2022), #1001955
- [37] Waszkowski, R.: Low-code platform for automating business processes in manufacturing. *IFAC-PapersOnLine* 52 (2019) 10, pp. 376–381
- [38] Deshmukh, R. A.; Jayakody, D.; Schneider, A. et al.: Data Spine: A Federated Interoperability Enabler for Heterogeneous IIoT Platform Ecosystems. *Sensors (Basel, Switzerland)* 21 (2021) 12, #4010
- [39] Dalibor, M.; Heithoff, M.; Michael, J. et al.: Generating customized low-code development platforms for digital twins. *Journal of Computer Languages* 70 (2022), #101117
- [40] Chaudhary, H. A. A.; Guevara, I.; Singh, A. et al.: Efficient Model-Driven Prototyping for Edge Analytics. *Electronics* 12 (2023) 18, #3881
- [41] Overeem, M.; Jansen, S.: Proposing a Framework for Impact Analysis for Low-Code Development Platforms. 2021 ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C), Fukuoka, Japan, 2021, pp. 88–97
- [42] Klarhorst, C.; Quirin, D.; Hesse, M. et al.: ML4ProFlow: A Framework for Low-Code Data Processing from Edge to Cloud in Industrial Production. 2022 IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA), Stuttgart, 2022, pp. 1–4
- [43] Phalake, V. S.; Joshi, S. D.; Rade, K. A. et al.: Optimization for achieving sustainability in low code development platform. *International Journal on Interactive Design and Manufacturing (IJDeM)* 18 (2024) 8, pp. 5717–5724
- [44] Bialou-Sama, A.: Simplify model design in MDE approaches. *MODELS '22: Proceedings of the 25th International Conference on Model Driven Engineering Languages and Systems: Companion* (2022), pp. 170 – 175



Jan Philipp Pydde, M.Sc. 
jan.philipp.pydde@ipa.fraunhofer.de
Tel. +49 711 / 970-1441
Foto: Fraunhofer IPA



Robin Dominic Ziegler, M.Eng. 
Foto: Fraunhofer IPA

Arthur Mezger, B.Sc.
Fraunhofer-Institut für Produktionstechnik
und Automatisierung IPA
Nobelstr. 12, 70569 Stuttgart
www.ipa.fraunhofer.de

LIZENZ



Dieser Fachaufsatz steht unter der Lizenz Creative Commons
Namensnennung 4.0 International (CC BY 4.0)