HMD Praxis der Wirtschaftsinformatik
https://doi.org/10.1365/s40702-024-01094-z

SCHWERPUNKT

Business Unit Development: Vorteile und Herausforderungen für beteiligte Mitarbeiter

Dominik Scharpf · Altus Viljoen · Andreas Hein · Helmut Krcmar

Eingegangen: 15. Februar 2024 / Angenommen: 19. Juni 2024
© The Author(s) 2024

Zusammenfassung Der anhaltende Mangel an Fachkräften in der IT-Branche stellt eine zentrale Herausforderung für Unternehmen dar und erschwert den Fortschritt bei der digitalen Transformation. Infolgedessen gewinnt der Einsatz von Low-Code/No-Code (LCNC) sowohl in der Praxis als auch in der Wissenschaft zunehmend an Interesse, um auch Mitarbeiter außerhalb der IT-Abteilung dazu zu befähigen, als sog. Business Unit Developer eigenständig Digitalisierungs- und Automatisierungslösungen zu entwickeln. Bisher konzentriert sich die wissenschaftliche Literatur auf die technischen Aspekte von LCNC, während Fragen zur optimalen Einführung in Unternehmen unbeantwortet bleiben. Diese Forschungsarbeit untersucht die praktische Implementierung von Business Unit Development Programmen in Unternehmen, um Auswirkungen auf beteiligte Mitarbeitende zu identifizieren. In Kooperation mit den Stadtwerken München (SWM) wurde eine umfassende Einzelfallstudie durchgeführt, bei der die Umsetzung einer Business Unit Development Initiative untersucht wurde. Die Ergebnisse zeigen unter anderem, wie das Business Unit Development von teilnehmenden Mitarbeitern wahrgenommen wird sowie welche individuellen Vorteile sich für diese ergeben. Zudem wird ersichtlich, mit welchen Herausforderungen Mitarbeitende im Rahmen des Business Unit Developments konfrontiert sind und wie sich die Entwicklung und der Einsatz von LCNC-Anwendungen auf

✉ Dominik Scharpf · Altus Viljoen · Helmut Krcmar
Technische Universität München, München, Deutschland
E-Mail: dominik.scharpf@tum.de

Altus Viljoen
E-Mail: altus.viljoen@tum.de

Helmut Krcmar
E-Mail: helmut.krcmar@tum.de

Andreas Hein
Universität St. Gallen, St. Gallen, Schweiz
E-Mail: andreas.hein@unisg.ch

Published online: 04 July 2024

deren Arbeitsalltag auswirkt. Die Ergebnisse tragen zu einem besseren Verständnis über den praktischen Einsatz von LCNC in Unternehmen bei und zeigen neue Forschungsfelder auf. Zudem können die Ergebnisse Unternehmen dabei unterstützen, organisatorische und strategische Entscheidungen im Rahmen des Aufbaus und der Skalierung einer Business Unit Development Initiative zu treffen.

Schlüsselwörter Business Unit Development · Citizen Development · Low-Code-Development-Platform · No-Code

Business Unit Development: Benefits and Challenges for Employees

Abstract The ongoing shortage of skilled workers in the IT sector is a key challenge for companies and is hampering progress in the digital transformation. As a result, the use of low-code/no-code (LCNC) is gaining increasing interest both in practice and in science to enable non-IT-employees to independently develop digitization and automation solutions as so-called business unit developers. So far, the scientific literature has focused on the technical aspects of LCNC, while questions about the optimal introduction in companies remain unanswered. This research examines the practical implementation of business unit development programs in companies to identify the effects on participating employees. In cooperation with Stadtwerke München (SWM) a comprehensive case study was conducted to investigate the implementation of a business unit development initiative. The results show how business unit development is perceived by participating employees and what individual benefits arise for them. Furthermore, they reveal challenges employees face in the context of business unit development and how the development and use of LCNC applications affects their day-to-day work. The results contribute to a better understanding of the practical use of LCNC in companies and point to new fields of research. In addition, the results can support companies in making organizational and strategic decisions in the context of setting up and scaling a business unit development initiative.

Keywords Business Unit Development · Citizen Development · Low-Code-Development-Platform · No-Code

1 Einleitung

Die digitale Transformation ist ein komplexer Prozess, der Unternehmen nicht nur dabei hilft, ihre Effizienz und Ressourcennutzung zu steigern, sondern auch durch zunehmende Automatisierung und die Erschließung neuer Geschäftsfelder und Märkte erhebliche Potenziale eröffnet (Nambisan et al. 2019; Sebastian et al. 2017). Diese Entwicklung ist essenziell, um die Wettbewerbsfähigkeit von Unternehmen in einem schnell wandelnden Marktumfeld zu sichern (Schlaepfer und von Radowitz 2017). Gleichzeitig führt die digitale Transformation zu einem stetig wachsenden Bedarf an IT-Ressourcen, einer Tendenz, die sich insbesondere während der Covid-

19-Pandemie verstärkt hat. Ein signifikantes Hindernis in diesem Kontext ist der Mangel an qualifizierten IT-Fachkräften, der sich zunehmend als zentrales Problem für deutsche Unternehmen herausstellt. Laut einer aktuellen Umfrage des Digitalverbands Bitkom beläuft sich die Lücke auf etwa 137.000 fehlende Fachkräfte, das sich als das „Haupthindernis bei der digitalen Transformation" manifestiert (Bitkom Reserarch 2022).

Eine mögliche Gegenmaßnahme gegen den Fachkräftemangel in der IT-Branche stellt das Konzept des Business Unit Developments dar. Hierbei werden auch Mitarbeiter außerhalb der IT-Abteilung mittels einer Low-Code Development Plattform (LCDP) ermächtigt, eigenständig Software-Lösungen zu entwickeln. Sie können so als sogenannte Business Unit Developer (BUD)[1] an der digitalen Transformation im Unternehmen mitwirken (Carroll et al. 2021). Damit birgt das Business Unit Development das Potenzial, die IT-Abteilung im Unternehmen zu entlasten (Alsaadi et al. 2021) sowie das Innovationspotenzial durch Eigenentwicklungen in den Fachbereichen zu stärken (Krejci et al. 2021). Basierend auf diesen Vorteilen integrieren immer mehr Unternehmen, wie beispielsweise die Deutsche Bahn (Kolarsch et al. 2021), Shell (Carroll und Maher 2023) oder auch Volvo (Novales und Mancha 2023) das Konzept des Business Unit Development in ihre IT-Strategie.

Auch in der wissenschaftlichen Forschung zeichnet sich ein stark wachsendes Interesse an den Themenbereichen Low-Code und Business Unit Development ab (Naqvi et al. 2024; Prinz et al. 2021). Dabei konzentrieren sich die meisten aktuellen Veröffentlichungen jedoch vorwiegend auf die technischen Aspekte der LCDPs und analysieren dabei unter anderem die technischen Merkmale von LCDPs (Bock und Frank 2021), vergleichen verschiedene Low-Code Plattformen miteinander (Frank et al. 2021; Sahay et al. 2020; Sanchis et al. 2019), beleuchten die Gemeinsamkeiten und Unterschiede zu ähnlichen Technologien wie dem Model-Driven-Development (Di Ruscio et al. 2022) oder behandeln den Einsatz von Low-Code in anderen Bereichen der Informatik, bspw. im Bereich des Machine Learning (Maiya 2022; Sundberg und Holmström 2023). Wie jedoch Low-Code im Rahmen von Business Unit Development in Unternehmen eingeführt werden kann, gilt als offener Forschungsbereich (Käss et al. 2022; Prinz et al. 2021; Viljoen et al. 2023). Ein zentraler Bestandteil ist dabei die individuelle Wahrnehmung und Einstellung der Mitarbeitenden zu ihrer Rolle als Business Unit Developer (Binzer und Winkler 2022). Darauf aufbauend wurden drei Forschungsfragen (FF) formuliert:

FF1 Welche Vorteile ergeben sich für Mitarbeiter durch das Business Unit Development?

FF2 Mit welchen Herausforderungen sind Mitarbeiter durch das Business Unit Development konfrontiert?

FF3 Welche Maßnahmen sind zur Realisierung der Vorteile aus dem Business Unit Development notwendig?

[1] Für die Bezeichnung des Anwenders der Low-Code Platform wird unter anderem auch der Begriff „Citizen Developer" verwendet. Dieser Artikel nutzt ausschließlich den Begriff „Business Unit Developer".

Zur Beantwortung dieser Forschungsfragen wurde in Kooperation mit den Stadtwerken München (SWM) ein Forschungsprojekt im Rahmen einer umfassenden Einzelfallstudie zu deren Business Unit Development Initiative durchgeführt (Yin 2018). Dabei wurden 12 Business Unit Developer sowie vier IT-Führungskräfte befragt und die Transkripte anschließend systematisch ausgewertet (Strauss und Corbin 1990). Gegenstand der Befragung waren die individuelle Wahrnehmung des Business Unit Developments sowie die im Unternehmen dafür geschaffenen organisatorischen Strukturen. Dieser Forschungsbeitrag beschränkt sich dabei auf die individuellen Aspekte des Business Unit Developments.

Die Ergebnisse der Arbeit beleuchten, wie Business Unit Development Programme von beteiligten Mitarbeitern in der Praxis wahrgenommen werden und zeigen auf, wie sich die Entwicklung und der Einsatz von Low-Code-Anwendungen sowohl unmittelbar auf den Arbeitsalltag als auch langfristig auf die berufliche Umgebung des Mitarbeitenden auswirken. Zugleich geben die Ergebnisse Einblicke in technische und soziale Herausforderungen, mit denen Mitarbeitende in ihrer zusätzlichen Rolle als Business Unit Developer konfrontiert sind. Damit ergänzen die Ergebnisse die bisher stark technisch ausgerichtete Forschung um die individuelle Perspektive auf die Low-Code-Entwicklung aus der Sicht von aktiven Business Unit Developern und unterstützen Unternehmen zudem dabei, Low-Code-Initiativen zielgerichtet und nachhaltig zu skalieren.

2 Theoretischer Hintergrund

Der Begriff „Business Unit Developer“ bezeichnet einen Mitarbeiter außerhalb der IT-Abteilung, der mithilfe von im Unternehmenskontext zugänglichen Tools aktiv Softwareanwendungen entwickelt (Gartner 2022; Li et al. 2022). Dabei besitzen Business Unit Developer keine formale Informatikausbildung (Hintsch et al. 2021). Aufgrund ihres ausgeprägten Wissens über die Anforderungen und Abläufe im eigenen Fachbereich können diese jedoch einfacher Automatisierungspotenziale erkennen (Steele 2021) und Anwendungen entwickeln, die spezifischer auf die Bedürfnisse ihres Fachbereichs zugeschnitten sind (Elshan et al. 2023b).

Low-Code-Entwicklungs-Plattformen bilden dabei zumeist die technologische Basis für das Business Unit Development, indem sie Entwicklungsumgebungen bereitstellen, mit denen sich Anwendungen über eine graphische Benutzeroberfläche entwickeln lassen. Der Begriff „Low-Code“ bezieht sich darauf, dass im Vergleich zu traditionellen Programmiersprachen wie Java, Python oder C++ weniger textbasierter Code benötigt wird (Richardson und Rymer 2014). Dadurch wird die Programmierung von Benutzeroberflächen und Programm-Logik sowie die Einbindung von Datenquellen stark abstrahiert, was den Bedarf an textbasiertem Quellcode minimiert (Vincent et al. 2019). Dies reduziert die Komplexität der Anwendungsentwicklung und ermöglicht es damit auch Endanwendern ohne Programmiererfahrung, Software zu entwickeln und zu nutzen (Bock und Frank 2021).

Das grundsätzliche Konzept hinter Business Unit Development, Endanwender zur Entwicklung von Software zu ermächtigen, besteht in der Informatik bereits länger und wird unter anderem im Forschungsbereich des „End-User Development“

(EUD) behandelt (Barricelli et al. 2019). Insbesondere der anhaltenden IT-Fachkräftemangel sowie der zugleich steigende Bedarf an digitalen Lösungen zeigen sich als zentrale Faktoren dafür, dass diese Thematik in letzter Zeit erneut an Bedeutung gewinnt (Käss et al. 2022; Klotz et al. 2022; Naqvi et al. 2024). Darüber hinaus spielt aus technischer Sicht auch die cloudbasierte Bereitstellung der Low-Code-Tools eine entscheidende Rolle, da diese die Zugänglichkeit der Entwicklungsumgebung sowie die Produktivsetzung von Anwendungen weiter vereinfacht (Di Ruscio et al. 2022). Unternehmen setzen daher vermehrt auf den Einsatz von LCDPs, um die Entwicklung von Softwareanwendungen auch Nicht-IT-Mitarbeitern zu ermöglichen und damit die begrenzten internen Ressourcen der IT zu entlasten (Alsaadi et al. 2021). Weiterhin bieten auch die Digitalisierung und Modernisierung von internen Prozessen (Elshan et al. 2023a), geringere Entwicklungskosten (Martinez und Pfister 2023), eine verkürzte Time-to-Market (Käss et al. 2022) und die Reduktion der Schatten-IT im Unternehmen (Breidenbach et al. 2023) zusätzliche Anreize für Unternehmen zur Adoption von Low-Code.

In Hinblick auf die Implementierung von Business Unit Development zeigen sich in aktuellen Veröffentlichungen bereits verschiedene Rahmenbedingungen, die die Adoption des Business Unit Development im Unternehmen fördern können. Diese umfassen beispielsweise die Einführung von Governance-Strukturen (Hoogsteen und Borgman 2022), den Aufbau einer unternehmensinternen Community zur Vernetzung und Teilen von Erfolgen (Carroll und Maher 2023) oder auch die Etablierung eines Low-Code Teams mit professionellen Entwicklern zur Ausbildung und Unterstützung der Mitarbeitenden (Novales und Mancha 2023). Jedoch gibt es zurzeit noch kaum Wissen darüber, wie Business Unit Development Programme von teilnehmenden Mitarbeitern aufgenommen werden, sowie, welche Vorteile und Herausforderungen sich durch den Einsatz von Low-Code im Arbeitsalltag für Mitarbeitende ergeben (Binzer und Winkler 2022). Zusätzliche Erkenntnisse hinsichtlich dieser Fragestellungen ermöglichen Einblicke, in welchem Maß Business Unit Development in der Praxis tatsächlich angenommen wird.

3 Methodik

Für diese Arbeit wurde eine Einzelfallstudie nach Yin (2018) als Forschungsmethode ausgewählt. Dabei konnten die Stadtwerke München als Kooperationspartner gewonnen werden. Die SWM versorgen die Stadt München mit Strom, Erdgas, Fernwärme, Fernkälte und Wasser und betreiben zudem zentrale öffentliche Nahverkehrsmittel, Schwimmbäder sowie Glasfaser- und Kommunikationsnetze (SWM 2024). Mit über 11.000 Mitarbeitern sind die SWM eines der größten kommunalen Unternehmen Deutschlands. Die SWM zeigen sich aus mehreren Aspekten als geeigneter Fallpartner für diese Studie. Zunächst verfügen die SWM über mehrjährige Erfahrung in der aktiven Nutzung von zwei führenden Low-Code-Plattformen im Unternehmen. Dies umfasst sowohl den Einsatz der Plattform OutSystems zur professionellen Anwendungsentwicklung innerhalb der IT als auch die Verwendung der Microsoft Power Platform im Rahmen eines unternehmensinternen Business Unit Development Programms, wodurch ein breites Spektrum an Low-Code-Expertise

vorhanden ist. Dabei weist insbesondere das Business Unit Development Programm eine hohe Dynamik auf und zählt zum Zeitpunkt der Studie über 500 registrierte Business Unit Developer. Zudem ist das Business Unit Development für alle Mitarbeiter im Unternehmen zugänglich, wodurch sich die Anwendungsfälle vielseitig über alle Fach- und Anwendungsbereiche erstrecken. Diese reichen beispielsweise von der Automatisierung von mehrstufigen Genehmigungsprozessen über digitale Wartungsprotokolle und Antragsformulare bis hin zu Anwendungen zur Raumbuchung und Mitarbeitervernetzung und bilden somit diverse Geschäftsprozesse ab, die sich teils auch in anderen Unternehmen vorfinden lassen. Insgesamt zeigt sich damit bei den SWM ein breites, bereits bestehendes Erfahrungsfeld, welches es im Rahmen dieser Fallstudie ermöglicht, sowohl positive als auch negative Erfahrungen mit dem Einsatz von Low-Code im Rahmen von Business Unit Development zu identifizieren.

Die Analyseeinheit dieser Fallstudie stellt die individuelle Perspektive von Mitarbeitenden der SWM dar, welche aktiv als Business Unit Developer tätig sind und mit Low-Code Anwendungen entwickeln. Der Fokus auf diese Perspektive ermöglicht es, individuelle Wahrnehmungen und Ansichten beim Einsatz von Low-Code im persönlichen Arbeitskontext zu erfassen. Die Erkenntnisse auf der individuellen Ebene des Business Unit Developments können dabei jedoch zugleich auch Implikationen auf die Team-Ebene sowie die unternehmensweite Organisation des Business Unit Development haben und bspw. dabei unterstützen, die Relevanz bestimmter Rahmenbedingungen zu bewerten.

Zur Datenerhebung wurden dabei zwischen August und Oktober 2023 16 Interviews durchgeführt, aufgeteilt in zwei Phasen. In der ersten Phase wurden Interviews mit 12 Business Unit Developern durchgeführt (BUD1-12), gefolgt von einer zweiten Phase von Interviews mit vier am Business Unit Development beteiligten Führungskräften der IT-Abteilung (ITM1-4). Für beide Phasen wurde jeweils angepasste Interview-Leitfäden ausgearbeitet, die Fragen zur individuellen Wahrnehmung des Business Unit Developments, der Zusammenarbeit mit anderen Mitarbeitern im Rahmen der Entwicklung sowie zur Organisation der Low-Code-Entwicklung im Unternehmen und erforderlichen Rahmenbedingungen umfassten. Da im Vorfeld der Gespräche der Umfang der bisherigen Erfahrungen des Gesprächspartners mit Low-Code in den meisten Fällen jedoch noch unklar war, wurde ein semi-strukturierter Interviewansatz verfolgt, der es ermöglichte auf spezifische forschungsrelevante Themen genauer einzugehen (Myers und Newman 2007). Die Interviews wurden virtuell über Microsoft Teams und in einem Fall in Präsenz durchgeführt und dauerten durchschnittlich 41 min an. Zusätzlich zu den Interviews wurden interne Dokumente der SWM gesichtet, um das Verständnis über den Aufbau und die Umsetzung der unternehmensinternen Business Unit Development Initiative zu vertiefen.

Für die Analyse der Interviews wurde der Ansatz von Gioia et al. (2013) verfolgt. Dies ermöglicht es, die Genauigkeit bei der Gewinnung von qualitativen Ergebnissen mittels induktivem Coding zu erhöhen, indem die Datenbasis in mehreren Stufen untersucht wird und dabei sowohl die Perspektive des Interviewpartners als auch die des Forschers berücksichtigt werden (Gioia et al. 2013). Zu Beginn wurden in zwei initialen Iterationen ausschließlich aus den Transkripten der Interviews textnahe Codes abgeleitet und diese verschiedenen Konzepten zugeordnet, die unmittelbaren

Tab. 1 Beispielhafte Entwicklung eines einzelnen Konzepts

Beispielhafte Textpassage	Initialer Code	Konzept
„Ich hab viele - meine Hauptaufgaben sind große Projekte gehen über Jahre, das ist jahrelanges Projektmanagement mit viel Kleinkram, E-Mail hier, Termin da und so weiter. Du hast wenig konkrete Erfolgserlebnisse. Und da hast du ja beim Citizen [Business Unit] Development ein kleines Projekt, wo du in eigentlich relativ kurzer Zeit ziemlich schnell das Erfolgserlebnis haben kannst. Und das ist sehr befriedigend.“ (BUD2)	Erfolgserlebnis	Höhere Zufriedenheit im Arbeitsalltag
„Also das ist halt auch einfach was, was mir mega Spaß macht. Deswegen bin ich immer froh, wenn ich irgendwas damit machen kann, deswegen wenn dann, dann ist das eher was Positives.“ (BUD4)	Spaß/Interesse	
„Also ich glaube, das war eigentlich, eher, kann man auch sagen, wenn man vielleicht mal ein bisschen Abstand braucht, von den, von den eigenen Themen gerade, dann kann man sich auch mal eine halbe Stunde irgendwie da drauf stürzen und, ich sag mal, in dem Bereich auch ein bisschen weiterbilden“ (BUD9)	Mehr Abwechslung	

Bezug zum Forschungsschwerpunkt dieser Arbeit aufwiesen. In Tab. 1 wird die Zuordnung der ursprünglichen Textpassagen und der daraus gebildeten initialen Codes zu einem bestimmten Konzept beispielhaft dargestellt.

Durch eine weitere Sortierung der Themenbereiche wurde schließlich eine Ergebnis-Struktur entwickelt, die sich aus mehreren aggregierten Dimensionen, übergeordneten Themen sowie einzelnen Konzepten zusammensetzt.

4 Ergebnisse

Die Ergebnisse der Forschung untergliedern sich in *individuelle Vorteile* und *individuelle Herausforderungen* im Rahmen der Entwicklung und Nutzung von Low-Code-Anwendungen. Im ersten Abschnitt werden in Hinblick auf die erste Forschungsfrage zunächst in der Praxis wahrgenommene Vorteile von Mitarbeitenden in Bezug auf deren Arbeitsalltag, Karriere sowie Wissen und Austausch im Unternehmen dargelegt. Der zweite Abschnitt bezieht sich auf die zweite Forschungsfrage und fokussiert sich auf Herausforderungen in Hinblick auf technische Einschränkungen bei der Realisierung von individuellen Anwendungsfällen sowie die Auswirkungen der zusätzlichen Rolle von Mitarbeitenden als Business Unit Developer auf deren persönliche Arbeitsbelastung und Verantwortungslast. Die Ergebnisse sind in Abb. 1 zusammengefasst und werden nachfolgend ausführlich erläutert.

4.1 Vorteile

Die Partizipation am Business Unit Development birgt das Potenzial, die Wahrnehmung des Arbeitsalltags, des Arbeitsumfelds sowie die Karrierechancen zu verändern und kann den Aufbau sowie den Austausch von Wissen fördern.

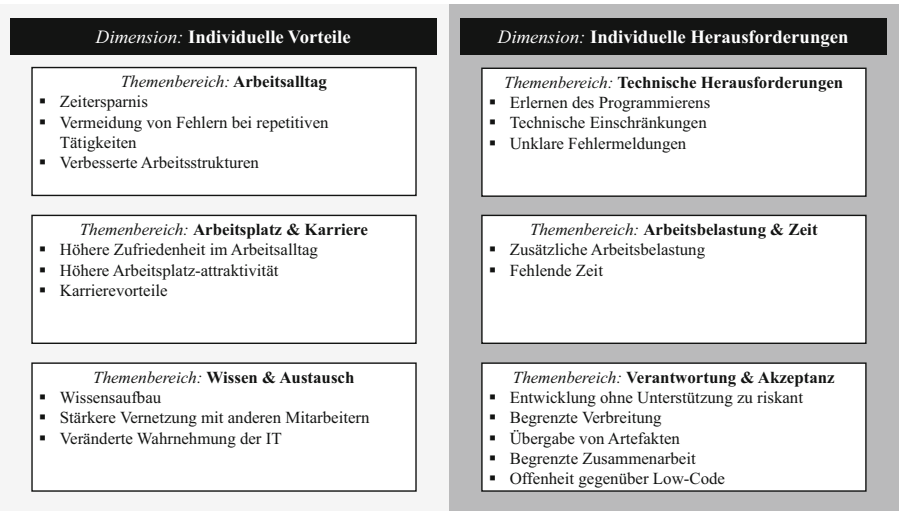


Abb. 1 Übersicht der identifizierten individuellen Vorteile und Herausforderungen aus der Einzelfallstudie

4.1.1 Arbeitsalltag

Zunächst wurden Vorteile wie Zeitersparnis, Vermeidung von Fehlern bei repetitiven Tätigkeiten und verbesserte Arbeitsstrukturen identifiziert, die sich unmittelbar auf den Arbeitsalltag des Business Unit Developers auswirken.

Im Rahmen von Business Unit Development entwickelte Low-Code-Anwendungen führen im Kontext dieser Forschung in vielen Fällen zu einer *Zeitersparnis* im Arbeitsalltag des Mitarbeiters. Zum einen resultieren diese aus der Digitalisierung eines bestehenden Prozesses. Ein Beispiel ist die Umstellung der bisher papierbasierten Dokumentation von Baukontrollen auf U-Bahn-Baustellen auf eine von BUD10 entwickelte digitale Dokumentationslösung auf Basis einer Power App. Diese kann über mobile Endgeräte direkt vor Ort ausgeführt werden und verringert damit die Anzahl der Fahrten des Business Unit Developers und seines Teams ins Büro (BUD10). Andererseits ergibt sich eine Zeitersparnis auch durch die Automatisierung von repetitiven Tätigkeiten. Beispiele stellen etwa die automatische Abfrage und Erinnerung an die Abgabe von Agendapunkten für wiederkehrende Jour Fixe (BUD6), die automatische Bearbeitung und Weiterleitung von externen Anträgen und Anfragen an die entsprechenden Fachabteilungen (BUD11) oder auch eine automatisierte Lösung für die Ablage von E-Mails mittels Power Automate (BUD12) dar. BUD6 sieht die Automatisierung dieser Tätigkeiten mit Low-Code insbesondere deswegen positiv, da es sich zumeist um Tätigkeiten handelt, die wenig komplex sind und ungern ausgeführt werden. Allerdings hebt BUD6 ebenfalls hervor, dass *„eine gewisse Routine oder Häufigkeit der Anwendung“* dieser Tätigkeit vorhanden sein sollte, damit die Zeitersparnis den initialen Entwicklungsaufwand rechtfertigt.

Die Automatisierung von Prozessen mit Low-Code hilft befragten Mitarbeitern ebenfalls dabei, *Fehler zu vermeiden*, die bei manueller Ausführung von repetitiven

Tätigkeiten entstehen. Bei BUD8 ist dies unmittelbar mit der Zeitersparnis verbunden, da dieser als Datenbeauftragter bei der Analyse von Mitarbeiter-Berechtigungen einen manuellen und fehleranfälligen Datenexport aus einem zentralen IT-System durchführen muss. Dieser Prozess wurde durch BUD8 mittels eines Power Automate Desktop Bots optimiert, der die manuellen Eingaben automatisch ausführt und durch die Vermeidung von Fehleingaben zu einer deutlichen Zeitersparnis führt. Die möglichen Fehler bei repetitiven Aufgaben können von geringfügiger Natur sein, wie bspw. das unbeabsichtigte Übernehmen einer falschen Jahreszahl in einer Termineinladung, jedoch potenziell auch weitreichendere Auswirkungen haben (BUD6).

Low-Code-Anwendungen können zudem bestehende *Arbeitsstrukturen optimieren*, indem sie die Prozessorganisation verbessern oder Arbeitsschritte vereinfachen (BUD7). Die Entwicklung eigener Anwendungen durch die Mitarbeiter führt in mehreren beobachteten Fällen zu maßgeschneiderten Lösungen, die sowohl die Effizienz des Einzelnen als auch die des Teams steigern. Bei BUD7 werden bspw. Mail-Anfragen zur Busvermietung automatisch ausgelesen und strukturiert abgespeichert, wodurch die Nachbereitung der Anfragen vereinfacht wird. Auch bei BUD12 erleichtert die automatische Mailablage die Arbeitsabläufe des Teams, da sich E-Mails leichter auffinden lassen und zugleich die Rechtssicherheit erhöht wird. Zuletzt bietet die Integration zwischen Systemen, wie von BUD4 angeführt, einen wesentlichen Vorteil, da aufgrund dieser Schnittstellen die Nutzerfreundlichkeit durch einen direkten Zugriff auf das Mitarbeiterverzeichnis oder die Einbettung von Power Apps in Microsoft Teams verbessert wird.

4.1.2 Arbeitsplatz und Karriere

In den Interviews wurden zudem positive Auswirkungen wie eine höhere Zufriedenheit im Arbeitsalltag, eine höhere Arbeitsplatzattraktivität und Karrierevorteile des Business Unit Developers identifiziert.

Insgesamt stellt das Business Unit Development für acht der zwölf befragten Mitarbeitenden eine höhere *Zufriedenheit* des Arbeitsalltags dar. Dies zeigt sich allen voran durch Spaß und Interesse der Mitarbeiter an der Entwicklung mit Low-Code. Das hohe Interesse an Low-Code führt bei manchen Mitarbeitern selbst dazu, dass diese sich auch außerhalb ihrer Arbeitszeit mit Low-Code beschäftigen (BUD2; BUD6). Insbesondere die Unterstützung und gemeinsame Entwicklung mit den Low-Code-Experten der IT wird positiv hervorgehoben (BUD7; BUD12). Auch das Auftreten von Erfolgserlebnissen im Rahmen der Low-Code-Entwicklung, etwa durch die Umsetzung von Anwendungen (BUD5) bzw. Features (BUD2) ohne fremde Hilfe, bestärkt die Mitarbeiter in ihrer Begeisterung von Low-Code. BUD2 hebt dabei zudem hervor, dass in dessen Haupttätigkeit ausschließlich an langfristigen Projekten gearbeitet wird, die über mehrere Jahre andauern. Das Business Unit Development stellt hierzu eine Abwechslung dar: *„Und da hast du ja beim Citizen [Business Unit] Development ein kleines Projekt, wo du in eigentlich relativ kurzer Zeit ziemlich schnell das Erfolgserlebnis haben kannst. Und das ist sehr befriedigend.“* (BUD2). Auch BUD9 sieht das Business Unit Development als Möglichkeit, vorübergehend *„Abstand“* von den eigenen Kerntätigkeiten zu gewinnen.

Das Vorhandensein einer Business Unit Development Initiative kann sich positiv auf die *Attraktivität des Arbeitsplatzes* bzw. des Arbeitgebers auswirken (BUD5). BUD5 wurde bereits während des Bewerbungsvorgangs auf das Business Unit Development Programm des Unternehmens aufmerksam und bezeichnet das Business Unit Development mittlerweile als entscheidenden Faktor für die Zufriedenheit mit dem aktuellen Arbeitsplatz: *„Also ich würde jetzt nicht meinen Job machen, wenn ich dieses Citizen [Business Unit] Development und Prozessoptimierung nicht hätte, dann wäre es mir zu öde. Also das (...) macht meinen Job spannend.“*.

Mitarbeiter des Fallpartners nehmen durch ihre Rolle als Business Unit Developer zudem einen positiven *Einfluss auf ihre Karriere* wahr (BUD4; BUD5). BUD5 erkennt dabei den zusätzlichen Wissensgewinn als persönlichen Nutzen, der dabei helfen kann, sich gegenüber anderen Mitbewerbern hervorzuheben. Auch BUD4 sieht die Teilnahme am Business Unit Development förderlich für die eigene Karriere, da sich insbesondere in der Anfangsphase der Business Unit Development-Initiative Expertise durch Erfahrung mit Low-Code im Unternehmen aufbauen lässt, die auch nach außen hin sichtbar wird und zu einem beruflichen Aufstieg helfen kann.

4.1.3 Wissen und Austausch

Zuletzt können sich auf der individuellen Ebene Vorteile wie Wissensaufbau, eine stärkere Vernetzung mit anderen Mitarbeitenden und eine veränderte Wahrnehmung der IT ergeben.

Eine der Motivationen von befragten Mitarbeitern sich im Business Unit Development zu engagieren ist der *Wissensaufbau* und der Drang zur Weiterbildung (BUD1; BUD4; BUD7; BUD8). Die Weiterbildung wird als nützlich erachtet, um Low-Code für den persönlichen Anwendungsbereich bzw. die Abteilung besser einsetzen zu können (BUD7; BUD8) und wird auch seitens des eigenen Fachbereichs unterstützt (BUD8). Auch Business Unit Developer, die bereits einige Anwendungsfälle umgesetzt haben und mehr Erfahrung sammeln konnten, sehen immer noch offenes Potenzial, das eigene Wissen zu erweitern und weitere Möglichkeiten von Low-Code in Erfahrung zu bringen (BUD1).

Zudem kann Business Unit Development den *Austausch zwischen Mitarbeitern fördern*, der auch über die Fachbereichsgrenzen hinausgeht. Dabei zeigt sich im Forschungskontext insbesondere ein Austausch in Bezug auf Low-Code-Lösungen, die bereits von Business Unit Developern umgesetzt wurden (BUD9; BUD12). Dies ermöglicht es Mitarbeitern einerseits, einen Einblick in andere Abteilungen und Unternehmensbereiche zu erhalten und andererseits auch Impulse und neue Ideen für die eigene Arbeit mitzunehmen, da Mitarbeiter oft auch im eigenen Fachbereich mit ähnlichen Problemen konfrontiert sind (BUD12). Zusätzlich werden jedoch auch Low-Code-Anwendungen im Unternehmen eingesetzt, die von Business Unit Developern explizit entwickelt wurden, um Mitarbeiter miteinander zu vernetzen (BUD9).

Darüber hinaus zeigt sich im Rahmen dieser Fallstudie ein deutlicher Einfluss des Business Unit Developments auf die *Wahrnehmung* der Mitarbeiter *zur IT-Abteilung*. Abgesehen vom Business Unit Development beschränkt sich der Austausch der befragten Mitarbeiter zumeist auf die Lösung von technischen Problemen mit

dem IT-Support (BUD7; BUD12) oder große IT-Projekte (BUD2; BUD3) und wird zu Teilen negativ wahrgenommen (BUD2). Das Business Unit Development trägt zu einer deutlich stärker ausgeprägten Zusammenarbeit mit der IT-Abteilung bei (BUD1; BUD7; BUD12). Dabei wird insbesondere der direkte persönliche Kontakt zu den Low-Code-Experten positiv betont, der dazu führt, dass die IT deutlich offener (BUD11) und *„total nahbar“* (BUD7) wahrgenommen wird. Durch das Business Unit Development entsteht zudem ein besseres Verständnis für die Tätigkeiten der IT: *„Man wächst auf jeden Fall mehr in die IT mit rein. Man kriegt mehr Einblick in die Prozesse hinter der IT.“* (BUD1). Auch von Seiten der IT-Führungskräfte wird diese Entwicklung begrüßt, da diese es ermöglicht die Herausforderungen und Risiken der Softwareentwicklung an Mitarbeitende in den Fachbereichen anschaulich zu vermitteln (ITM3).

4.2 Herausforderungen

4.2.1 Technische Herausforderungen

Für die befragten Mitarbeiter ergeben sich mitunter verschiedene Herausforderungen, die mit technischen Schwierigkeiten wie dem Erlernen des Programmierens, technischen Einschränkungen der LCDP und dem Umgang mit unklaren Fehlermeldungen verbunden sind.

Trotz der im Vergleich zur klassischen Programmierung vereinfachten Entwicklung mit Low-Code ist der Einstieg und das *Erlernen des Programmierens* für befragte Mitarbeiter teils herausfordernd (BUD3; BUD5). BUD3 betont dabei bspw., dass insbesondere die Verwendung von eingebauten Formeln eine Schwierigkeit darstellt und auch mit bisheriger Erfahrung mit Excel-Formeln die Programmierung in Power Apps und Power Automate nicht direkt intuitiv ist. Dabei ist die Einstiegshürde der Entwicklung mit Low-Code jedoch auch stark vom jeweiligen Low-Code-Tool abhängig. Beispielsweise empfindet BUD5 die Entwicklung von lokalen Automatisierungslösungen mit Power Automate Desktop wesentlich intuitiver als die Entwicklung von Apps und Flows mit den cloudbasierten Tools Power Apps und Power Automate. Insgesamt zeigt sich zudem, dass Vorerfahrung im Programmieren für den Einstieg in Low-Code stark förderlich ist (BUD2; BUD8).

In verschiedenen Fällen stoßen Business Unit Developer auf *Einschränkungen der Power Platform*, die eine zusätzliche Herausforderung darstellen und die Umsetzung ihres Anwendungsfalls erschweren. Beim Anwendungsfall von BUD1 ist bspw. eine Offline-Funktionalität zwingend erforderlich, um die Wartungsdokumentation innerhalb der U-Bahn-Anlagen vor Ort durchführen zu können. Diese Funktionalität ist jedoch nativ nicht in Power Apps integriert und muss eigenständig implementiert werden. Auch an anderen Stellen (BUD2; BUD9) stoßen Business Unit Developer an Grenzen der LCDP und sind daraufhin angewiesen, *„Bastellösungen“* (BUD9) zu nutzen, um bestimmte Funktionalitäten im Rahmen ihres Anwendungsfalls umsetzen zu können.

Business Unit Developer des Forschungspartners sind zudem mit *unklaren Fehlermeldungen* konfrontiert. Beispielsweise traten während der Entwicklung der Mail-Archivierung von BUD12 Fehler auf, deren Beschreibung jedoch nicht aussage-

kräftig war, sodass erst im *„Try-And-Error“*-Verfahren das Problem auf unzulässige Sonderzeichen im Betreff der Mails zurückgeführt werden konnte. Auch bei BUD2 zeigten sich unklare Fehlermeldungen als Herausforderung während der Entwicklung und erschwerten bspw. das Debugging eines Zeitauswahlfeldes, bei dem es zu Fehlverhalten aufgrund einer falschen Zeitzone kam.

4.2.2 Arbeitsbelastung und Zeit

Zudem wurden Herausforderungen in Verbindung mit der Arbeitsbelastung und der für das Business Unit Development benötigten Zeit identifiziert.

Die Entwicklung der Low-Code-Anwendung stellt für die befragten Mitarbeiter teilweise eine spürbare *zusätzliche Arbeitsbelastung* dar (BUD2; BUD7; BUD10; BUD12). Daher zeigt sich ein klares Zeitmanagement erforderlich, um die Entwicklung neben den eigentlichen Tätigkeiten in den eigenen Arbeitsalltag zu integrieren (BUD7; BUD11). In manchen Fällen wird zudem die Entwicklungszeit flexibel aufgeteilt, wie bspw. bei BUD11, der die Entwicklung der Anwendung zur Mail-Archivierung über ein halbes Jahr gestreckt hat. Die zusätzliche Arbeitsbelastung bezieht sich jedoch insbesondere auf die initiale Entwicklung der Anwendung und wird nach der Fertigstellung und Produktivsetzung der Anwendung nicht mehr wahrgenommen (BUD7). Zudem wird der zusätzliche Arbeitsaufwand durch die Entwicklung von Business Unit Developern auch positiv aufgefasst, da diese mit Spaß und eigenem Interesse verbunden sind und zu mehr Abwechslung im Arbeitsalltag führen (BUD2; BUD4; BUD9).

Für viele der befragten Mitarbeiter ist die *fehlende Zeit* im Arbeitsalltag generell eine Herausforderung für die Beschäftigung mit Low-Code. Oft sind Mitarbeiter bereits im Rahmen ihrer eigentlichen Tätigkeiten mit einem hohen Zeitdruck konfrontiert (BUD10) und insbesondere in kleineren Teams ist dadurch nur wenig Freiraum, um sich mit zusätzlichen Themen wie Low-Code zu beschäftigen (BUD11). Dies führt auch dazu, dass offenes Potenzial für die Umsetzung weiterer Anwendungsfälle zur Prozessoptimierung nicht genutzt wird (BUD10; BUD12). BUD6 bezeichnet das Problem der fehlenden Zeit im Zusammenhang mit der Beschäftigung mit Low-Code jedoch auch als *„Teufelskreis“*, da durch die Automatisierung und Optimierung von Prozessen wieder Zeit freigeräumt werden kann und somit Low-Code *„unterstützend wirken kann“*.

4.2.3 Verantwortung und Akzeptanz

Zuletzt zeigten sich in den Interviews Herausforderungen, die sich auf die Verantwortung des Mitarbeiters für die Low-Code-Artefakte beziehen oder mit der Akzeptanz des Business Unit Developments von Kollegen und anderen Mitarbeitern verbunden sind. Diese Herausforderungen reichen von der Entwicklung ohne Unterstützung, über die begrenzte Verbreitung des Business Unit Developments im Team, bis hin zur Übergabe von Artefakten, der begrenzten Zusammenarbeit bei der Entwicklung und der mangelnden Offenheit über Low-Code.

Die Entwicklung mit Low-Code *ohne Unterstützung* durch einen Low-Code-Experten aus der IT-Abteilung wird vereinzelt als zu riskant betrachtet. Dabei befürch-

ten Business Unit Developer einen Schaden verursachen zu können und sehen zu wenig eigene Erfahrung mit der Entwicklung (BUD7; BUD12). BUD12 sieht daher eine Trennung der Entwicklungsumgebungen in ein Test- und eine Produktiv-Umgebung als mögliche Lösung, um negative Auswirkungen durch eigene Fehler beim Experimentieren mit Low-Code zu vermeiden. Zudem betrachtet BUD9 auch die Einhaltung der Datenschutzkonformität beim Umgang mit personenbezogenen Daten als herausfordernd und erachtet bei der Umsetzung eines Berechtigungskonzepts die Unterstützung durch die Low-Code-Experten als sinnvoll.

Eine weitere Herausforderung kann eine *mangelnde Verbreitung* des Business Unit Developments im Team darstellen. Die Mehrheit der befragten Business Unit Developer hat in ihrem direkten Umfeld keine anderen Mitarbeiter, die ebenfalls am Business Unit Development teilnehmen und mit Low-Code entwickeln. Bei BUD2 zeigt sich dies insbesondere als Problem, da bei neuen Vorschlägen zur Optimierung von internen Prozesse Austausch zur konkreten Umsetzung nur begrenzt möglich ist und anderen Kollegen das Verständnis für die Möglichkeiten von Low-Code fehlt. Dies führt auch dazu, dass die Verantwortung für die Umsetzung und Entwicklung der Anwendung ausschließlich bei BUD2 liegt: *„Und wenn ich jetzt sage, ich entwickle was und dann hängt das alles an mir, dann bin ich diejenige, die es verantwortet.“*.

Die begrenzte Verbreitung des Business Unit Developments in den einzelnen Teams kann für Business Unit Developer auch zu Schwierigkeiten bei der *Übergabe von Artefakten* führen, sobald eine längere Abwesenheit oder ein Job-Wechsel eintritt. Dies stellt beispielsweise für BUD6 eine aktuelle Herausforderung dar, da aufgrund dessen internen Wechsels nun Mitarbeiter gefunden werden müssen, die bereit sind, sich in die bestehenden und im Arbeitsalltag integrierten Low-Code-Lösungen einzuarbeiten und die Verantwortung für deren Wartung und Weiterentwicklung zu übernehmen. Auch BUD5 sieht die Herausforderung durch eine mögliche zukünftige Übergabe der eigenen Artefakte besorgt und befürchtet, dass sich durch die Übergabe an Mitarbeiter mit weniger Low-Code-Erfahrung eine zu starke Abhängigkeit zum Low-Code-Team der IT-Abteilung entwickelt. In vielen Fällen existieren zudem keine Dokumentationen für die Anwendungen, wodurch das Verständnis für den Aufbau der Anwendung unmittelbar an den ursprünglichen Entwickler gebunden ist (BUD5). Bei BUD12 wurden jedoch bereits während der Entwicklung Vorkehrungen getroffen, um die Wartbarkeit zu verbessern und eine Übergabe zu vereinfachen, bspw. durch eine Aufteilung der Flows und die Einbindung des gesamten Teams in den Entwicklungsprozess sowie die Vergabe der notwendigen Berechtigungen.

Außerhalb der „Brown Bag Sessions“[2] findet beim Fallpartner zudem aktuell *kaum intensiver Austausch und Kollaboration* zwischen Business Unit Developern statt. BUD2 betrachtet die Entwicklung momentan als *„isoliert“* und befürchtet dadurch, das Entstehen von *„Insellösungen“*, welches durch eine bessere Vernetzung von Business Unit Developern untereinander verhindert werden könnte. Sowohl Business Unit Developer als auch die IT fassen dabei eine verstärkte Zusammenarbeit während der Entwicklungsphase positiv auf und sehen eine fachbereichsüber-

[2] Wöchentlich virtuell stattfindende Termine, bei denen Business Unit Developer eigens entwickelte Low-Code Anwendungen vorstellen.

greifende Kollaboration zwischen Business Unit Developern als mögliche zukünftige Entwicklung der Community.

Insgesamt zeigt sich eine gemischte *Einstellung von Mitarbeitern gegenüber Low-Code* sowie generell gegenüber Veränderungen durch Digitalisierung und Optimierung von Prozessen. Mehrere Business Unit Developer sprechen von negativen Erfahrungen in Bezug auf die Akzeptanz Low-Code (BUD2; BUD5; BUD10). Beispielsweise bezeichnet es BUD5 als *„herausfordernd"*, die Digitalisierung im eigenen Fachbereich voranzutreiben und stößt mit neuen Ideen bei Kollegen auf Ignoranz oder sogar *„Gegenwind"*, wodurch starkes Eigenengagement und Überzeugungsarbeit erforderlich ist, um Veränderungen durchzusetzen: *„Man muss da schon echt sehr viel Begeisterung mitbringen, damit man da auch andere mitzieht, weil sonst würde nichts passieren"*. Auch BUD2 beobachtet eine *„Ablehnung gegenüber IT-Lösungen"* bei anderen Mitarbeitern und sieht dabei insbesondere fehlendes IT-technisches Verständnis als Ursache. Dabei kann auch bereits der Begriff *„Programmieren"* abschreckend wirken und insbesondere ältere Mitarbeiter davon abhalten, sich grundsätzlich mit Low-Code zu beschäftigen (BUD10). Jedoch ist die Akzeptanz von Low-Code und Digitalisierung im Allgemeinen auch stark abhängig vom jeweiligen Fachbereich. Beispielsweise zeigt sich das Team von BUD8 positiv gegenüber dessen Tätigkeit als Business Unit Developer und hat den Anspruch *„innovativ"* zu sein. Auch BUD11 erlebt bei sich positive Erfahrungen und hat im Team ein gemeinsames Verständnis, dass alte Prozesse mithilfe der vorhandenen Tools optimiert werden müssen.

5 Diskussion

5.1 Vorteile und Herausforderungen durch den Einsatz von LCDPs

Die digitale Transformation stellt Unternehmen vor die Herausforderung, ihre Arbeitsprozesse kontinuierlich zu optimieren und zu digitalisieren, um im dynamischen Wettbewerbsumfeld bestehen zu können. Ein signifikantes Problem in diesem Kontext ist der anhaltende Mangel an IT-Fachkräften, der die Entwicklung und Implementierung digitaler Lösungen erheblich erschwert. Theoretisch besteht eine Lücke im Verständnis darüber, wie Unternehmen durch die Ermächtigung von Mitarbeitern außerhalb der IT-Abteilungen, mittels LCNC-Plattformen als Business Unit Developer zu agieren, diesen Herausforderungen effektiv begegnen können. Zudem fehlen tiefgreifende Erkenntnisse darüber, wie solche Business Unit Development Initiativen die Arbeitsweise, die Effizienz und die Innovationsfähigkeit von Unternehmen beeinflussen.

Die durchgeführte Forschung hat zentrale Erkenntnisse hervorgebracht, die sowohl das theoretische als auch das praktische Verständnis von Business Unit Development in Unternehmen erweitern. In Bezug auf die erste Forschungsfrage *„Welche Vorteile ergeben sich für Mitarbeiter durch das Business Unit Development?"* zeigen die Ergebnisse, dass Business Unit Development erhebliche Vorteile für die beteiligten Mitarbeiter und ihre Arbeitsprozesse mit sich bringen kann. Das Erzielen von Effizienzsteigerungen bei bestehenden Prozessen, die Vermeidung von Fehlern bei

repetitiven Aufgaben sowie die Optimierung von Arbeitsstrukturen lassen sich dabei auf die direkte Beteiligung von Fachbereichsmitarbeitern an der Anwendungsentwicklung zurückführen. Diese sind mit den Geschäftsabläufen und Prozessen vertraut und werden durch die Verfügbarkeit von LCDPs ermächtigt, Automatisierungslösungen von hoher Präzision und Relevanz zu implementieren. Zudem ermöglicht es der Einsatz von LCDPs Mitarbeitern, grundlegende Informatik- und Software-Engineering-Kenntnisse zu erwerben und ein Verständnis für die Realisierung von digitalen Lösungen mithilfe von Low-Code aufzubauen. Dies trägt dazu bei, die Wissenslücke zwischen der fachlichen und der technischen Domäne zu schließen, und resultiert in einer gestiegenen Zufriedenheit der Mitarbeiter, da diese durch das zusätzliche Wissen eigene Anforderungen und Bedürfnisse eigenständig umsetzen können. Schließlich unterstützen der Aufbau von theoretischem Wissen und das Sammeln von praktischen Entwicklungserfahrungen auch die Annäherung zwischen der IT-Abteilung und einzelnen Fachbereichen. Dabei ermöglicht Low-Code eine gemeinsame Sicht auf die Abbildung von fachlichen Anforderungen in technischen Lösungen und macht zugleich auch dabei auftretende Herausforderungen und Risiken für beide Seiten nachvollziehbar.

In Hinblick auf die zweite Forschungsfrage, *„Mit welchen Herausforderungen sind Mitarbeiter durch das Business Unit Development konfrontiert?“* zeigt sich, dass neben technischen Herausforderungen insbesondere auch soziale Herausforderungen von aktiven Business Unit Developern wahrgenommen werden. Die Demokratisierung der Anwendungsentwicklung durch LCDPs ermöglicht es Mitarbeitern, die digitalen Transformation mitzugestalten, erfordert jedoch zugleich auch das Vorhandensein von Fokuszeitblöcken und Wissensresourcen für die persönliche Weiterbildung sowie ein gemeinsames Verständnis innerhalb der Fachbereiche für die Rolle des Business Unit Developers und dessen Potenzial. Die beobachteten individuellen Herausforderungen legen damit nahe, dass die erfolgreiche Implementierung von Business Unit Development Initiativen eine adäquate Unterstützung durch das Management und die IT-Abteilung, eine innovationsfördernde Unternehmenskultur und eine gezielte Förderung der Mitarbeiterkompetenzen voraussetzt.

5.2 Maßnahmen zur Vorteilsrealisierung und Handlungsempfehlungen

Zur Beantwortung der dritten Forschungsfrage *„Welche Maßnahmen sind zur Realisierung der Vorteile aus dem Business Unit Development notwendig?“* ermöglichen es die Erkenntnisse, mehrere Mechanismen zur Realisierung der Vorteile von Business Unit Development zu identifizieren sowie Handlungsempfehlungen zur effektiven Adressierung der identifizierten Herausforderungen zusammen zu fassen. Diese sind in Tab. 2 dargestellt und werden im Nachfolgenden erläutert.

Die erste Dimension zur Umsetzung von Business Unit Development in Unternehmen beschreibt den Mechanismus der *Effizienzsteigerung durch Prozessautomatisierung im Arbeitsalltag*. Dieser Mechanismus erlaubt es, durch die Entwicklung von Strategien zur Identifikation und Priorisierung automatisierbarer Prozesse, unter Berücksichtigung der Arbeitsbelastung der Mitarbeiter, die täglichen Arbeitsprozesse durch Business Unit Developer zu optimieren. Die Implementierung eines Feedbacksystems, das die Auswirkungen der Automatisierung auf den Arbeitsalltag

Tab. 2 Dimensionen, Mechanismen und Handlungsempfehlungen zur Einführung von Business Unit Development in Unternehmen

Dimension	Mechanismen zur Vorteilsrealisierung	Handlungsempfehlung zur Adressierung identifizierter Herausforderungen
Arbeitsalltag	Effizienzsteigerung durch Prozessautomatisierung	Entwickeln von Strategien zur Identifikation und Priorisierung automatisierbarer Prozesse unter Berücksichtigung der Arbeitsbelastung der Mitarbeiter. Implementieren eines Feedbacksystem, das es ermöglicht, die Auswirkungen der Automatisierung auf den Arbeitsalltag kontinuierlich zu bewerten und anzupassen, um Überforderung zu vermeiden und die Akzeptanz zu erhöhen
Arbeitsplatz & Karriere	Erhöhung der Arbeitszufriedenheit Entwicklung von Karrierepfaden	Integrieren von Business Unit Development die Personalentwicklungsstrategie, indem spezifische Karrierepfade für Business Unit Developer definiert und transparent gemacht werden. Etablieren einer Unternehmenskultur, die Innovation und Eigeninitiative wertschätzt, und Anerkennung sowie Aufstiegsmöglichkeiten für Mitarbeiter, die sich in diesem Bereich engagieren und weiterbilden, anbietet
Wissen & Austausch	Förderung von Wissensaustausch und kollaborativer Innovation	Etablieren einer Low-Code-Community als Plattform zur Vernetzung und Kollaboration zwischen Business Unit Developern Anbieten von Schulungen, Workshops und regelmäßigen Austauschformaten

kontinuierlich bewertet, ermöglicht eine Anpassung, um Überforderung zu vermeiden und die Akzeptanz zu erhöhen. Diese Ergebnisse spiegeln sich in der Literatur wider, wo die direkte Beteiligung der Endnutzer am Entwicklungsprozess als Schlüssel zur maßgeschneiderten Digitalisierung von Prozessen gesehen wird (Bies et al. 2022; Krejci et al. 2022), und unterstreichen die Bedeutung der Nutzerzentrierung in Business Unit Development Initiativen. Als Implikation steht hierbei die Aufgabe von Unternehmen eine Balance zwischen der Automatisierung von Aufgaben und der Arbeitsbelastung der Mitarbeiter zu finden, um die Effizienz und Zufriedenheit im Arbeitsalltag zu maximieren.

Die zweite Dimension zur Umsetzung von Business Unit Development zeigt, dass eine *Erhöhung der Arbeitszufriedenheit* und *Entwicklung von Karrierepfaden* eine integrative Einbettung in die Personalentwicklungsstrategie notwendig macht. Durch die Definition spezifischer Karrierepfade für Business Unit Developer und die Schaffung einer Unternehmenskultur, die Innovation und Eigeninitiative wertschätzt, können Unternehmen die Motivation und das Engagement ihrer Mitarbeiter in Bezug zu Business Unit Development steigern. Diese Handlungsempfehlungen stehen im Einklang mit in den in der Literatur diskutierten Konzepten, die die positive Wirkung der direkten Endnutzerbeteiligung auf die Arbeitszufriedenheit aufgrund von gestiegener Autonomie und Sinnhaftigkeit der Arbeit hervorheben (Li et al. 2022; Staaby et al. 2021). Als Implikation zeigt sich die Notwendigkeit für Unternehmen Mechanismen zur Unterstützung von Business Unit Developer zu etablieren, die nicht nur die technische Entwicklungsfähigkeit, sondern auch die berufliche Weiterentwicklung und Zufriedenheit fördern.

Als dritte Dimension zur Umsetzung von Business Unit Development in Unternehmen zeigt sich die Förderung von *Wissensaustausch und kollaborativer Innovation*. Die Etablierung einer unternehmensinternen Business Unit Development Community sowie die Durchführung von Low-Code-Schulungen und Workshops unterstützen Business Unit Developer, sich mit der Low-Code-Entwicklung vertraut zu machen, und fördern die kontinuierliche Weiterbildung. Diese Handlungsempfehlungen finden Unterstützung in der Literatur, welche die Bedeutung eines Community-zentrierten Ansatzes, einer Betreuung durch Low-Code-Experten aus der IT und des Wissensaustauschs innerhalb von Unternehmen durch engere Kollaboration zwischen den Fachbereichen betonen (Carroll und Maher 2023; Krejci et al. 2021; Masili 2023). Als Implikation zeigt sich die Notwendigkeit zur Förderung des Austausches und der Zusammenarbeit zwischen Business Unit Developern sowie mit der IT-Abteilung. Beispielsweise können regelmäßig stattfindende Austauschformate, etwa zu umgesetzten Anwendungsfällen oder Best Practices und Learnings, sowie die Organisation von Low-Code-Hackathons dabei helfen, die Verbreitung von Business Unit Development in Unternehmen zu stärken und damit eine kollektive Begeisterung für Innovation mithilfe von Low-Code hervorzurufen.

6 Fazit

Insgesamt verdeutlichen die Ergebnisse dieser Forschung, dass für die erfolgreiche Etablierung einer Business Unit Development Initiative nicht nur die reine Absicht

des Unternehmens, sondern insbesondere auch die Motivation und Beteiligung der Mitarbeitenden eine wesentliche Rolle spielt. Am Beispiel der Business Unit Development Initiative der SWM zeigt sich, dass sich für teilnehmende Mitarbeiter diverse persönliche Vorteile ergeben können und die Möglichkeit zur Entwicklung und Verwendung von Low-Code-Anwendungen im Arbeitsalltag äußerst positiv wahrgenommen wird. Gleichzeitig veranschaulichen die identifizierten Herausforderungen, dass die reine Bereitstellung von Low-Code-Development-Plattformen nicht ausreicht, um einen effektiven und nachhaltigen Mehrwert für das Unternehmen zu bieten. Daher zeigen sich allen voran eine innovationsfördernde Unternehmenskultur, die Etablierung einer Low-Code-Community sowie die umfassende Unterstützung von Business Unit Developer über technische Aspekte hinaus als positive Einflussfaktoren für den erfolgreichen Aufbau und die Skalierung eines Business Unit Development Programms im Unternehmen.

Diese Forschung hat zugleich bestimmte Einschränkungen. Zunächst basieren die Ergebnisse ausschließlich auf der Analyse des Business Unit Development Programms eines einzelnen Unternehmens, was neue und interessante Einblicke in Business Unit Development Programme ermöglicht. Jedoch könnte eine Vergleichsstudie die Generalisierbarkeit und Aussagekraft der Ergebnisse erhöhen. Zudem verwenden die befragten Business Unit Developer ausschließlich die Microsoft Power Platform, was spezifische technische Herausforderungen mit sich bringt, die bei anderen LCDPs möglicherweise nicht auftreten. Obwohl zwölf aktive Business Unit Developer mit unterschiedlichen fachlichen und technischen Hintergründen befragt wurden, könnte schließlich auch eine größere Stichprobe und der Einbezug quantitativer Methoden dabei helfen, individuelle Auswirkungen genauer zu verstehen und die Häufigkeitsverteilung präziser darzustellen.

Zuletzt bieten die Ergebnisse auch mehrere Ansatzpunkte für weitere Forschung im Bereich des Business Unit Development. Zunächst zeigt sich ein möglicher Ansatzpunkt in Bezug auf den Einsatz von Low-Code zur Prozessautomatisierung. Die beispielhaft aufgezeigten resultierenden Effizienzsteigerungen verdeutlichen die positiven Auswirkungen von Low-Code-Anwendungen auf den Arbeitsalltag der Mitarbeitenden. Dabei könnte zukünftige Forschung darauf eingehen, wie sich Effizienzsteigerungen durch Prozessautomatisierung mithilfe von Low-Code quantifizieren lassen, was eine objektive Bewertung des Nutzens von Business Unit Development ermöglichen würde. Darüber hinaus zeigt sich die begrenzte Verbreitung des Business Unit Developments innerhalb der Fachbereiche als Herausforderung für Mitarbeitende. Dabei wäre insbesondere interessant zu untersuchen, welche konkreten individuellen Faktoren Mitarbeitende davon abhalten, sich mit Low-Code zu beschäftigen. Schließlich zeigt sich auch die konkrete Realisierung von spezifischen Maßnahmen im Rahmen Business Unit Development Initiativen als weiteres offenes Forschungsfeld. Beispielsweise könnte eine genaue Analyse der Umsetzung von Low-Code Hackathons, Austauschformaten oder auch Betreuungsangeboten der IT-Abteilungen, Unternehmen dabei helfen, strategische Entscheidungen zu treffen, um Mitarbeiter zur Low-Code Entwicklung zu motivieren und vorhandene Hemmschwellen abzubauen.

Funding Open Access funding enabled and organized by Projekt DEAL.

Open Access Dieser Artikel wird unter der Creative Commons Namensnennung 4.0 International Lizenz veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Artikel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.

Weitere Details zur Lizenz entnehmen Sie bitte der Lizenzinformation auf http://creativecommons.org/licenses/by/4.0/deed.de.

Literatur

Alsaadi H, Radain D, Alzahrani M, Alshammari W, Alahmadi D, Fakieh B (2021) Factors that affect the utilization of low-code development platforms: survey study. Rev Rom Inform Autom 31:123–140. https://doi.org/10.33436/v31i3y202110

Barricelli B, Cassano F, Fogli D, Piccinno A (2019) End-user development, end-user programming and end-user software engineering: A systematic mapping study. J Syst Softw 149:101–137. https://doi.org/10.1016/j.jss.2018.11.041

Bies L, Weber M, Greff T, Werth D (2022) A mixed-methods study of low-code development platforms: drivers of digital innovation in SMEs. In: 2022 international conference on electrical, computer, communications and mechatronics engineering (ICECCME), S 1–6 https://doi.org/10.1109/ICECCME55909.2022.9987920

Binzer B, Winkler T (2022) Democratizing software development: a systematic multivocal literature review and research agenda on citizen development. In: Proceedings of the 13th International Conference on Software Business (ICSOB), S 244–259 https://doi.org/10.1007/978-3-031-20706-8_17

Bitkom Reserarch (2022) Trotz Krieg und Krisen: In Deutschland fehlen 137.000 IT-Fachkräfte. https://www.bitkom.org/Presse/Presseinformation/Deutschland-fehlen-137000-IT-Fachkraefte. Zugegriffen: 18. Apr. 2023

Bock A, Frank U (2021) Low-code platform. Bus Inf Syst Eng 63(6):733–740. https://doi.org/10.1007/s12599-021-00726-8

Breidenbach K, Mathes M, Riedel M, Müller T (2023) Citizen Developer: Fachbereiche emanzipieren sich von der IT. CIO. https://www.cio.de/a/fachbereiche-emanzipieren-sich-von-der-it,3671061. Zugegriffen: 29. Juni 2023

Carroll N, Maher M (2023) How shell fueled a digital transformation by establishing DIY software development. MIS Q 22(2):99–127. https://doi.org/10.17705/2msqe.00076

Carroll N, Móráin L, Garrett D, Jamnadass A (2021) The importance of citizen development for digital transformation. Cut It J 34(3):5–9

Di Ruscio D, Kolovos D, De Lara J, Pierantonio A, Tisi M, Wimmer M (2022) Low-code development and model-driven engineering: Two sides of the same coin? Softw Syst Model 21(2):437–446. https://doi.org/10.1007/s10270-021-00970-2

Elshan E, Dickhaut E, Ebel P (2023a) An investigation of why low code platforms provide answers and new challenges. In: Hawaii International Conference on System Sciences, S 6159–6168

Elshan E, Germann D, Dickhaut E, Li M (2023b) Faster, cheaper, better? Analyzing how lowcode developement platforms drive bottom-up innovation. In: European Conference on Information Systems (ECIS)

Frank U, Maier P, Bock A (2021) Low code platforms: Promises, concepts and prospects. A comparative study of ten systems. ICB Reseach Reports (70). https://www.econstor.eu/ handle/10419/248826

Gartner (2022) Definition of citizen developer—Gartner information technology glossary. Gartner. https://www.gartner.com/en/information-technology/glossary/citizen-develo. Zugegriffen: 6. Febr. 2023

Gioia D, Corley K, Hamilton A (2013) Seeking qualitative rigor in inductive research: notes on the Gioia methodology. Organ Res Methods 16(1):15–31. https://doi.org/10.1177/1094428112452151

Hintsch J, Staegemann D, Volk M, Turowski K (2021) Low-code development platform usage: towards bringing citizen development and enterprise IT into harmony. In: 32nd Australasian Conference on Information Systems (ACIS2021)
Hoogsteen D, Borgman H (2022) Empower the workforce, empower the company? Citizen development adoption. In: Hawaii International Conference on System Sciences https://doi.org/10.24251/HICSS.2022.575
Käss S, Strahringer S, Westner M (2022) Drivers and inhibitors of low code development platform adoption. In: 2022 IEEE 24th Conference on Business Informatics (CBI), S 196–205 https://doi.org/10.1109/CBI54897.2022.00028
Klotz S, Westner M, Strahringer S (2022) Critical success factors of business-managed IT: it takes two to tango. Inf Syst Manag 39(3):220–240. https://doi.org/10.1080/10580530.2021.1938300
Kolarsch C, Mohn C, László Á, Böning-Cortierer D, Gerner J, Kacer R, Stefan G (2021) Digital.Trend.Studie Do It Yourself Computing—Die Demokratisierung der IT. https://www.dbsystel.de/resource/blob/9108632/23abdd2e9b6245f5cb45feb49376903c/Download-Trendstudie-Do-It-Yourself-Computing-data.pdf. Zugegriffen: 8. Nov. 2023
Krejci D, Iho S, Missonier S (2021) Innovating with employees: an exploratory study of idea development on low-code development platforms. ECIS 2021 Research Papers
Krejci D, Küng L, Missonier S (2022) A case study of enterprise-wide digital innovation: involving non-IT employees. ECIS 2022 Research Papers
Li M, Peters C, Poser M, Eilers K, Elshan E (2022) ICT-enabled job crafting: how business unit developers use low-code development platforms to craft jobs. ICIS2022 Proceedings
Maiya A (2022) ktrain: a low-code library for augmented machine learning. J Mach Learn Res 23(1):1–6
Martinez E, Pfister L (2023) Benefits and limitations of using low-code development to support digitalization in the construction industry. Autom Constr. https://doi.org/10.1016/j.autcon.2023.104909
Masili G (2023) No-code development platforms: breaking the boundaries between IT and business experts. Int J Econ Behav 13(1):33–49. https://doi.org/10.14276/2285-0430.3705
Myers M, Newman M (2007) The qualitative interview in IS research: Examining the craft. Inf Organ 17(1):2–26. https://doi.org/10.1016/j.infoandorg.2006.11.001
Nambisan S, Wright M, Feldman M (2019) The digital transformation of innovation and entrepreneurship: Progress, challenges and key themes. Res Policy: 1–9
Naqvi S, Zimmer M, Drews P, Lemmer K, Basole R (2024) The low-code phenomenon: mapping the intellectual structure of research. In: Proceedings of the 57th Hawaii International Conference on System Sciences
Novales A, Mancha R (2023) Fueling digital transformation with citizen developers and low-code development. MIS Q 22(3)
Prinz N, Rentrop C, Huber M (2021) Low-code development platforms—A literature review. In: AMCIS 2021 Proceedings
Richardson C, Rymer J (2014) New development platforms emerge for customer-facing applications. Forrester
Sahay A, Indamutsa A, Di Ruscio D, Pierantonio A (2020) Supporting the understanding and comparison of low-code development platforms. In: 2020 46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA) https://doi.org/10.1109/SEAA51224.2020.00036
Sanchis R, García-Perales Ó, Fraile F, Poler R (2019) Low-code as enabler of digital transformation in manufacturing industry. Appl Sci 10(1):12. https://doi.org/10.3390/app10010012
Schlaepfer R, von Radowitz K (2017) Digitale Zukunftsfähigkeit – Wie wappnen sich Unternehmen für die Chancen und Herausforderungen der Digitalisierung. https://www2.deloitte.com/content/dam/Deloitte/ch/Documents/consumer-business/ch-de-cip-swiss-transformation.pdf. Zugegriffen: 6. Febr. 2024
Sebastian I, Ross J, Beath C, Mocker M, Moloney K, Fonstad N (2017) How big old companies navigate digital transformation. MIS Q 16(3):6. https://doi.org/10.4324/9780429286797-6
Staaby A, Hansen K, Grønli TM (2021) Automation of routine work: a case study of employees' experiences of work meaningfulness. In: Hawaii International Conference on System Sciences (HICSS) https://doi.org/10.24251/HICSS.2021.018
Steele T (2021) Democratizing innovation: low-code/no-code platforms & the citizen developer. Capco
Strauss A, Corbin J (1990) Basics of qualitative research: Grounded theory procedures and techniques. SAGE, Thousand Oaks
Sundberg L, Holmström J (2023) Democratizing artificial intelligence: How no-code AI can leverage machine learning operations. Bus Horiz 66(6):777–788. https://doi.org/10.1016/j.bushor.2023.04.003
SWM (2024) Geschäftsbericht 2023

Viljoen A, Nguyen J, Kauschinger M, Hein A (2023) Fostering scalable citizen development in organizations: towards a guiding framework. AMCIS 2023 Proceedings
Vincent P, Iijima K, Driver M, Wong J, Natis Y (2019) Magic quadrant for enterprise low-code application platforms. https://www.gartner.com/en/documents/3956079. Zugegriffen: 9. Juni 2023
Yin R (2018) Case study research and applications: design and methods. SAGE, Thousand Oaks

Hinweis des Verlags Der Verlag bleibt in Hinblick auf geografische Zuordnungen und Gebietsbezeichnungen in veröffentlichten Karten und Institutsadressen neutral.