

Trend geht nach oben:

Low-Code für digitale Prozesse in Unternehmen

Trend geht nach oben: Low-Code für digitale Prozesse in Unternehmen

Low-Code-Plattformen werden immer beliebter. Prognosen, die die Entwicklung solcher Plattformen betreffen, schätzen das Marktpotential im Jahr 2022 auf circa 27, 23 Milliarden US-Dollar. Die jährliche Wachstumsrate soll dann bei über 44 Prozent liegen.⁹

Wie man es dreht und wendet: Am Thema „Low-Code“ kommt man nur schwer vorbei. Deshalb wird es Zeit sich mit diesem Trend genauer zu beschäftigen. Was verbirgt sich hinter dem Begriff, welche Vorteile, aber auch Herausforderungen bietet Low-Code und wie sieht der Einsatz dieser Plattformen in der Praxis aus? Fragen, auf die das vorliegende Whitepaper Antworten gibt.

„Enterprise development teams are adopting low-code development platforms, and the market’s growth prospects appear rosy. [...] Digital businesses’ demand for ever more software ever more quickly is the big driver of adoption.“

John R. Rhymer & Rob Koplowitz (2019). Forrester, The Forrester Wave™: Low-Code Development Platforms For AD&D Professionals, Q1 2019

Inhaltsverzeichnis

Low-Code – ein mächtiges Instrument

Aktuelle Anforderungen und Herausforderungen an Unternehmen und klassische Softwareentwicklung	4
Ursprung, Bedeutung und Entwicklung von Low-Code	6
Neue Ära von Entwicklern: Citizen Developer	8
Low-Code ist nicht gleich No-Code	9
Merkmale von Low-Code	10
Low-Code als Digitalisierungswerkzeug Einsatzszenarien.....	11



Aktuelle Anforderungen und Herausforderungen an Unternehmen und klassische Softwareentwicklung

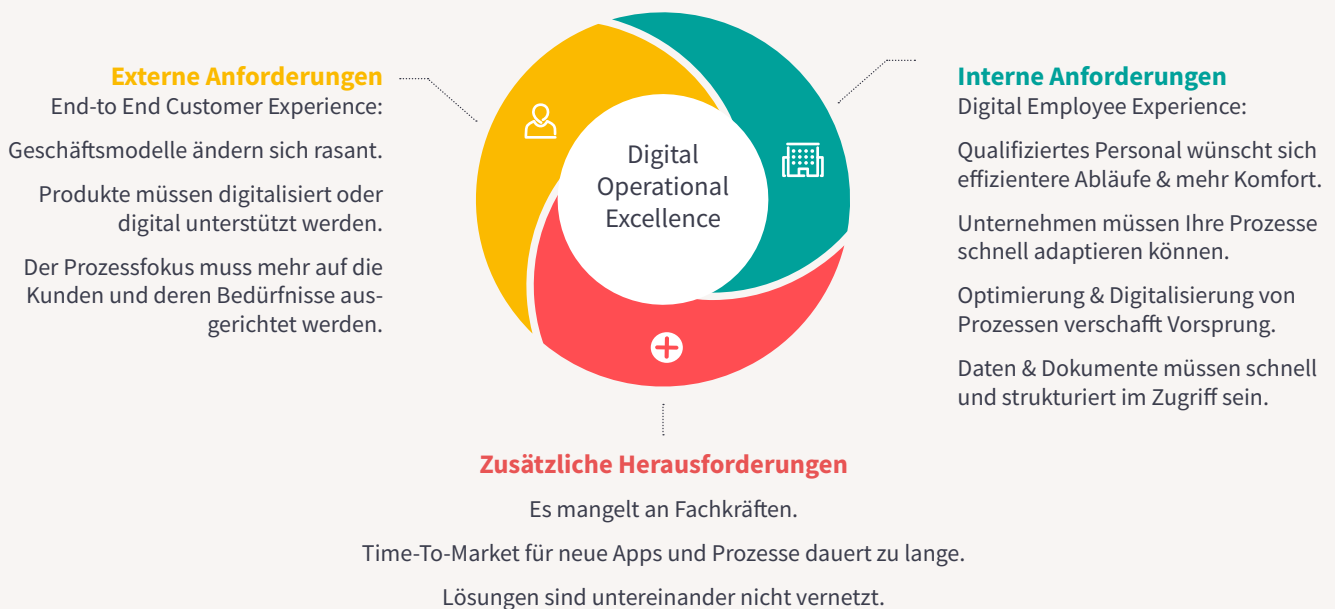
Software qualitativ hochwertig und in möglichst geringer Zeit zu entwickeln, ist ein anspruchsvolles Unterfangen. Das liegt aber nicht nur an der Tätigkeit an sich, sondern auch an den sich stetig wechselnden Anforderungen. Genau die sind es nämlich, die Unternehmen in die Knie zwingen und zum Handeln auffordern. Extern sind es Geschäftsmodelle, die sich auf dem Markt rasant ändern oder unternehmenseigene Produkte, die digitalisiert oder zumindest digital unterstützt werden müssen, um wettbewerbsfähig zu bleiben. „Internet of Things“, „Augmented“ oder „Virtual Reality“ haben beispielsweise zu Anforderungsänderungen in der Softwareentwicklung beigetragen. Aber auch Geschäftsprozesse müssen sich letztendlich mehr auf Kundenbedürfnisse ausrichten, um so die Käuferfahrung möglichst positiv zu gestalten.

Hinzu kommen Anforderungen, die interner Natur sind und mit denen Unternehmen oft tagtäglich konfrontiert sind. Qualifizierte Mitarbeiterinnen und Mitarbeiter wünschen sich unter anderem effizientere Verfahren, die ihnen mehr Komfort bieten. Ein digitaler Ablauf beim Ausfüllen eines Urlaubsantrags geht zum Beispiel einfacher und schneller, als diesen manuell abzuwickeln. Außerdem müssen Unternehmen heute mehr denn je in der Lage sein, Ihre Prozesse schnell an die Bedürfnisse der Beschäftigten adaptieren zu können. Dies bedeutet auch, Ihnen benötigte Daten und Dokumente schnell zur Verfügung zu stellen. Alles in allem verschaffen sich Unternehmen mit der Optimierung und Digitalisierung von Prozessen einen Vorsprung gegenüber der Konkurrenz und damit einen wichtigen Wettbewerbsvorteil.

Zusätzlich machen weitläufigere Herausforderungen die Digitalisierungsarbeit in Unternehmen besonders schwer: Es gibt einen Mangel an Fachkräften, insbesondere an Programmierern. Daraus resultiert, dass diese wenig Zeit für die Fülle an Projekten haben. Somit dauert der „time to market“ für neue Prozesse und Apps oftmals zu lange. Ein weiteres Problem stellt oft die unzureichende Vernetzung von einzelnen Insellösungen im Unternehmen dar – verschiedene Fachabteilungen setzen auf unterschiedliche Systeme, die ohne Bezug zueinander und ohne Überwachung der IT auskommen müssen.

Der ständige Versuch, Anforderungen und Hindernisse gleichermaßen zu bewältigen, führt zu einem ewigen Kreislauf, der dem Unternehmen dauerhaft schaden kann, zu Unzufriedenheit führt und das Wachstum hemmt. Die Relevanz von Low-Code-Plattformen liegt somit auf der Hand: Sie fangen viele der Anforderungen und Herausforderungen auf und bieten damit einen Lösungsweg.

Anforderungen an Unternehmen heute



Low-Code ist der richtige Ansatz, weil...

... digitale Transformation es notwendig macht, Unternehmensanforderungen ständig neu zu definieren.

... Prozesse abteilungsübergreifend schnell angepasst werden sollten, um wettbewerbsfähig zu bleiben.

... Personalengpässe in IT-Unternehmen die Situation zusätzlich erschweren.

Ursprung, Bedeutung und Entwicklung von Low-Code

Der Ansatz ist schnell erklärt: Mit nur wenig Codieren lassen sich Aufgaben, Algorithmen und Anforderungen, zum Beispiel eines Prozesses, erstellen. Mittels einer grafischen Benutzeroberfläche können Anwender, ohne viel Programmierkenntnisse, auf einer Plattform Lösungen für die eigene Abteilung, ihre Arbeit oder gar das ganze Unternehmen definieren. Einfach ausgedrückt bedeutet das, dass Nutzer die Applikation grafisch modellieren und definieren, während im Hintergrund eine Plattform diese „Befehle“ verarbeitet. Diese Beschreibung macht deutlich, dass Anwender nur im geringen Maße Programmcode in Form von Skripten erstellen müssen.

Die ersten Verfahrensweisen für Low-Code-Technik gab es bereits in der 80er Jahren, mit sogenannten „Fourth-Generation-Languages“, kurz 4GL. Der Begriff stammt aus der Softwareentwicklung und beschreibt eine Programmiersprache, die sich hinsichtlich ihres Abstraktionsniveaus von Sprachen aus anderen Generationen unterscheidet. Momentan gibt es fünf verschiedene GL:

Momentan gibt es fünf verschiedene GL:

- 1GL: Maschinensprache
- 2GL: Assemblersprache
- 3GL: Prozedurale Programmierung
- 4GL: Deklarative Programmierung
- 5GL: Künstliche Intelligenz

Während die Maschinensprache (1GL) noch sehr an der Hardware orientiert war, zeichnet sich die vierte Generation durch möglichst simple Ansätze, also weg von der klassischen Maschinensprache, aus. Assemblersprache (2GL) ist ebenfalls ähnlich zur ersten Generation mit dem Unterschied, dass ein Assembler, ein sogenanntes Computerprogramm, Assemblercodes in Maschinensprache übersetzt. Die prozedurale Programmierung (3GL) ist heute noch gängig: Viele, auch nicht Programmierer, kennen bestimmt C# oder objektorientierte Sprachen wie C++, Java oder JavaScript. 3GL und 4GL haben unterschiedliche Zielsetzungen: Während bei der prozeduralen Programmierung die Entwicklung von standardisierten Kontrollstrukturen im Fokus steht, soll deklarative Programmierung (4GL) mit wenig Codezeilen schnell eine bestimmte Funktion oder eine komplette Anwendung abbilden.

Salonfähig ist der Begriff „Low-Code“ allerdings für diese besondere Art von Softwareentwicklung erst 2014 durch Forrester Research geworden³. Die Zielsetzung gleicht der von 4GL: Nutzer sollen auch ohne Programmierkenntnisse in der Lage sein, Prozesse und Anwendungen zu entwickeln – ganz ohne die Unterstützung von Experten, beispielsweise aus der IT.

Die Aussichten sind rosig: Nach einer aktuellen Studie planen unter 2438 befragten europäischen und amerikanischen Entwicklern rund 20 Prozent eine Low-Code-Anwendung in ihrem Unternehmen einzuführen. Weitere 20 Prozent haben solch eine Plattform bereits integriert. Zwar herrscht unter 60 Prozent der Befragten noch Skepsis und Tatenlosigkeit. Die Autoren gehen aber davon aus, dass die Zweifel mit der Zeit schwinden, da die Vorteile von Low-Code Entwickler und Führungskräfte mehr und mehr überzeugen werden. Die Ergebnisse lassen zusätzlich Rückschlüsse zu, welche Art von Unternehmen Low-Code praktizieren. Entgegen der Erwartungen sind es vor allem große und mittelständische Firmen, die durch diesen Ansatz profitieren⁷.

Vorteile von Low-Code-Entwicklung im Überblick:



Spart Zeit



Leichtes
Erlernen von
Know-how



Nachträgliche
Anpassung von
Anwendungen
möglich



Nur geringe
Programmier-
kenntnisse
notwendig



Tendenziell
niedrige
Kosten

Neue Ära von Entwicklern: Citizen Developer

Low-Code-Plattformen bringen durch ihre einfache Handhabung eine neue Ära von Entwicklern hervor - die sogenannten Citizen Developer. Hinter dieser Personengruppe verbergen sich Business User beziehungsweise Fachanwender oder Entwickler, für die die Effizienz und Einfachheit von Programmierung im Fokus steht. Es sind der Definition nach Personen, die Anwendungen für ihre Abteilung, das ganze Unternehmen oder sogar für die Öffentlichkeit im Stil der angesprochenen vierten Generation (4GL) und Cloud Computing-Diensten erstellen. Citizen Developer können, müssen aber nicht, über Programmierkenntnisse verfügen. Demnach steht das Verständnis über Logik und Zusammenhänge mehr im Vordergrund als das Wissen über Technik. Sie sind zudem in der Lage, Applikationen eigenständig zu gestalten, holen sich aber wenn nötig Hilfe bei der Einrichtung von Schnittstellen zu anderen Systemen wie SAP.

Es gibt zwei denkbare Art und Weisen, wie Citizen Developer in ihrem Unternehmen arbeiten. Die erste Variante besteht darin, unabhängig von der IT zu agieren. Sie werden in diesem Fall zum Beispiel in ihrer Abteilung für die Optimierung und Automatisierung von Abläufen einer Software eingesetzt. Erst bei schwerwiegenden Beeinträchtigungen ziehen sie IT-Spezialisten hinzu. Bei der zweiten Variante ist der Citizen Developer Teil der IT-Abteilung. Er bildet mit anderen Citizen Developern oder Entwicklern ein Team, das beispielsweise für Prozessautomatisierung im Unternehmen zuständig ist. Er hat somit direkten Kontakt zu IT-Profis, die ihn bei Bedarf unterstützen können. Damit sind seine Handlungsweisen auch nicht eingeschränkt. Dieses Modell wird vor allem in der Schulungsphase benutzt und hat zum Ziel, dass Citizen Developer gleichwertige Entwickler werden.



Citizen Developer

- haben einen 20 Prozent höheren Anteil an Frauen*
- sind im Durchschnitt 5 Jahre jünger*
- arbeiten häufiger in administrativen Abteilungen**
- sind mehr daran interessiert, den Umsatz eines Unternehmens zu steigern*
- haben ein höheres Interesse daran, die Customer Experience, Produkte und Service sowie die Innovationsfähigkeit ihres Unternehmens zu verbessern*
- haben sich eher zum Ziel gesetzt, den Einfluss auf den Markt sowie die Markenreichweite zu steigern

*verglichen mit befragten Entwicklern, die kein Low-Code nutzen
Basis: 3.228 global developers, 746 low-code adopters, 1.486 nonadopters

**Anteil von 20 Prozent bei allen Entwicklern und 30 Prozent bei Low-Code-Entwicklern
Basis: 3.111 global developers, 708 low-code adopters, 1.443 nonadopters

Quelle: John R. Rymer & Bill Seguin (2019). Forrester, Understanding Low-Code Developers.

Durch ihre Perspektive von außen, sind sie für Unternehmen von großem Vorteil: Citizen Developer wissen oft, an was es Nutzern fehlt und gehen Schwächen praxisrelevant an. Hilfreich ist, dass sie sich in den Anwender hineinversetzen können. Insbesondere Citizen Developer, die in Fachabteilungen tätig sind, kennen Kundenanforderungen und die darauf basierenden Prozesse deutlich besser als andere Akteure im Unternehmen. Außerdem erhöhen sie durch ihre Eigeninitiative die Produktivität eines Unternehmens, da sie zusätzliche Softwareentwicklung leisten. Daher sind Citizen Developer für Organisationen sehr wertvoll.

Low-Code ist nicht gleich No-Code

Wer sich mit dem Thema Low-Code beschäftigt, wird ebenfalls um den Begriff „No-Code“ nicht herumkommen. Damit sind Entwicklungen gemeint, für die überhaupt keine Codierung notwendig ist. Es handelt sich hierbei um eine Unterform von Low-Code, die vor allem für die eben vorgestellten Citizen Developer praktisch ist, da entsprechende Plattformen noch einmal deutlich einfacher aufgebaut sind als Low-Code-Plattformen. Apps und Prozesse können mit nur wenigen Klicks entworfen werden. Durch No-Code verlieren Anwendungen aber an Flexibilität: Durch die einfache und vordefinierte Bauweise, sind bei der No-Code-Variante nur eingeschränkt Prozesse abbildbar. Dementsprechend kommen Nutzer mit ihren Anwendungsmöglichkeiten an Grenzen, die sie nicht mehr überwinden können und die ein Upgrade auf eine andere Plattform erfordern. Das macht No-Code mitunter zu einer kostspieligen Sache. No-Code eignet sich daher nur für Unternehmen, die kleine Prozesse und Applikationen abbilden möchten. Durch die einfache Handhabung spricht der Ansatz ausschließlich Fachanwender, sogenannte Citizen Developer, an, die in der Regel bei ihrer täglichen Arbeit keine Berührungspunkte mit Coding haben.

Dahingegen sprechen Low-Code-Plattformen sowohl Fachanwender als auch Entwickler, die an schnellen Lösungen und weniger an der Technik im Hintergrund interessiert sind, an. Möglichst wenig Code, aber niemals kein Code lautet die Devise. Damit können Nutzer auch größere Prozesse und Applikationen abbilden – durch Low-Code gibt es die Möglichkeit, Prozesse individueller und auf weitere Anforderungen auszurichten.

Den Unterschied zwischen Low-Code und No-Code macht das folgende Beispiel deutlich: WordPress ist wohl einer der beliebtesten Content-Management-Systeme weltweit und basiert auf dem Low-Code-Ansatz. Mit WordPress können Nutzer relativ leicht anhand eines Baukastensystems und vorgefertigten Designs eine eigene Website erstellen. Gleichzeitig gibt es aber auch die Möglichkeit, in den Code einzugreifen, um so individuelle Webseitewünsche zu realisieren. Würde WordPress lediglich auf dem No-Code-Prinzip basieren, so wären keine persönlichen Interessen umsetzbar. Nutzer müssten sich somit ausschließlich mit vorgefertigten Bausteinen zufriedengeben.

Low-Code und No-Code im Vergleich

Low-Code

- ✓ Spricht Fachanwender und Entwickler an
- ✓ Möglichst wenig Code
- ✓ Für große Prozesse & Applikationen
- ✓ Keine Limits, da Erweiterung durch Programmierung möglich

No-Code

- ✓ Spricht ausschließlich Fachanwender an
- ✓ Keine Coding-Optionen
- ✓ Für kleine Prozesse & Applikationen
- ✓ Nur begrenzte Anwendungsmöglichkeiten

Merkmale von Low-Code

Low-Code ist vielseitig: Die dazugehörigen Plattformen haben unterschiedliche Schwerpunkte, zum Beispiel auf Prozessautomatisierung oder auf Content Management. Nichtsdestotrotz gibt es Merkmale, die auf alle zutreffen.

Visuelle Entwurfsmöglichkeit:

Low-Code-Plattform ermöglichen es, Geschäftsabläufe, sogenannte Workflows, inklusive Benutzerschnittstellen und Datenmodelle in einem Entwurf zu skizzieren. Auf diese Weise haben Anwender eine visuelle Projektgrafik, mit der sie anderen ihre Arbeitsprozesse oder ihr weiteres Vorgehen erklären können.

Automatisierte Datenbankintegration:

Low-Code-Plattformen überführen Datenmodelle in relationale Tabellen und SQL-Abfragen. Zusätzlich werden Daten aus externen Programmierschnittstellen automatisch in der Anwendung zur Verfügung gestellt.

Vollständiges Lifecycle-Management:

Mit einer Vielzahl von Optionen reduzieren Low-Code-Plattformen Probleme, die bei Bereitstellungen, Wartungen und Änderungen auftreten können. Besser entwickelte Low-Code-Plattformen bieten außerdem eine Bereitstellung per Mausklick, eine automatische Skalierung und Projektmanagement-Tools an.

Es gibt zusätzlich noch ein paar weitere Charakteristiken, die den Markt für Low-Code-Plattformen beschreiben: Er wird als zukunftssträchtig mit steigenden Umsätzen, hohen Wachstumsraten und steigender Akzeptanz bei Unternehmen eingestuft. Auch ist der Markt für Low-Code-Plattformen schon ausgereift, aber in sich noch fragmentiert. 2017 listete Forrester beispielsweise 67 verschiedene Anbieter – daneben existieren noch viele weitere, die aber entweder zu klein oder zu dezentralisiert organisiert sind, um für eine vollständige Abdeckung zu sorgen. Hinzukommen kommen regelmäßig neue Anbieter, die ihr Glück in Nischenbereichen und durch die Ausrichtung auf enge Anwendungsfälle, versuchen⁵.



Low-Code als Digitalisierungswerkzeug | Einsatzszenarien

Die Auswahl einer Low-Code-Plattform will gut überlegt sein. Oftmals ist es so, dass die Plattformen herstellergebunden sind. Sprich: Die damit entwickelten Anwendungen können nicht ohne weiteres in ein anderes System überführt werden – eine Export-Funktion ist nicht vorgesehen. Dadurch müssen sie mühsam im neuen System nachgebaut werden. Vor dem Kauf sollte deshalb eine Kosten-Nutzen-Abwägung stattfinden, nicht nur in Bezug auf die Portierbarkeit, sondern auch hinsichtlich der Konvertierung und Flexibilität.

Allerdings kommt es bei der Auswahl auch darauf an, was Unternehmen mit der Plattform erreichen möchten: Sollen damit Softwareprodukte/Apps entwickelt werden, die auch verkauft werden sollen oder ist die Plattform dazu da, (automatisierte) Prozessabläufe des Unternehmens unter Einbindung von Kunden und Lieferanten abzubilden? Letzteres führt beispielsweise dazu, dass die Plattform nicht nur Low-Code können sollte, sondern auch auf Prozessautomatisierung in Form von Business Process Management ausgerichtet ist.

Folgende Fragen können bei der Auswahl hilfreich sein:

1

Unterstützt die Plattform diejenigen Features und Integrationen, die wir für unsere Arbeit benötigen?

2

Gibt es eine Support-Funktion seitens des Herstellers und wird die Plattform stetig weiterentwickelt?

3

Lassen sich mit der Plattform alle vorgesehenen Anwendungen entwickeln und ist sie flexibel genug, um ebenfalls noch nicht absehbare Entwicklungswünsche zu entsprechen?

Es gibt gute Gründe, warum Low-Code und digitale Transformation oft in einem Atemzug genannt werden. Low-Code bahnt den digitalen Weg, weil der Ansatz schnell ist, weil er Ressourcen schont und weil sich Prozesse schon vorab visuell modellieren lassen. Die Einsatzmöglichkeiten von Low-Code reichen ins Unendliche. Ob im Finanzsektor, in der Industrie, in der Personalabteilung oder der Verwaltung, Arbeitsabläufe lassen sich dadurch effizien-

ter abbilden. Ein einfaches Beispiel: Die Bearbeitung von Dokumenten, wie zum Beispiel Rechnungen. Mit Low-Code lassen sich automatisierte Prozesse erstellen, in der Rechnungen nicht nur systemseitig eingescannt und ausgelesen, sondern auch gleich an die jeweiligen Stellen im Unternehmen zur Freigabe weitergeleitet werden. Zusätzlich lassen sich die Rechnungen in einem System verwalten – so wird Zeit und Tonnen von Papier gespart.

Schlusswort

Sie haben gesehen, dass sich hinter dem Trend Low-Code, ein Ansatz verbirgt, der viel Potential mit sich bringt und für Entlastung sorgen soll – hinsichtlich Zeit, Kosten und Personal. Nicht zuletzt lässt sich der Ansatz nutzen, um den wachsenden Anforderungen in Unternehmen die Stirn zu bieten. Diese Vorteile erkennen mittlerweile immer mehr Unternehmen und schreiten zur Tat. Daher gilt auch für Sie: Nutzen Sie Ihr gewonnenes Wissen und die Vorteile der Digitalisierung für sich.

Quellen

¹Christian Rentrop & Stephan Augsten (2018): Low-Code-Entwicklung – Ansatz und Tools. Online verfügbar unter <https://www.dev-insider.de/low-code-entwicklung-ansatz-und-tools-a-763889/>.

²Christian Rentrop & Stephan Augsten (2019): Weniger Code, mehr Effizienz in der Entwicklung. Online verfügbar unter <https://www.dev-insider.de/weniger-code-mehr-effizienz-in-der-entwicklung-a-811805/>.

³Clay Richardson & John R. Rymer (2014): Forrester, New Development Platforms Emerge For Customer-Facing Applications.

⁵John R. Rhymer (2017): Forrester, Vendor Landscape: A Fork In The Road For Low-Code Development Platforms.

⁶John R. Rhymer & Rob Koplowitz (2019): Forrester, The Forrester Wave™: Low-Code Development Platforms For AD&D Professionals, Q1 2019.

⁷John R. Rymer & Bill Seguin (2019): Forrester, The State Of Low-Code Platform Adoption, 2018.

⁸John R. Rymer & Bill Seguin (2019): Forrester, Understanding Low-Code Developers.

⁹Marketsandmarkets (2019): Low-Code Development Platform Market. Online verfügbar unter <https://www.marketsandmarkets.com/Market-Reports/low-code-development-platforms-market-103455110.html>.

¹⁰Martin Otten (2018): Low-Code: Vier Erfolgsfaktoren entscheiden. Online verfügbar unter <https://midrange.de/low-code-vier-erfolgsfaktoren-entscheiden-2/>.

¹¹Matthew Revell (2017): What Is a Low-Code Development Platform? Online verfügbar unter <https://www.outsystems.com/blog/low-code-development-platform.html>.

¹²René Jahn (2016): Low-Code Development Plattform. Online verfügbar unter <https://www.informatik-aktuell.de/entwicklung/programmiersprachen/low-code-development-plattform.html>.

¹³Stephan Augsten & zeroshope (2019): Was ist ein Citizen Developer? Online verfügbar unter <https://www.dev-insider.de/was-ist-ein-citizen-developer-a-827524/>.

¹⁴Thomas Peham (2018): Der nächste Trend in der Softwareentwicklung? Low-Code-Plattformen! Online verfügbar unter <https://t3n.de/news/low-code-plattformen-944783/>.

Über JobRouter

Die JobRouter AG ist ein global aktiver Anbieter von Lösungen zur digitalen Unternehmenstransformation und Hersteller einer branchenübergreifenden Low-Code-Digitalisierungsplattform zur Prozessautomatisierung für Kunden aller Unternehmensgrößen. Damit unterstützt JobRouter Unternehmen, ihren Workflow zu optimieren, ihre Daten und Dokumente effizient zu verwalten und ihre internen sowie externen Geschäftsprozesse zu digitalisieren. Die modular angelegten Bausteine der JobRouter®-Technologie erlauben es zudem, die Lösungen den Anforderungen entsprechend zu erweitern und individuell zu organisieren. Unternehmen erhalten so ein umfassendes und auf sie zugeschnittenes System mit einer Vielzahl agiler Schnittstellen.

Gemeinsam mit einem internationalen Netzwerk aus 170 Partnern und eigenen Niederlassungen in Großbritannien, der Türkei, Polen und den USA, liefert die JobRouter AG an mittlerweile über 1200 Kunden weltweit; darunter führende Unternehmen wie Euromaster, HARI-BO, HELMA Eigenheimbau, Hochschule Konstanz, Klöckner & Co., MEYER WERFT, S&G Automobil, Universität Bern oder ZF TRW.

Die JobRouter AG hat ihren Hauptsitz in Mannheim. Weitere Informationen stehen unter www.jobrouter.com bereit.

JobRouter AG

Besselstraße 26

68219 Mannheim

office@jobrouter.com

www.jobrouter.com

Social Media

twitter.com/jobrouter

facebook.com/jobrouter.workflow

linkedin.com/company/jobrouter-ag

youtube.com/JobRouter

xing.com/companies/jobrouterag

JobRouter Newsletter

go.jobrouter.com/newsletter