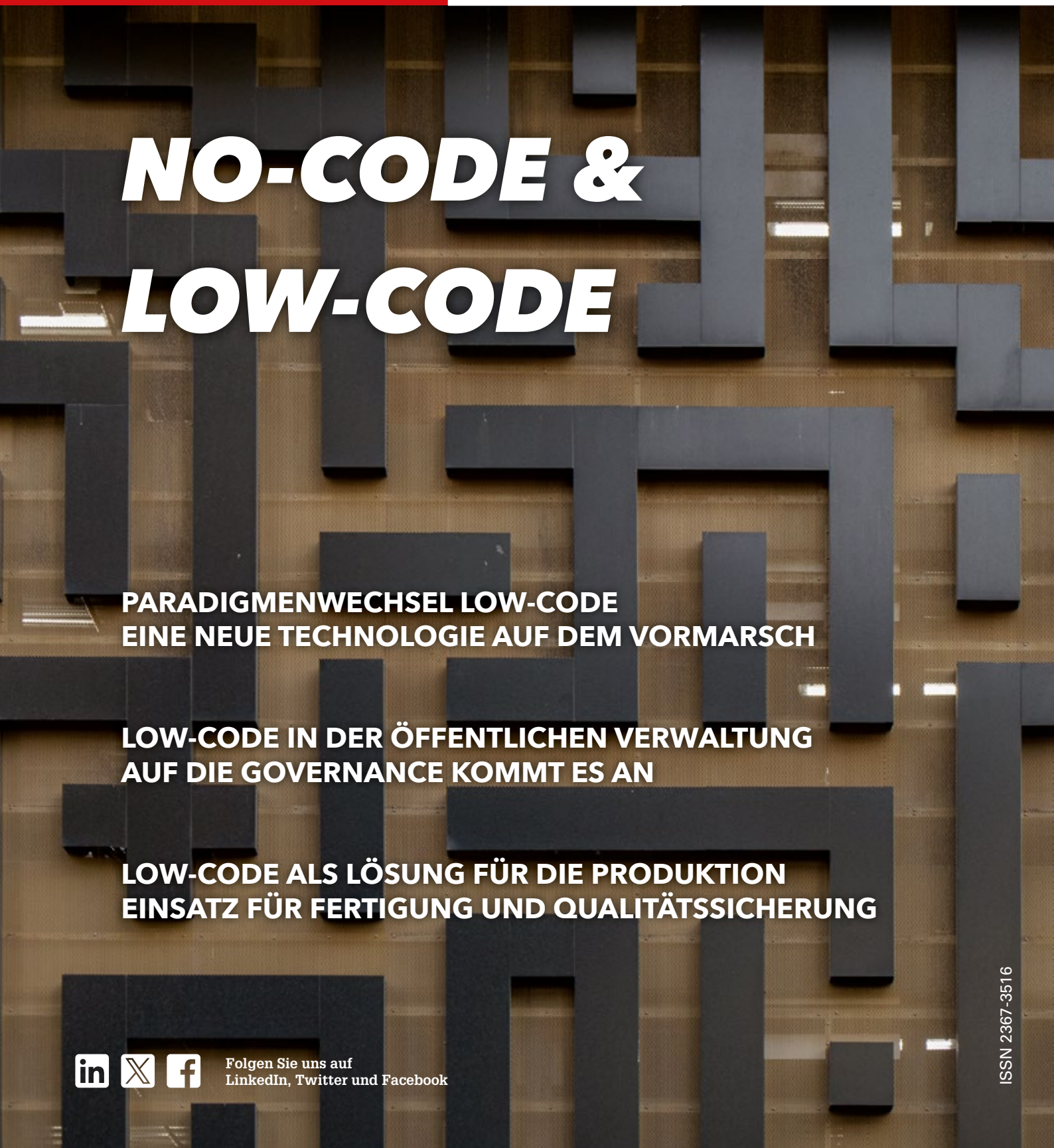
SQ MAGAZIN
ASQF®
Das Expertennetzwerk
Ausgabe 67 | Dezember 2023



NO-CODE & LOW-CODE

PARADIGMENWECHSEL LOW-CODE
EINE NEUE TECHNOLOGIE AUF DEM VORMARSCH

LOW-CODE IN DER ÖFFENTLICHEN VERWALTUNG
AUF DIE GOVERNANCE KOMMT ES AN

LOW-CODE ALS LÖSUNG FÜR DIE PRODUKTION
EINSATZ FÜR FERTIGUNG UND QUALITÄTSSICHERUNG

Folgen Sie uns auf
LinkedIn, Twitter und Facebook

ISSN 2367-3516

Der Arbeitskreis Softwarequalität & -Fortbildung e.V.

WÜNSCHT ALLEN EINE SCHÖNE WINTERZEIT

Fach- & Arbeitsgruppen

Fortbildungen

Events & Networking

ASQF net week

Days & Nights

ASQF®
Das Expertennetzwerk

www.asqf.de
+49 (0) 331 231 810 29
info@asqf.de

Liebe Leserinnen und Leser,

in Deutschland gibt es zu wenig Innovationsdynamik. Das war die zentrale Nachricht des Innovationsindikators 2023, der in diesem Jahr vom Bundesverband der Deutschen Industrie (BDI) veröffentlicht wurde. Demnach sei Deutschlands Wirtschaft zwar stabil, aber wenig dynamisch. Die Gründe hierfür sind vielfältig: Zum einen fehlt es noch immer an Wagniskapital-Investitionen, zum anderen mangelt es an qualifizierten Fachkräften. Darüber hinaus wird Innovation durch zusätzliche Aufgaben in wertschöpfungsfernen Prozessen, wie zum Beispiel Berichtspflichten behindert. Wie also können wir Innovationen fördern, gerade in der heutigen Zeit? Die wirtschaftlichen Aussichten für Deutschland sind derzeit gedämpft und deutsche Unternehmen stehen international unter Innovationsdruck, insbesondere im Bereich der Digitalisierung. Die Automotive-Branche, der Maschinen- und Anlagenbau sowie die Chemie- und Pharmaindustrie stehen beim Transformationsprozess vor den größten Herausforderungen. Aber auch der Handel, der Bereich der Dienstleistungen und die IT-Branche müssen verstärkte Anstrengungen unternehmen, um ihre wirtschaftliche Gesamtentwicklung zu stabilisieren.

Glaubt man den Vorhersagen von Gartner, könnten neue Technologien wie Low-Code und No-Code-Technologien einen entscheidenden Beitrag zur Lösung dieses Problems leisten. Benutzerfreundliche Drag-and-Drop-Plattformen, die es auch Nicht-Programmierern ermöglichen, Anwendungen zu entwickeln, könnten dem Fachkräftemangel entgegenwirken und Know-how aus verschiedenen Branchen einbringen, um Innovationen zu schaffen. Ist dies vielleicht die Lösung für unser Problem? Die Befürworter von Low-Code-Technologien sehen ihr Potenzial, insbesondere in Bereichen, in denen die Digitalisierung noch nicht weit fortgeschritten ist, wie beispielsweise in der Verwaltung. Gleichzeitig müssen wir aber auch die damit verbundenen Probleme und Fragen berücksichtigen: Wie steht es um die Qualität der entwickelten Software? Wer garantiert, dass die Lösungen den erwarteten Standards entsprechen? Werden wir zu abhängig von den Herstellern? Vereinfachen wir mit Low-Code wirklich alles oder schränken wir uns im Vergleich zur Individualentwicklung ein? Die Zukunft wird es zeigen.

In diesem SQ Magazin beleuchten wir die Vor- und Nachteile, Probleme und Chancen der Low-Code- und No-Code-Technologie und fragen uns, welche Rolle die Technologie in Zukunft übernehmen wird.

Im Artikel von Karsten Noack, dem Vorsitzenden der Deutschen Low-Code-Association, erhalten Sie eine Einführung zum Status Quo der Low-Code-Technologie. Kerstin Stier geht

auf die Potenziale von Low-Code in der Industrie ein. Maciej Dabrowski und Michael Thiele beleuchten in ihrem Artikel, was es für Voraussetzungen braucht, um Low-Code in der Verwaltung gewinnbringend einzusetzen. Im Interview mit Kim Lauenroth und Stan Bühne erfahren Sie, warum Requirements Engineering in Zeiten von Low-Code relevant bleibt. Und in der Kolumne von Richard Seidl lesen Sie eine kritische Betrachtung zur Softwarequalität im Kontext von Low-Code.

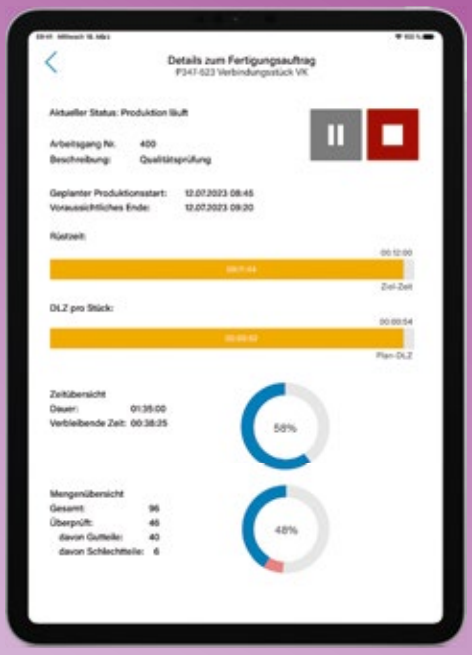
Ich wünsche Ihnen viel Freude mit diesem Heft!

Ihr Stephan Goericke

Was ist Ihre Meinung oder Erfahrung? Teilen Sie mir Ihre Gedanken mit! Ich freue mich auf Ihre Zuschrift und einen anregenden Austausch zu den verschiedensten Themen.
s.goericke@sq-magazin.de

13
ABSCHLUSSTREFFEN DER ARBEITSGRUPPE SECURE SOFTWARE ENGINEERING

35
DIE FUNKTION VON LOW-CODE AM BEISPIEL DER ERP-ENTWICKLUNG



31
LOW-CODE IN FERTIGUNG UND QUALITÄTSSICHERUNG

Inhalt

SCHWERPUNKT

5 PARADIGMENWECHSEL LOW-CODE

29 LOW-CODE: DIGITALISIERUNG EINFACH MACHEN

31 LOW-CODE IN FERTIGUNG UND QUALITÄTSSICHERUNG

40 DAS POTENZIAL VON LOW-CODE-PLATTFORMEN: ZWISCHEN MAGIE UND REALITÄT

BEST PRACTICE

9 LOW-CODE IN DER ÖFFENTLICHEN VERWALTUNG

35 DIE FUNKTION VON LOW-CODE AM BEISPIEL DER ERP-ENTWICKLUNG

IM FOKUS

17 DYNAMISCHES TESTEN MIT JUNIT 5

IM GESPRÄCH

38 REQUIREMENTS ENGINEERING IM ZEITALTER DER LOW-CODE-ENTWICKLUNG

KOLUMNE

41 LOW/NO-CODE – BRAUCHT NOCH WER `NEN TESTER?

Klimaneutral
Druckprodukt
ClimatePartner.com/11102-2312-1001

PARADIGMENWECHSEL LOW-CODE

AUTOR: **KARSTEN NOACK**

Low-Code gilt als eine der disruptivsten Technologien unserer Zeit. Analysten sagen voraus, dass Low-Code die klassische Individualentwicklung schon in wenigen Jahren nahezu vollständig ersetzen wird und die Anwendungsentwicklung wie kaum eine andere technische Neuerung auf den Kopf stellen wird. Was ist dran an diesen Voraussagen? Dieser Artikel soll die Transformationskraft von Low-Code beleuchten und sich der Frage widmen: Leitet Low-Code einen Paradigmenwechsel ein?

VON HIGH-CODE ZU LOW-CODE: DER "URKNALL" DER LOW-CODE TECHNOLOGIE

Die Idee, Software ohne oder fast ohne Programmcode zu schreiben, kam schon in den Neunziger Jahren auf, zunächst im Rahmen der damals als sehr aussichtsreich angesehenen, aber letztlich nicht erfolgreichen Bemühungen um ein Computer Aided Sofware Engineering (CASE). Das Berliner Unternehmen „Scopeland" (Anm. d. Red.: der Autor ist Gründer dieses Unternehmens) gab im Jahr 1998 die weltweit erste voll funktionsfähige Low-Code Plattform bekannt. Der „Urknall" der Low-Code-Entwicklung passierte im Jahr 2014 – ausgelöst durch eine griffige Begriffsdefinition von John Rymer, Analyst bei Forrester® Research. Er prägte den Begriff „Low-Code" und löste damit eine Welle an Transformationen aus: In den Folgejahren etablierten sich diverse Anbieter in den USA, deren unterschiedliche Konzepte und Arbeiten unter John Rymers Sammelbegriff „Low-Code" fielen. Rymer gelang es, das Potenzial der Entwicklung mit visuellen Applikationsdesign-Werkzeugen aufzuzeigen und dank einer intuitiv gut fassbaren Bezeichnung schlagartig bekannt zu machen. Schon kurze Zeit später schossen Low-Code-Startups wie Pilze aus dem Boden. Viele große Softwarehersteller merkten zudem, dass das eine oder andere ihrer Produkte auch halbwegs gut in diese Schublade passen würde, nannten es ebenfalls „Low-Code" und halfen dabei, dieser Entwicklung noch mehr Schub zu verleihen.

Inzwischen bescheinigen Gartner® und etliche andere US-Analysten, dass Low-Code die klassische Individualentwicklung binnen weniger Jahre nahezu vollständig ersetzen wird.

WAS GENAU IST LOW-CODE?

Low-Code steht für eine völlig neue Herangehensweise in der Softwareentwicklung. Es geht nicht nur darum, die Erstellung von Programmcode zu verbessern oder zu vereinfachen. Der Grundgedanke von Low-Code ist, überhaupt keinen individuellen Programmcode mehr zu schreiben bzw. wenn überhaupt, nur mit sehr geringem Aufwand. Dabei legt sich „Low-Code" nicht darauf fest, auf welche Art und Weise dies erreicht wird. Tatsächlich ist das heutige Angebot an Low-Code-Plattformen sehr breit gefächert. Es gibt datenbank- und prozessbezogene Low-Code-Technologien, ebenso wie interpretierende und code-generierende Systeme. Die einen bauen Software per Drag & Drop zusammen, während andere mehr auf deklarative Ausdrücke und Regeln (Business Rules) setzen. Schließlich laufen sowohl die Tools als auch die damit erstellten Anwendungen entweder in einer bestimmten Cloud oder sie werden on-premise in klassischen Rechenzentren installiert. Die Ausprägung der einzelnen Angebote richtet sich dabei nach der Zielgruppe, die sie bedienen.

EINE SALOMONISCHE LÖSUNG FÜR DAS PROBLEM DER ENDLICHKEIT

Der häufigste Kritikpunkt an der Low-Code-Idee: Es sei zwar überzeugend, wie schnell man ca. 98 % der benötigten Funktionalität zusammenklicken kann, aber an den verbleibenden 2 % würde der Ansatz letztlich scheitern: Es ist nicht zu vermeiden, dass es fast immer einige Anforderungen gibt, die mit rein visuellen oder deklarativen Methoden nicht erfüllbar sind. Wenn es hingegen gelingt, diese Anforderungen zu kapseln und dafür kleine Programmbausteine, Scripts oder Makros in die Low-Code-Anwendung einzubetten, dann ist das kein Show-Stopper mehr, sondern ein regulärer, planbarer Teil des Projekts.

Die Einbindung solcher Bausteine in die Low-Code Plattform erfolgt so, dass man die Anwendung jederzeit verändern kann, ohne das händisch Programmierte zu verlieren. Dies gilt insbesondere dann, wenn Scripte keine eigenen Daten- oder Objektemodelle benötigen und sich mit grundlegenden Programmierkenntnissen schreiben lassen.

WIE IST DAS BEI NO-CODE?

Die meisten sogenannten „No-Code"-Plattformen erheben übrigens nicht den Anspruch, auch noch die letzten

2 % visuell oder deklarativ umsetzen zu können. No-Code-Plattformen verstehen sich als eine besondere Ausprägung von Low-Code-Plattformen, die sich auf solche, meist einfachere Anwendungsfälle konzentrieren, bei denen die damit nicht abgedeckte Funktionalität verzichtbar ist.

Die enorme Vielfalt der Low-Code-Plattformen

Der Markt der Low-Code-Plattformen ist vielfältig. Jede Low-Code-Plattform setzt auf eine vorgefertigte Funktionalität. Da die Anforderungen an Software enorm vielfältig sind, kann kein Hersteller dieser Welt alle Aspekte abdecken. Deshalb lässt es sich nicht vermeiden, dass sich verschiedene Low-Code-Plattformen an bestimmten Arten von Anwendungsfällen oder Zielmärkten ausrichten. Beispielsweise bevorzugen Low-Code-Produkte, die sich auf den Public Sector, Banken oder Versicherungen konzentrieren, häufig codegenerierende und für heterogene IT-Infrastrukturen ausgelegte Architekturen, die in den kundeneigenen Rechenzentren installiert werden. Wer seinen Hauptmarkt im Mittelstand sieht, wird hingegen eher dazu neigen, alles serverfrei aus der Cloud heraus anzubieten. Bei größeren Wirtschaftsunternehmen und Verwaltungen muss hingegen entschieden werden, welche Art von Anwendungsfällen vorrangig adressiert und ob daten- oder prozessorientierte Lösungsansätze verwendet werden sollen.

Ebenso stellt sich die Frage, ob wirklich alle Funktionen mit Low-Code abgedeckt werden müssen: Viele Low-Code-Plattformen beschränken sich ganz bewusst auf (rein interaktive) Kernbereiche der Anwendungsentwicklung, um mit schlanken Ansätzen und leichter Erlernbarkeit zu punkten. Andere hingegen setzen darauf, auch für große und anspruchsvolle Projekte alles mit Low-Code umzusetzen, angefangen von intelligenter, toolgestützter Datenmodellierung, über automatische Prozesse und automatisierte Performanceoptimierung bis hin zur Generierung komplexer Output-Dokumente wie Fördermittelbescheide oder Projektstatusberichte. Wie bei anderen Tools auch, bilden sich hier Teilmärkte für einerseits günstigere Mainstream-Tools und andererseits umfassendere High-End-Toolsets heraus.

Die digitale Transformation findet in Unternehmen und Verwaltungen auf unterschiedlichen Ebenen und in verschiedenen Zusammenhängen statt. Für verschiedene Aufgaben können auch mehrere Low-Code- und No-Code-Plattformen gleichzeitig Einsatz finden.

Eine für alle?

Die eine, für alle Aufgabenstellungen ideale Low-Code-Plattform, gibt es nicht. Die Beschaffung von Low-Code-Plattformen bzw. deren Nutzung in der Cloud, kommt mit spezifischen Fachaufgaben und aus den Fachbereichen selbst heraus. Die Ausschreibungen dazu sollten keine Grundsatzentscheidung sein und nicht allein der IT-Abteilung überlassen werden.

Quelle: Low-Code Association e. V.

Dies mag für einige überraschend klingen, denn schließlich galt es immer, die Toolpalette zu reduzieren und zu konsolidieren. Das ist aber kein Widerspruch: Denn selbst wenn alle Anwendungen eines Unternehmens auf Basis einiger Low-Code-Plattformen laufen, dann ist das immer noch deutlich konsolidierter als mit 50 oder mehr Individualprogrammierungen zu arbeiten – selbst wenn diese alle die gleiche Programmiersprache nutzen. Low-Code konsolidiert auf einer anderen Ebene. Hierbei sollte man sich immer wieder klarmachen, dass es nichts Proprietäreres gibt als individuellen Programmcode. Diesen zu minimieren, dabei hilft Low-Code.

LOW-CODE DEVELOPER: EIN NEUER BERUF?

Das Aufkommen von Low-Code wird von dem Missverständnis begleitet, Low-Code sei etwas für „Citizen Developer", also für Fachanwender, die sich für den Eigenbedarf ihre Software einfach mal schnell selbst „zusammenklicken". Das hingegen war nie der vordergründige Sinn und Zweck von Low-Code. Low-Code adressiert die digitale Transformation als solche, in voller Breite und quer über alle Bereiche hinweg. Laut John Rymer betrifft Low-Code die Gesamtheit der Anwendungsentwicklung, mit Ausnahme eines schmalen Segments extrem komplexer und extrem sicherheitskritischer Spezialanwendungen. US-Analysten wie Gartner, Forrester und IDC sagen voraus, dass Low-Code in Zukunft immer verbreiteter werden und geschriebenen Code allmählich ablösen wird.

Folglich steht heute nicht mehr die Frage, ob das eine oder andere Entwicklungsvorhaben vielleicht auch mit Low-Code umsetzbar sei, sondern andersrum: Low-Code wird für unternehmensspezifische Anwendungssoftware die erste Wahl sein, und es braucht schon eine gute Begründung, warum im Ausnahmefall mal das eine oder andere Projekt nicht mit Low-Code umsetzbar sei.

WO BRAUCHT ES DANN NOCH PROGRAMMIERUNG?

Programmierung wird auch weiterhin erforderlich sein, etwa in der Entwicklung von Softwareprodukten, für sehr exotische Aufgabenstellungen, für Firmware aller Art und vieles auf Systemebene, aber eben kaum noch im Bereich der Anwendungsentwicklung. Die Anwendungsentwicklung wird zukünftig immer mehr in die Hände der Low-Code Developer fallen: Das sind ebenso professionelle Softwareentwickler, nur mit anderen Werkzeugen. Sie organisieren sich in Low-Code Developer Teams, die wahlweise in den Fachbereichen, in der IT oder irgendwo dazwischen angesiedelt werden können. In der Praxis findet man bereits heute alle drei Konstellationen.

Professionelle Low-Code Developer rekrutieren sich aus ehemaligen Citizen Developern und Anwendersystem-Betreuern, aus Business Developern wie beispielsweise Wirtschafts-, Bio- oder Geoinformatikern, Quereinsteiger:innen und „richtigen" Informatikern, die die weitaus höhere Produktivität und den direkten Kontakt zu den Anwender:innen zu schätzen wissen.

Typisch für Low-Code Developer ist eine ausgeprägte Kunden- bzw. Anwenderorientiertheit. Der ideale Low-Code Developer ist alles in einem: Requirements Engineer, Usability und User Experience (UUX) Experte, Front- und Backend-Entwickler, DevOps-Verantwortlicher, Qualitäts- und Projektmanager. Die Offenheit und das Breitenwissen des Universalisten schlagen die Tiefen-Expertise des Codierers. Aufgrund der breiteren Aufstellung der Low-Code-Entwickler können auch die Teams kleiner sein.

DAS WIRD SICH DURCH LOW-CODE VERÄNDERN

Was wird sich also durch Low-Code ändern? Alle bisher üblichen Entwicklerrollen werden neu definiert werden müssen. Auch das Ausbildungsprofil zukünftiger Softwareentwickler ist zu überdenken, weil ein Großteil von ihnen künftig als Low-Code Developer tätig sein wird, mit einem ganz anderen Anforderungsprofil.

Der Beruf der Softwareentwickler:innen nähert sich in gewisser Weise dem der Ingenieur:innen an. Softwarekünstler sind nicht mehr gefragt, sondern Pragmatiker: Leute, die sich gut in die betrieblichen Prozesse, Organisationsstrukturen und in die Befindlichkeiten der Anwender hineindenken können. Sie sollten zwar grundlegende IT-Prinzipien gut verstehen, müssen aber nicht mehr zwingend auf einem hohen Level programmieren können. Es hilft, Scripts und Makros auch schnell selbst schreiben zu können; wichtiger sind aber kommunikative Fähigkeiten, Projektmanagement- und UUX-Expertise.

Auch die Anforderungsaufnahme vereinfacht sich. Mockups könnten zukünftig gleich durch erste Entwürfe in der richtigen Software ersetzt werden. Indem Software durch Low-Code schrittweise und evolutionär verbessert werden kann, braucht es zudem vielleicht bald keine vorgeschalteten Prototypen mehr.

Manche sagen auch, dass es dank Low-Code kein SCRUM mehr braucht. Der strenge Sprint-Rhythmus von SCRUM dient dazu, die Entwicklerstränge zu synchronisieren. Ist das noch relevant, wenn jedes Ticket in kürzester Zeit erledigt werden kann? Auch hierzu liefert die Low-Code-Bewegung neue Antworten, etwa mit speziell auf Low-Code ausgerichteten Vorgehensmodellen, wie z. B. dem „Phasenagilen Vorgehensmodell", welches die Vorteile von klassischer, stringenter Projektsteuerung mit denen der agilen Softwareentwicklung in Einklang bringt.

Neu ist auch die Überlegung, die Prinzipien des „Design Thinkings" auf alle Projektphasen auszuweiten. Auf diese Weise werden die späteren Anwender besser in den gesamten Entwicklungsprozess eingebunden, was letztlich zu besseren Entwicklungsergebnissen und zu einem geringeren Dokumentations- und Schulungsbedarf führt.

Entwicklerdokumentation ist bei Low-Code-Anwendungen ohnehin weitgehend obsolet, weil die Konfiguration der Software ja im Wesentlichen selbstbeschreibend ist. Low-Code-Anwendungen dokumentieren sich selbst.

Auch die Qualitätssicherung kann sich vereinfachen, weil man bei der fast ausschließlichen Verwendung vorgefertigter Funktionalitäten von weniger Systemfehlern ausgehen kann. Und selbst Logikfehler können beim interaktiven Vorgehen im Projekt schneller bemerkt und sofort behoben werden.

AUCH DIE FRAGE „BUY-OR-BUILD" STELLT SICH NEU

Lange hatten IT-Strategien das Ziel, Individualsoftware wo immer möglich durch Standardsoftware zu ersetzen, aus nachvollziehbaren Gründen. Denn Individualprogrammierung bringt einige Herausforderungen mit sich: Angefangen von den Schwierigkeiten bei der Anforderungsdefinition und der Besetzung der wichtigsten Rollen im Entwicklungsprozess, über schwer kalkulierbare Projektlaufzeiten bis hin zur Verfügbarkeit der jeweiligen Programmierer:innen für spätere Anpassungen. All das sind Herausforderungen, mit denen der Low-Code-Ansatz weitaus besser umgehen kann als ein klassischer Programmieransatz.

Aufgrund wesentlich geringerer Projektkosten und kürzerer Projektlaufzeiten lohnt es sich, eine maßgeschneiderte Low-Code-Lösung selbst zu bauen oder von einem Dienstleister bauen zu lassen, anstatt sich in eine nur mäßig gut passende „Standardlösung" hineinzuzwingen. Individuelle Lösungen helfen den Unternehmen dabei, sich im Markt zu differenzieren und schneller auf Änderungen im Marktumfeld zu reagieren. Deshalb läutet die Umstellung auf Low-Code-Technologien gerade auch eine Trendumkehr ein. Dank Low-Code ist maßgeschneiderte Anwendungssoftware wieder attraktiv, wenn nicht sogar die bevorzugte Wahl. ■

DER AUTOR:

KARSTEN NOACK ist Vorstandsvorsitzender der Low-Code Association e. V., des Verbands der im deutschsprachigen Raum aktiven Low-Code-Anbieter. Er beschäftigt sich seit mehr als 30 Jahren mit der Frage, ob, wie und mit welchen Methoden und Technologien es möglich ist, anspruchsvolle Softwarelösungen ohne oder fast ohne manuelle Programmierung aufzusetzen. Als Gründer und CEO der Scopeland Technology GmbH, im Hauptvorstand der BITKOM, als Forumsprecher im SIBB (Forum Digital Transformation) und als Vorstand der Low-Code Association engagiert er sich dafür, dieser neuen Technologie immer breitere Einsatzmöglichkeiten zu erschließen.

LOW-CODE IN DER ÖFFENTLICHEN VERWALTUNG

AUF DIE GOVERNANCE KOMMT ES AN

AUTOREN: **MICHAEL THIELE** UND **MACIEJ DABROWSKI**

Die öffentliche Verwaltung in Deutschland setzt große Hoffnungen in Low-Code-Plattformen. Gerade auf Bundes- und Länderebene wollen Behörden ihre Anwendungsentwicklung verschlanken und IT-Verfahren beschleunigen. Das Ganze gelingt jedoch nicht ohne eine stringente Governance und Standardisierung.

Das Thema Low-Code im Public Sector nimmt derzeit Fahrt auf. Das Momentum könnte die CIOs in der Verwaltung und bei den öffentlichen IT-Dienstleistern durchaus kalt erwischen. Sorgfältig ausgearbeitete IT-Strategien und Vorgaben werden links liegen gelassen, und die Kraft des Faktischen schafft Ergebnisse. Diese Ergebnisse heißen oftmals Proof of Concept, Markterkundung oder Ausnahme und sind die langlebigen Provisorien, die sich später kaum noch mit vertretbarem Aufwand bereinigen lassen. Im Allgemeinen gilt für neue Plattformen oder Frameworks, die für fachliche Aufgaben eingesetzt werden, dass sie zwischen fünf und 15 Jahre in Betrieb bleiben.

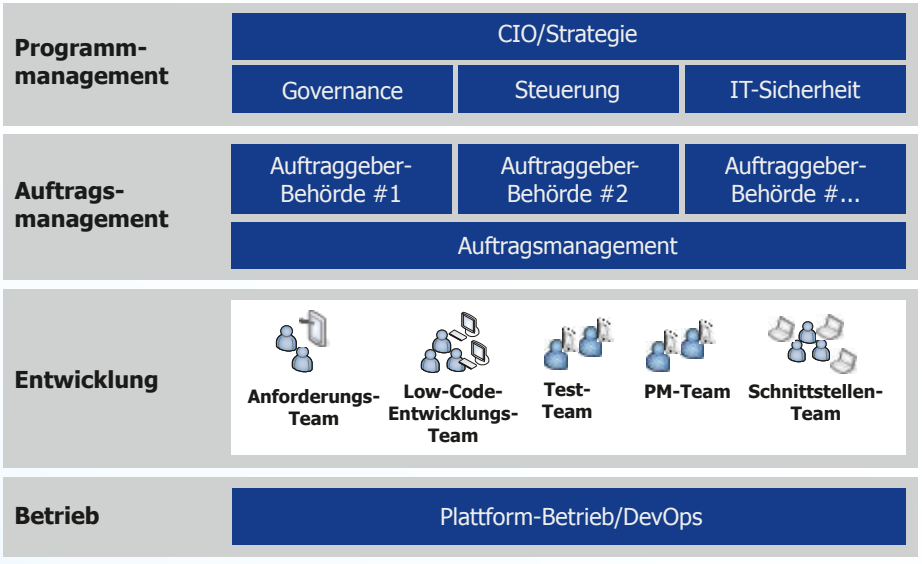
Auf der anderen Seite muss die IT darauf gefasst sein, vorausschauend alte Zöpfe abzuschneiden. Jeder kennt die IT-Dinosaurier, die nur deswegen überleben, weil ihre Entwickler oder Sponsoren von einst die Investition schützen oder das Gesicht wahren möchten. Daher sollte jegliche Einführung neuer Plattformen wohlüberlegt sein.

Klar ist: Die Einführung neuer Low-Code-Plattformen bietet Chancen zur Konsolidierung. Ohne gezielte Steuerung drohen aber IT-Wildwuchs und Dekonsolidierung. Und genau deshalb ist eine Governance mit Ableitungen für Strategie und operativen Betrieb so wichtig. Sie verhindert genau diese Negativeffekte.

CITIZEN DEVELOPER – SACHBEARBEITER ZU ENTWICKLERN?

Die öffentliche Hand ist aufgrund des harten Wettbewerbs und der demografischen Entwicklung besonders stark vom IT-Fachkräftemangel betroffen. Grundsätzlich ist der Ansatz nachvollziehbar, diesem Umstand mit dem bereits vorhandenen Personal entgegenzuwirken.

Die Krux ist: Das vorhandene Personal hat nur ein überschaubares Interesse an der aktiven Arbeit mit Low-Code-Plattformen. Es gibt offenbar



Quelle: Sopra Steria

Low-Code-Organigramm mit Teams, Auftraggeber und IT-Dienstleister

schon in den Stellenbeschreibungen Hürden – diese sehen Tätigkeiten für die Entwicklung und Anpassung von Low-Code-Anwendungen nicht vor. Darüber hinaus besteht in den Fachteams die Sorge, dass man eine Sonderaufgabe „IT-Entwicklung" nicht mehr los wird. Letztlich kommen noch Bedenken der Vorgesetzten dazu, dass das Know-how der Beschäftigten bei Versetzungen verloren geht.

Daher sind potenzielle Low-Code-Entwickler:innen in der öffentlichen Verwaltung voraussichtlich eher bei IT-affinen Junior-Developern und bei den internen öffentlichen oder deren externen Dienstleistern zu suchen und weniger bei den klassischen Sachbearbeiter:innen.

GOVERNANCE ALS LÖSUNGSANSATZ

Eine Governance beschreibt die notwendigen Rollen, Teamgrößen und Grenzen der operativen Teams. Sie legt die Zuständigkeiten und die Zusammenarbeit zwischen der auftraggebenden Behörde und dem IT-Dienstleister fest. Eine Governance nach Einführung und Etablierung von Low-Code-Plattformen erfordert im Endausbau ein hohes Maß an Arbeitsteilung:

Zusammengehalten wird die Governance durch ein übergreifendes **Programmmanagement**. Dieses definiert die Strategie und die wesentlichen Elemente sowie Prozesse der Governance und stellt deren Umsetzung sicher.

Das **Auftragsmanagement** sorgt an der Schnittstelle zu den Auftraggeber-Behörden für die Priorisierung und Koordination der einzelnen Teilaufgaben sowie die Ressourcenzuweisung und sorgt für eine Kontrolle der Qualität der Teamarbeit.

Die Teams in der **Entwicklung** unterteilen sich den Aufgaben entsprechend in klassisches Projektmanagement, Anforderungsmanagement, Low-Code-Entwicklung, Testen, Schnittstellen Testen und Schnittstellen.

- Die Aufgaben im Projektmanagement und Anforderungsmanagement unterscheiden sich nicht wesentlich von anderen IT-Projekten, die Teams arbeiten jedoch meist agil. Die Zuständigkeiten und Tätigkeiten der anderen Teams werden durch die Governance begleitet.

- Die Low-Code-Entwicklung orientiert sich primär an den fachlichen Vorgaben und wird durch das Auftragsmanagement je nach Zuständigkeit begrenzt. Ein großer Vorteil bei diesem Ansatz liegt in der Skalierungsmöglichkeit des Personals, da hier keine gestandenen Java- oder C++-Entwickler:innen benötigt werden.

- Testen, insbesondere automatisiertes Testen, ist eine wichtige Voraussetzung für effizientes DevOps oder klassisches Releasemanagement. Die Governance legt fest, in welchem Maße noch manuelles Testen stattfindet. Zudem regelt sie die Verantwortlichkeiten und wie die Testfälle definiert, implementiert und durchgeführt werden. Wer hier noch weitgehend ohne automatisierte Tests agiert und die Fachabteilungen als Tester „missbraucht", wird die Low-Code-Potenziale kaum heben.

- Das Schnittstellen-Team schafft wiederum die Voraussetzungen dafür, dass Aufgaben überhaupt end to end bewältigt werden können.

Der Betrieb übernimmt eine zentrale Aufgabe. Denn die für den Betrieb zuständigen Abteilungen der IT-Dienstleister im Public Sector agieren selten agil, und DevOps ist oftmals noch kein täglich Brot. Die Governance liefert hier unabdingbare Leitlinien dafür, die Prozesse umzustellen und zu beschleunigen, damit das alles nicht zu Lasten der Sicherheit geht. Maßstab sollte sein, ein getestetes Release nach 15 Minuten freizuschalten, statt einmal die Woche das Deployment anzustoßen. Ist DevOps noch Fiktion, braucht es ein geregeltes Releasemanagement. Der Betrieb muss zudem Skalierungsmöglichkeiten aufzeigen und die Performance im Sinne von Durchsatz, Nutzeranzahl, Latenz und Antwortzeiten belegen. Diese KPIs fließen wiederum in die Governance ein.

GRENZEN VON LOW-CODE

Die Herausforderungen bei der Entwicklung von IT-Verfahren entstehen sicherlich nicht beim Erstellen eines komplexeren Formulars mit einem ausgereiften Workflow. Anspruchsvoll ist eher die Einbettung in die gesamte heterogene, gewachsene IT der Verwaltungsebene sowie die Nachnutzung vorhandener Services wie E-Akte oder Payment. Zudem sind ein geordnetes Releasemanagement und der sichere Betrieb ebenfalls nicht zu unterschätzen.

Bausteine wie Prozesse oder Formulartemplates lassen sich prinzipiell wiederverwenden. Man sollte jedoch beachten, dass sich bereits fertige Verfahren, die nicht von Anfang an als „wiederverwendbar" gedacht und gebaut wurden, eben nicht aufwandsarm nachnutzen lassen. In welchem Maß die Verantwortlichen vorausdenken müssen, regelt wiederum die Governance. Wenn die IT-Landschaft steht, die Schnittstellen vorhanden sind und das Betriebsmodell festgelegt wurde, kann das die Nachnutzung von Low-Code-Komponenten obsolet machen, weil neue Verfahren für ähnliche Anwendungsfälle dann relativ aufwandsarm neu aufgesetzt werden können.

Low-Code-Plattformen sollten zudem nur eingeschränkt für die Implementierung von sehr komplexen Verfahren eingesetzt werden. So sollten das Haushalts-, Kassen- und Rechnungswesen (HKR) und das Personalmanagementsystem vielleicht nicht als erste Aufgabe angegangen

werden. Low-Code-Plattformen sind jedoch eine willkommene Ergänzung dieser Verfahren an deren Schnittstellen. Über die Governance lässt sich der Grad der Komplexität festlegen, ohne die immer gleichen Diskussionen führen zu müssen.

LOW-CODE-PLATTFORMEN UND HERSTELLERABHÄNGIGKEIT – MAKE OR BUY?

Wenn die Ressourcen und das Personal begrenzt sind, muss sehr genau überlegt werden, unter welchen Voraussetzungen die Entwicklung „eigener" Low-Code-Plattformen verfolgt und durchgehalten werden kann. Selbstentwickelte Plattformen, z. B. auf der Basis von Open Source, erscheinen immer dann sinnvoll, wenn spezielle Aufgaben effizienter und kostengünstiger bewältigt werden können als mit einer Kaufsoftware. Wenn hinter den „Eigenentwicklungen" dann doch nur wieder externe Softwareentwickler:innen stecken, die Exklusivwissen aufbauen, oder die Lösungen aufgrund des Tool-Stacks eher exotisch sind, wird die digitale Souveränität unter Umständen sogar konterkariert.

Im Zweifel agiert der Markt schneller. Die in vielen aktuellen Plattformen bereits integrierten Funktionen, beispielsweise DevOps-Prozesse, vorbereitete Schnittstellen oder die Erzeugung nativer Apps für iOS- und Android-Geräte, zeigen exemplarisch, wo die Messlatte für Eigenentwicklungen heute angesetzt werden muss.

Eine Multi-Vendor-Strategie kann ebenfalls zum Ziel führen. Dies bedeutet zwar die Beschaffung mehrerer Plattformen, was die Anwendung eher auf Bundes- und Landesebene einschränkt, allerdings reduziert sich auch die Herstellerabhängigkeit. Das lohnt sich selbstredend nur, wenn auf allen Plattformen genügend Verfahren umgesetzt werden. Idealerweise werden dann Low-Code-Plattformen mit speziellen Stärken beschafft, z. B. bei Formularen, Workflows oder Schnittstellen.

Für ein Betreibermodell kommen drei Hauptvarianten in Betracht, die nicht jeder Hersteller anbietet:

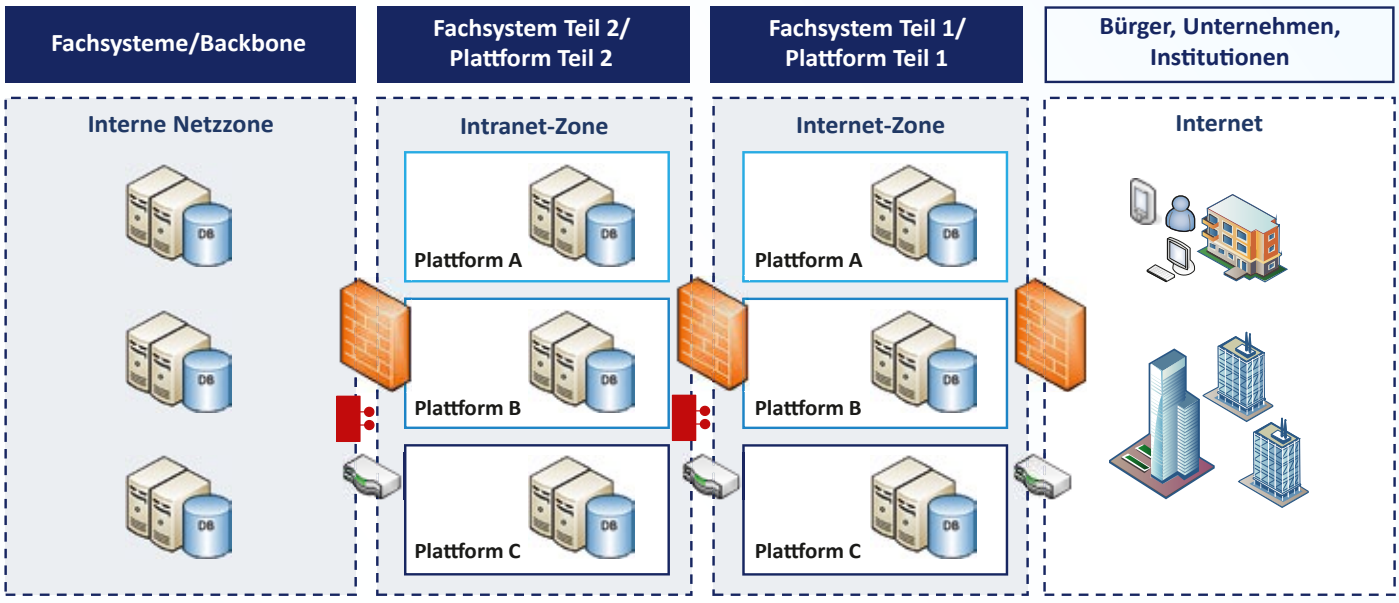
- On-Premise – im eigenen Rechenzentrum des jeweils gesetzten Dienstleisters der öffentlichen Hand
- Private Cloud – Cloud des Herstellers, aber speziell für nur einen Kunden; der Begriff „private" Cloud darf hierbei durchaus kritisch gesehen werden
- Hersteller-Cloud – Kunden sind durch Prozesse, Rechte und Rollen getrennt

Bei Anbietern außerhalb der EU muss die Entwicklung von Safe Harbour, Privacy Shield bis zum EU-US Data Privacy Framework sorgfältig beobachtet werden und die richtige Wahl des Betreibermodells vorausschauend erfolgen. Die Sicherheitseinschätzungen fließen in die Governance ein.

WEITERE HERAUSFORDERUNGEN

Leider beseitigen ein neues Framework, eine neue Plattform und ein neues Tool per se nicht alle Herausforderungen gewachsener Strukturen. Dazu gehören z. B. unklare Anforderungen und der Einsatz wenig erfahrener Projektmanager auf allen Seiten.

Effizienz kann in der IT häufig nur durch Standardisierung und Automatisierung erreicht werden. Leider wird häufig schon der individuelle, unkoordinierte Einsatz ein und derselben Plattform als Erfolg gewertet – eine typische Falle für Entscheider:innen. Die Governance regelt in diesem Fall die Anforderungen an die Tiefe einer Standardisierung und die Freiheitsgrade der Auftraggeber.



BU: Architekturübersicht mit drei Low-Code-Plattformen, Diensten und Internetanbindung (Quelle: Sopra Steria)

In der öffentlichen Verwaltung bilden Gesetze, Verordnungen und Rechtsvorschriften den groben Rahmen. Ein gewünschtes sinnvolles, einheitliches Erscheinungsbild auf Bundes- oder Landesebene sowie die Grundsätze zur Kostensparsamkeit setzen Limits für die Individualisierung. Die Governance sollte eine Rechtsauffassung adaptieren und dokumentieren, die zum Beispiel mittels Rechtsgutachten erarbeitet wurde. In Fragen der Schutzbedarfe und Mandantentrennung ist das BSI eine gute Adresse und sollte bei Bedarf vorausschauend und transparent eingebunden werden.

Die Governance regelt ebenfalls die Zusammenarbeit an der Schnittstelle zwischen Behörde und IT-Dienstleister. Die teilweise ernüchternden Ergebnisse der Anwendung agiler Ansätze in vielen Projekten der vergangenen Jahre legen nahe, genau an dieser Schnittstelle wieder mehr auf das V-Modell XT zu setzen. Die agile Umsetzung bleibt dabei völlig unbenommen. Die Governance setzt hier Leitplanken für eine erfolgreiche Umsetzung.

Abschließend müssen die bereits vorhandenen IT-Strategien sowie die daraus abgeleiteten Architekturrichtlinien um ein Low-Code-Kapitel erweitert und verzahnt werden.

FAZIT

Eine leichtgewichtige, aber verbindliche Governance ist unabdingbar, um IT-Wildwuchs zu verhindern. Das Regelwerk selbst sollte dabei nicht sämtliche operativen Aspekte en détail regeln. Es muss Ziele aufzeigen, Grenzen setzen und diese mit klaren, einzuhaltenden und überprüfbaren Vorgaben untermauern. Außerdem regelt die Governance Aspekte der Rollenaufteilung, des Betreibermodells, der Programmsteuerung, der Dienstleistersteuerung sowie des Releasemanagements und bietet Input für die Anpassung der IT-Strategie und der Architekturrichtlinien.

Mit diesen Leitplanken schaffen Verwaltungen die absolut notwendigen Voraussetzungen, um Mehrwerte durch Low-Code-Strategien zu erzielen. ■

DIE AUTOREN:

MICHAEL THIELE ist seit 2001 Berater bei Sopra Steria im Umfeld Public mit den Schwerpunkten IT-Strategien, IT-Architekturen und Leitung „kritischer" Projekte. Er berät Länder und den Bund bei der Auswahl und Positionierung neuer Technologieansätze und Organisationsstrukturen. Des Weiteren wird er bei der Standortbestimmung kritischer Projektsituationen und der Überführung solcher Projekte in ruhige Fahrwasser aktiv. Michael Thiele hat an der RWTH Aachen Elektrotechnik (Diplom) studiert und gründete im Anschluss eine kleine Beratungsgesellschaft. Er hat zwei Kinder und wohnt in Köln.

MACIEJ DABROWSKI berät die öffentliche Verwaltung seit 1997 in Strategie- und Organisationsfragen. Unter anderem berät er das Bundesministerium des Innern bei der IT-Dienste-Konsolidierung des Bundes in Architektur- und Beschaffungsthemen. Mangels Entwicklungskapazität hat er in seiner Karriere zahlreiche Problemstellungen mit Hilfe von Low-Code-Ansätzen gelöst. Maciej Dabrowski hat an der Ruhr-Universität Bochum Sozialwissenschaften studiert und gründete im Anschluss eine kleine Beratungsgesellschaft. Er hat vier Kinder und wohnt in Hamburg.

ABSCHLUSSTREFFEN DER ARBEITSGRUPPE SECURE SOFTWARE ENGINEERING

Die Arbeitsgruppe für Secure Software Engineering (SSE) traf sich zur Finalisierung der Fortbildung am Jahresende in Nürnberg. Bei dem Treffen in den Räumen der iSyst ging es nicht nur um die Zusammenfassung und die Arbeit an dem Trainingsschema. Viele Mitglieder freuten sich, sich nach langer vorwiegend digitaler Zusammenarbeit auch persönlich zu treffen.

Zunächst führte Gastgeber Sebastian Dengler von der iSyst GmbH durch die Räumlichkeiten des Nürnberger Standortes, vielen Dank an dieser Stelle an die iSyst für die Gastfreundschaft. Des Weiteren nahmen an dem Treffen Vera Gebhardt (Tecmata GmbH), Max Perner (infoteam AG), Prof. Jürgen Mottok (OTH Regensburg), Thomas Fehlmann (euro project office AG) teil. Der ASQF wurde von Marcel Schwarzmeier und Vizepräsident Norbert Kastner vertreten.

In der Sitzung der Arbeitsgruppe ging es nach einer kurzen Einführung und Vorstellungsrunde zunächst um eine gesamte Zusammenfassung und die Vermarktung des Ausbildungsschemas. Inhalte des Lehrplans sind u. a. Security Testing, Bedrohungsanalysen und -Anforderungen. Einig war sich die Gruppe, dass der SSE Kurs vor allem für Entwicklungs- und Testteams, Qualitätsbeauftragte und Projektmanager:innen interessant und von großem Vorteil ist. Auch unterhielt die Gruppe sich über Trainingsmethoden und wie Trainingsprovider und Hochschulen den Kurs anbieten können. Des Weiteren war das Erstellen eines Lehrbuchs basierend auf dem SSE-Lehrplan im Gespräch.

Die Arbeitsgruppe traf sich aber nicht nur zum Arbeiten. Zum Abschluss und als Dankeschön lud der ASQF e. V. alle Mitglieder zu einer gemütlichen Feier und einem gemeinsamen Abendessen in einen Nürnberger Gasthof ein. Bei leckeren regionalen Speisen und Getränken wurde die Arbeit und Leistung der SSE-Arbeitsgruppe gewürdigt.

Das SSE Weiterbildungsschema ist in deutscher und englischer Sprache verfügbar. Wie bei allen Schemen des ASQF erhalten die akkreditierten Trainingsprovider und Hochschulen den kompletten Foliensatz um sofort eine Schulung durchführen zu können. Die ersten Trainings wurden bereits durchgeführt. Die Prüfungen, auf Deutsch und Englisch, werden in 2024 bei den Zertifizierungsinstituten verfügbar sein.

https://www.asqf.de/fortbildungen/sse

Die Inhalte des Schemas und den Syllabus gibt es auf der ASQF Webseite: ASQF® Secure Software Engineering (SSE) https://www.asqf.de/fortbildungen/sse

Wer Interesse hat, den Kurs zu belegen oder Trainingsprovider werden möchte, kann sich direkt an Marcel Schwarzmeier (E-Mail: Marcel.Schwarzmeier@asqf.de) wenden. ■

ASQF TESTING DAY NRW 2024

Die Testing Community trifft sich seit vielen Jahren zum Testing Day NRW. Hat Corona uns noch kurz gestoppt, so stehen uns jetzt wieder die Möglichkeiten des direkten Austausches und des Networkings bereit. Wir freuen uns auf eine rege Teilnahme vor Ort beim ASQF Testing Day am 20. Juni in Düsseldorf.

Die Themenvielfalt um die Software-Qualitätssicherung wird immer größer. Nachdem agile Software-Entwicklung immer noch spannend, aber inzwischen ein „alter Hut“ ist, werden Themen wie Security wichtiger und neue Themen wie KI werden omnipräsent.

CALL FOR PAPERS

Wir möchten alle Interessenten Aufrufen uns einen kurzen Abstract zu diesen oder weiteren spannenden Themen, wie Testautomatisierung, Testorganisation, Testen im CI/CD, Social Skills im Testen, Schulungskonzepte, Testmethoden, Gamification etc. zuzusenden. Die Beiträge sollten einen Zeitslot von ca. 30-45 Minuten abdecken. ■

Deadline zur Einreichung von Beiträgen: 28.02.2024 per Mail an info@asqf.de

EXKLUSIVE RABATTE

FÜR ASQF-MITGLIEDER

Software Quality Days 2024

24. – 25. April 2024, Wien
https://www.software-quality-days.com/
20 % RABATT AUF KONFERENZTICKETS FÜR ASQF-MITGLIEDER

REConf 2024

22. – 26. April 2024, München
https://www.hood-group.com/reconf/
10 % RABATT AUF EARLY-BIRD-KONFERENZTICKETS FÜR ASQF-MITGLIEDER

Bei Interesse melden Sie sich bitte unter events@asqf.de

Trainingsprovider Termine

	Ort	Datum	Tage	Anbieter
CPPM	Berlin	16.01.2024	4	HTG Projektmanagement
CPPM	Wismar	12.02.2024	4	HTG Projektmanagement
CPPM	München	25.03.2024	4	HTG Projektmanagement
CPPM	Online	auf Anfrage	4	imbus AG
CPPM	Online	auf Anfrage	4	expleo group
CPPM	Online	22./23.02. und 26./27.02.2024	4	sepp.med GmbH
CPIoT	Online	19.-21.03.2024	3	sepp.med GmbH
CPIoT	Online	auf Anfrage	3	oose.de

ASQF® Certified Professional for Project Management

ASQF® Certified Professional for IoT

MEHR INFORMATIONEN ZU DEN ZERTIFIZIERUNGEN FINDEN SIE UNTER WWW.ASQF.DE/FORTBILDUNGEN

ASQF MITGLIEDERVERSAMMLUNG 2023

Mit der diesjährigen Mitgliederversammlung wurde das ASQF-Präsidium in seiner Arbeit bestätigt. ASQF-Präsident Ludger Meyer hieß in Potsdam wieder- und neugewählte Mitstreiter:innen willkommen.

Vizepräsidenten sind Karin Vosseberg und Norbert Kastner. Beisitzer sind Manfred Baumgartner, Vera Gebhardt, Joachim Hornegger, Maria Petzold, Alex Schladebeck, Bernhard Sechser und Felix Winter.

Ein großes Dankeschön ging außerdem an Nadine Wiechatzek und Andreas Spillner, die nicht erneut antraten. ■

Das ASQF-Stellenportal

Wir haben den passenden Job für Sie!

Parship GROUP

QUALITY ASSURANCE MANAGER (M/W/D)
Standort: Hamburg

Qytera

(JUNIOR) AGILE SOFTWARE TESTER/
AGILE JAVA TEST ENGINEER (M/W/D)
Standort: Eschborn (Frankfurt/Main)

sogeti Part of Capgemini

SENIOR CONSULTANT TESTMANAGEMENT SAP/
TESTMANAGER SAP (W/M/D)

TRAINER IT-QUALITÄTSSICHERUNG (W/M/D)

LAST- UND PERFORMANCE TESTER (W/M/D)

JUNIOR TESTMANAGER/TESTKOORDINATOR (W/M/D)

JUNIOR CONSULTANT (D/W/M)
SOFTWARE TESTING UND QUALITÄTSSICHERUNG

SENIOR CONSULTANT TESTMANAGEMENT/
TESTMANAGER (W/M/D)

TESTAUTOMATION EXPERT (W/M/D)

QUALITY MANAGER IT (W/M/D)

AGILE SOFTWARE TESTER (M/W/D)
Standorte: Ratingen, Frankfurt am Main, Stuttgart, München, Hamburg, Nürnberg, Dortmund

Neue Mitglieder im ASQF e.V.

Prosper Consulting GmbH
https://www.prosperconsulting.de/

viadee Unternehmensberatung AG
https://www.viadee.de/

QESTIT GmbH
https://qestit.com/

WERDEN SIE MITGLIED IM ASQF!
www.asqf.de/mitglied-werden

DYNAMISCHES TESTEN MIT JUNIT 5

AUTOR: **FREDERIC UNTERDÖRFER**

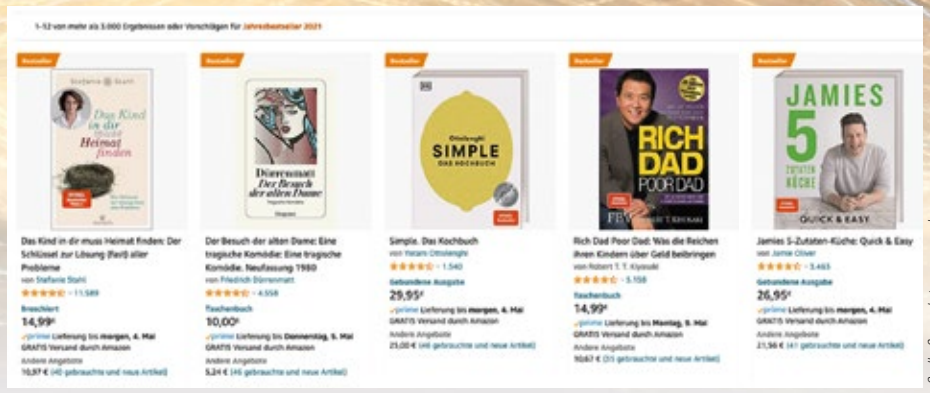
Quelle: Foto von Maxime VALCARCE auf Unsplash

Moderne Websites sind heutzutage deutlich dynamischer als ihre Vorgänger vor 20 Jahren. News-Seiten, Blogs oder Produktseiten führen meist viele Aktualisierungen pro Tag durch. Bisherige Testansätze kommen in Anbetracht der großen Flut an Inhalten schnell an ihre Grenzen. Diese Anleitung zeigt, wie man mit JUnit5 und dessen neuen dynamischen Tests seine Webseite beherrschen kann.

WARUM BRAUCHEN WIR DYNAMISCHE TESTS?

Um die Problematik zu verdeutlichen, wollen wir uns eines der dynamischsten Beispiele aus dem E-Commerce-Bereich anschauen: den Webshop. Bei den Giganten der Branche wie Amazon oder Ebay kommen sekündlich neue Produkte in den Shop, alte Produkte werden entfernt und bestehende Produktseiten werden aktualisiert. Jede einzelne Produktseite ist dabei wichtig und muss gewissen Anforderungen entsprechen. Ein fehlendes Bild oder ein falscher Preis kann hierbei schnell zu entgangenem Umsatz führen. In einer idealen Welt würde man also jede Produktseite am besten automatisiert abtesten.

In der Realität steht jedoch eine große Anzahl von Produktseiten einer kleinen Anzahl an Entwickler:innen gegenüber. Auch wenn sich große Teile des Testcodes sicherlich wiederverwenden lassen, ist der Aufwand für das Erstellen und Warten der Tests, die jeweils nur eine spezifische Produktseite testen, immens. Deswegen werden oft nur einzelne Produkte als Stichprobe mit einem solchen Test überprüft. So lassen sich zwar seitenübergreifende Fehler entdecken, spezifische Fehler für einzelne Produktseiten wie beispielsweise Rechtschreibfehler oder fehlende Bilder bleiben jedoch unerkannt.



Quelle: Screenshot von amazon.de

Wer alle Produktseiten vollständig testen möchte, hat viel zu tun.

Eine mögliche Lösung für dieses Problem ist, dem Test mehr Intelligenz zu verleihen und eine Art „Mini-Crawler“ zu schreiben. Ausgehend von einem beliebigen Startpunkt würde dieser Crawler alle Links, die er auf der Seite finden kann, einer Liste hinzufügen. Im nächsten Schritt würde er zum ersten Link aus dieser Liste navigieren. Ist der Link eine Produktseite, führt der Crawler einen Test auf Korrektheit aus. Ist es keine Produktseite, sucht er nach Links, die bisher noch nicht gefunden wurden und fügt diese der Liste hinzu.

Auf diese Weise werden alle vom Startpunkt aus erreichbaren Produktseiten nach und nach aufgerufen und abgetestet. Sind alle Produktseiten in Ordnung, ist der Test erfolgreich. Doch was soll passieren, wenn eine oder mehrere Produktseiten fehlerhaft sind? Bricht der Test schon auf der ersten fehlerhaften Seite ab, fehlen unter Umständen viele Seiten und der Test muss wiederholt werden. Sammelt der Tests alle Fehler zusammen, müssen die Entwickler:innen dafür sorgen, dass am Ende des Tests alle Fehler gut lesbar berichtet werden.

Hinzu kommt, dass bei hunderten bis tausenden Produkten die Laufzeit eines solchen Tests sehr hoch ist. Man würde das Ganze also idealerweise noch parallelisieren wollen, was zusätzlichen Aufwand und Fehlerquellen bedeutet.

Im Idealfall wollen wir die Parallelisierbarkeit und das Error Reporting von einzelnen Tests mit dem dynamischen Suchen der Testfälle kombinieren. An dieser Stelle kommt das dynamische Testen von JUnit 5 ins Spiel. Dieser Testtyp besteht aus zwei Phasen. In der ersten sogenannten TestFactory Phase werden die späteren Tests zusammengebaut. Ein Test besteht dabei aus einem Namen und einer Executable. Diese Executable beinhaltet den eigentlichen Testcode und funktioniert dabei wie jeder andere JUnit Test. In der zweiten Phase, der Testphase, werden die Tests aus der ersten Phase durchgeführt, auf Wunsch sogar parallel. Schlägt ein Test fehl, wird dieser einzeln als fehlgeschlagen gemeldet, alle anderen Tests laufen trotzdem weiter. Wie genau das Ganze funktioniert, wollen wir uns anhand eines Beispiels anschauen. Doch zunächst wollen wir einen kurzen Blick auf JUnit 5 werfen.

WAS IST JUNIT 5?

Das JUnit Testframework ist eines der populärsten Testframeworks für Java und gehört zum Standardrepertoire der Java Entwicklung. Mit JUnit 5 hat sich das Entwicklerteam entschieden, den Code komplett neu zu schreiben, um einige Altlasten des mittlerweile über 20 Jahre alten Frameworks loszuwerden.

Als Ergebnis kamen mit Version 5 viele neue Features hinzu, die das Entwickeln von Unit-Tests deutlich effizienter machen. So können beispielsweise Tests parametrisiert werden, um unnötiges Copy-and-paste zu vermeiden. Zusätzlich lassen sich Tests mühelos aktivieren oder deaktivieren, indem sogenannte Execution Conditions definiert werden, die zur Laufzeit ausgewertet werden. Nicht zuletzt ermöglicht JUnit 5 die parallele Ausführung von Testsuiten, was in verschiedenen Szenarien erheblich kürzere Ausführungszeiten zur Folge haben kann.

Für unser Beispiel ist vor allen Dingen das Feature „Dynamische Tests“ interessant. Bei dieser Testart schreiben wir eine Methode, die als Ergebnis eine Liste von Tests zurückgibt. Diese Tests werden dann wie normale JUnit Tests ausgeführt. Der große Vorteil: Wie die Tests aussehen, können wir dynamisch zur Laufzeit entscheiden!

WIR BAUEN EINE TESTFABRIK

Das Herzstück der dynamischen Tests ist die @Testfactory-Methode. In dieser Methode werden die einzelnen Tests zusammengebaut, als Liste gesammelt und an das JUnit Framework zurückgegeben. Wir beginnen an dieser Stelle mit der Erstellung eines lokalen ChromeDrivers und navigieren zu unserer lokalen OpenCarts-Instanz. Wir nutzen die Methoden der von uns gebauten PageObjects, um zur Übersichtsseite der Notebooks zu navigieren.

```
1 @TestFactory
2 public Collection<DynamicTest> TestNotebookCatgeory () {
3   //Starte einen ChromeDriver
4   ChromeDriver driver = new ChromeDriver();
5   //Navigiere zur Laptopübersichtseite
6   driver.get("http://localhost:80");
7   HomePage home = new HomePage(driver);
8   NotebookOverview notebookOverview = home.navigateToLaptops();
9   ...
```

Quelle: Testfabrik bei allen Code-Bildern vermerken

Als Nächstes müssen wir alle Produkte auf der Übersichtsseite finden und den Namen sowie den Link zur eigentlichen Produktseite extrahieren. Mit diesen Informationen können wir nun dynamisch den Testcode der einzelnen Produktseiten zusammenbauen. Hierfür bauen wir für jedes Produkt eine Executable, die den gewünschten Testcode beinhaltet. In unserem Fall bauen wir einen neuen ChromeDriver, navigieren zu den Produktseiten und überprüfen, ob das große Hauptbild existiert und ob es eine Beschreibung gibt. Ihr Testcode kann an dieser Stelle genauso komplex ausfallen wie jeder andere Selenium-Test.

Haben wir den Produktnamen extrahiert und die Executable gebaut, erstellen wir einen dynamischen Testfall durch den Aufruf der statischen Methode „dynamicTest(...)“ mit den entsprechenden Parametern. Als Endergebnis der TestFactory-Methode erhalten wir eine Liste von dynamischen Tests, bei denen jeder einzelne eine andere Produktseite testen kann.

Sie können nun die Tests über ihre Entwicklungsumgebung oder dem folgenden Kommando ausführen lassen:

Der Test navigiert zunächst zur Übersichtsseite der Kategorie „Notebooks“. Nachdem die Tests zusammengebaut wurden, schließt sich der Chrome und die Ausführung der einzelnen Tests beginnt. Jeder Test öffnet eine neue Instanz von Chrome, navigiert auf die entsprechende Produktseite und überprüft Bild und Beschreibung. Stellt ein Test einen Fehler fest, wird nur dieser als fehlgeschlagen markiert. Alle anderen Tests werden trotzdem regulär ausgeführt. Sollten neue Notebooks hinzukommen oder

```
...
//Sammel alle Produkte auf der Seite ein
List<WebElement> products = notebookOverview.getProducts();

  //Funktion zum bauen der einzelnen Tests
  Function<WebElement, DynamicTest> buildDynamicTests = element -> {
  // Informationen vom Element einsammeln
  WebElement caption = element.findElement(By.xpath(".//*[contains(@class, \"caption\")]")).findElement(By.xpath(".//a"));
    String productName = caption.getText();
    String linkToProduct = caption.getAttribute("href");

    //Testfunktion für den Test bauen
    Executable testFunction = () -> {
      ChromeDriver testdriver = new ChromeDriver();
      try {
        //Navigiere zum Produkt
        testdriver.get("http://localhost:80");
        testdriver.get(linkToProduct);
        // Überprüfe ob Bild und Beschreibung existieren
        ProductPage productPage = new ProductPage(testdriver);
        assertTrue(productPage.checkMainImageExists(), "Main Image is missing on Product Page!");
        assertTrue(productPage.checkDescriptionExists(), "Description is missing on Product Page!");
      }
      finally {
        //Beende Driver nach dem Test
        testdriver.quit();
      }
    };
  // Aufruf von dynamicTest() zum Bauen des Tests
  return dynamicTest("Product Test: " + productName, testFunction)
  };

  // Für jedes gefundene Produkt, baue einen Dynamischen Test
  List<DynamicTest> testList = products.stream().map(buildDynamicTests).collect(Collectors.toList());
driver.quit();

// Gib die Liste von dyamischen Tests zurück
return testList;
}
```

```
mvn test -Dwebdriver.chrome.driver=</Path/To/chromedriver-executable> --fail-at-end
```

```
junit.jupiter.execution.parallel.enabled = true
junit.jupiter.execution.parallel.mode.default = concurrent
```

Weitere Informationen zur parallelen Ausführung von Tests finden Sie in der offiziellen Dokumentation[1].

Notebooks aus dem Webshop gelöscht werden, passen sich die Tests automatisch entsprechend an.

… UND SCHALTEN DEN TURBO EIN

JUnit 5 unterstützt auch die parallele Ausführung von Tests als experimentelles Feature. In Kombination mit dynamischen Tests, ergibt das vor allen Dingen dann Sinn, wenn viele Tests auf einmal generiert werden. Die eigentliche Orchestrierung übernimmt dabei JUnit5. Die Testklasse muss hierfür mit der Annotation @Execution(ExecutionMode. CONCURRENT) versehen werden. Anschließend müssen folgende Einträge in der junit-platform.properties gesetzt werden. Die Tests werden nun automatisch parallel ausgeführt.

1 https://junit.org/junit5/docs/current/user-guide/#writing-tests-parallel-execution

FAZIT

Mit dynamischen Tests von JUnit 5 wird eine elegante und wartungsarme Lösung für das Testen von dynamischem Content angeboten, beispielsweise Produkte in einem Webshop oder Blogeinträge auf einer Website. Diese Art von Test skaliert automatisch mit neuem Content und ermöglicht so eine viel breitere Testabdeckung als von Hand geschriebene Tests. Besonders in Kombination mit der parallelen Ausführung kann so eine Vielzahl von Tests in kurzer Zeit ausgeführt werden, vorausgesetzt die passenden Ressourcen können bereitgestellt werden. Gleichzeitig bleibt das

IT-Wissen auf den Punkt gebracht

C. Lilienthal · H. Schwentner

Domain-Driven Transformation

Monolithen und Microservices zukunftsfähig machen

312 Seiten · in Farbe
Broschur · € 34,90 (D)
ISBN 978-3-86490-884-2

J. Shore

Die Kunst der agilen Entwicklung

Grundlagen, Methoden und Praktiken

744 Seiten
Festeinband · € 49,90 (D)
ISBN 978-3-86490-860-6

T. Linz

Testen in agilen Projekten

Methoden und Techniken für Softwarequalität in der agilen Welt

Aus- und Weiterbildung zum ISTQB® Certified Tester: Agile Tester, Agile Technical Tester, Agile Test Leadership at Scale

3. Auflage · 326 Seiten
Festeinband · € 39,90 (D)
ISBN 978-3-86490-961-0

R. Pichler

Strategisches Produkt-management

Produktstrategien und Roadmaps für digitale Produkte und agile Teams

Übersetzung der 2. Auflage
212 Seiten
Broschur · € 29,90 (D)
ISBN 978-3-86490-965-8

C. Bendermacher · C. Kampschuur · R. van Solingen

Toolkit für Agile Coaches und Scrum Master

99 kreative Übungen und Workshop-Formate

348 Seiten
Broschur · € 36,90 (D)
ISBN 978-3-86490-967-2

dpunkt.verlag

Bundle up!
Print & E-Book nur auf www.dpunkt.de

Reporting übersichtlich und kann so zu einer schnellen Fehlerbehebung beitragen.

UNSER KLEINER BEISPIELSHOP

Damit Sie das dynamische Testen in einer eigenen kleinen Umgebung selbst ausprobieren können, haben wir auf https://webmedia.testfabrik.com/WebshopExample.zip ein kleines Paket geschnürt. Als Voraussetzungen benötigen Sie auf Ihrem Rechner docker, Java, Maven sowie Chrome mit passendem ChromeDriver. Im Paket enthalten sind der Beispielcode aus diesem Artikel sowie eine vorinitialisierte Instanz von OpenCart, einem Open-Source E-Commerce System. Dieses kann einfach lokal per docker-compose gestartet werden.

Das Shopsystem ist bereits mit einigen Beispielprodukten ausgestattet, von denen auch manche Fehler enthalten, die durch unsere Tests entdeckt werden sollen. Wenn Sie eigenständig Änderungen am Shopsystem vornehmen wollen, können Sie sich unter „http://localhost/admin“ mit dem Usernamen „admin“ und dem Passwort „changeme“ in den Adminbereich einloggen. Der Beispielcode in Java liegt im Ordner „dynamic-test-example“. ■

DER AUTOR

FREDERIC UNTERDÖRFERS
Begeisterung für Software Testing wurde während seines Studiums an der Universität des Saarlandes geweckt und hat ihn seither nicht mehr losgelassen. Er ist derzeit als Produktmanager für die Plattform webmate bei der Testfabrik AG tätig.

Hier geht es zum Beispielshop

Schulungen 2023/24

Dezember 2023

Certified-Schulungen werden ausschließlich von akkreditierten Unternehmen durchgeführt. Das iSQI fungiert hier als Vermittler. Anmeldeformular und Preise unter www.isqi.org.

Termine **to go**
einfach aus der Heftmitte herausstrennen

Mehr als 100 weitere Termine finden Sie unter: *www.isqi.org/de*

iSQI SUCCESS IS YOURS

ALLE ANGABEN OHNE GEWÄHR,
BITTE INFORMIEREN SIE SICH ZUSÄTZLICH AUF DEN WEBSEITEN DER SCHULUNGSANBIETER.

Ort	Startdatum	Tage	Anbieter
A4Q Software Development Engineer in Test (SDET)			
online	08.01.2024	3	QualityDojo - TestSolutions Academy
Frankfurt	08.01.2024	3	QualityDojo - TestSolutions Academy
Köln	08.01.2024	3	QualityDojo - TestSolutions Academy
A4Q Security Essentials			
online	19.02.2024	2	QualityDojo - TestSolutions Academy
Berlin	19.02.2024	2	QualityDojo - TestSolutions Academy
München	19.02.2024	2	QualityDojo - TestSolutions Academy
ISTQB® Certified Tester – Foundation Level			
online	18.12.2023	3	oose Innovative Informatik eG
Berlin	19.12.2023	3	Loyal Team GmbH
online	19.12.2023	3	Loyal Team GmbH
online	08.01.2024	20	alfatraining Bildungszentrum GmbH
online	08.01.2024	4	CGI Deutschland B.V. & Co. KG / Cegos Integrata GmbH
online	08.01.2024	4	QualityDojo - TestSolutions Academy
Berlin	08.01.2024	4	QualityDojo - TestSolutions Academy
München	08.01.2024	4	QualityDojo - TestSolutions Academy
Hamburg	15.01.2024	3	oose Innovative Informatik eG
online	22.01.2024	3	CGI Deutschland B.V. & Co. KG/Cegos Integrata GmbH
Berlin	24.01.2024	3	Loyal Team GmbH
online	24.01.2024	3	Loyal Team GmbH
Zürich	29.01.2024	3	Infometis AG
Zürich	29.01.2024	3	Infometis AG
online	29.01.2024	4	CGI Deutschland B.V. & Co. KG/Cegos Integrata GmbH
online	05.02.2024	20	alfatraining Bildungszentrum GmbH
online	05.02.2024	4	QualityDojo - TestSolutions Academy
Frankfurt	05.02.2024	4	QualityDojo - TestSolutions Academy
Köln	05.02.2024	4	QualityDojo - TestSolutions Academy
Berlin	14.02.2024	3	Loyal Team GmbH
online	14.02.2024	3	Loyal Team GmbH
Düsseldorf	19.02.2024	4	CGI Deutschland B.V. & Co. KG / Cegos Integrata GmbH
Wiesbaden	27.02.2024	3,5	dictaJet QC GmbH & Co. KG
ISTQB® Certified Tester – Foundation Level Extension, Agile Tester			
Berlin	14.12.2023	2	Loyal Team GmbH
online	14.12.2023	2	Loyal Team GmbH
online	14.12.2023	2	QualityDojo - TestSolutions Academy
Berlin	14.12.2023	2	QualityDojo - TestSolutions Academy
Frankfurt	14.12.2023	2	QualityDojo - TestSolutions Academy
München	14.12.2023	2	QualityDojo - TestSolutions Academy
online	18.12.2023	2	dictaJet QC GmbH & Co. KG

wps · alfatraining · methodpark by UL · sepp.med Qualität sichert Erfolg · INNOTIVUM · oose. · loyal team for software quality · .msg · accenture · sogeti Part of Capgemini · SERVIEW · QUALITYDOJO TestSolutions Academy · T Systems · nagarro · CGI · infometis · integrata cegos · expleo academy

Ort	Startdatum	Tage	Anbieter
online	04.01.2024	2	CGI Deutschland B.V. & Co. KG / Cegos Integrata GmbH
online	08.01.2024	20	alfatraining Bildungszentrum GmbH
online	24.01.2024	2	dictaJet QC GmbH & Co. KG
Berlin	24.01.2024	2	Loyal Team GmbH
online	24.01.2024	2	Loyal Team GmbH
online	01.02.2024	2	QualityDojo - TestSolutions Academy
Frankfurt	01.02.2024	2	QualityDojo - TestSolutions Academy
Köln	01.02.2024	2	QualityDojo - TestSolutions Academy
online	05.02.2024	3	oose Innovative Informatik eG
online	05.02.2024	20	alfatraining Bildungszentrum GmbH
München	15.02.2024	2	CGI Deutschland B.V. & Co. KG/Cegos Integrata GmbH
ISTQB® Certified Tester – Advanced Level, Test Manager			
Zürich	15.01.2024	3	Infometis AG
online	15.01.2024	5	CGI Deutschland B.V. & Co. KG/Cegos Integrata GmbH
online	15.01.2024	5	QualityDojo - TestSolutions Academy
Berlin	15.01.2024	5	QualityDojo - TestSolutions Academy
München	15.01.2024	5	QualityDojo - TestSolutions Academy
Berlin	22.01.2024	5	Loyal Team GmbH
online	22.01.2024	5	Loyal Team GmbH
online	12.02.2024	5	QualityDojo - TestSolutions Academy
Frankfurt	12.02.2024	5	QualityDojo - TestSolutions Academy
Köln	12.02.2024	5	QualityDojo - TestSolutions Academy
München	19.02.2024	5	CGI Deutschland B.V. & Co. KG/Cegos Integrata GmbH
online	18.12.2024	5	CGI Deutschland B.V. & Co. KG/Cegos Integrata GmbH
ISTQB® Certified Tester – Advanced Level, Test Analyst			
online	15.01.2024	4	QualityDojo - TestSolutions Academy
Frankfurt	15.01.2024	4	QualityDojo - TestSolutions Academy
Köln	15.01.2024	4	QualityDojo - TestSolutions Academy
online	12.02.2024	4	CGI Deutschland B.V. & Co. KG/Cegos Integrata GmbH
ISTQB® Certified Tester – Advanced Level, Technical Test Analyst			
online	26.02.2024	3	CGI Deutschland B.V. & Co. KG/Cegos Integrata GmbH
ISTQB® Certified Tester – Advanced Level, Security Tester			
online	17.01.2024	3	CGI Deutschland B.V. & Co. KG/Cegos Integrata GmbH
ISTQB® Certified Tester – Advanced Level, Test Automation Engineer (TAE)			
online	22.01.2024	4	QualityDojo - TestSolutions Academy
Berlin	22.01.2024	4	QualityDojo - TestSolutions Academy
München	22.01.2024	4	QualityDojo - TestSolutions Academy
online	26.02.2024	4	QualityDojo - TestSolutions Academy
Frankfurt	26.02.2024	4	QualityDojo - TestSolutions Academy
Köln	26.02.2024	4	QualityDojo - TestSolutions Academy
online	04.03.2024	3	CGI Deutschland B.V. & Co. KG/Cegos Integrata GmbH
IREB® Certified Professional for Requirements Engineering – Foundation Level			
Frankfurt	18.12.2023	3	Cegos Integrata GmbH
Berlin	19.12.2023	3	Loyal Team GmbH
online	19.12.2023	3	Loyal Team GmbH
online	03.01.2024	3	Cegos Integrata GmbH
online	08.01.2024	20	alfatraining Bildungszentrum GmbH
online	08.01.2024	5	Cegos Integrata GmbH
Zürich	22.01.2024	3	Infometis AG
Hamburg	22.01.2024	5	oose Innovative Informatik eG
online	22.01.2024	3	Cegos Integrata GmbH
Berlin	24.01.2024	3	Loyal Team GmbH
online	24.01.2024	3	Loyal Team GmbH
Düsseldorf	29.01.2024	5	Cegos Integrata GmbH
online	29.01.2024	3	QualityDojo - TestSolutions Academy
Berlin	29.01.2024	3	QualityDojo - TestSolutions Academy
München	29.01.2024	3	QualityDojo - TestSolutions Academy
online	29.01.2024	3	QualityDojo - TestSolutions Academy

Ort	Startdatum	Tage	Anbieter
Berlin	29.01.2024	3	QualityDojo - TestSolutions Academy
München	29.01.2024	3	QualityDojo - TestSolutions Academy
online	05.02.2024	20	alfatraining Bildungszentrum GmbH
Hamburg	12.02.2024	3	oose Innovative Informatik eG
München	12.02.2024	3	Cegos Integrata GmbH
online	19.02.2024	5	Cegos Integrata GmbH
IREB® Certified Professional for Requirements Engineering – Advanced Level, Requirements Modeling			
Düsseldorf	10.01.2024	3	Cegos Integrata GmbH
IREB® Certified Professional for Requirements Engineering – Advanced Level, Requirements Management			
online	07.02.2024	3	Cegos Integrata GmbH
IREB® – RE@Agile Primer			
online	13.12.2023	3	Cegos Integrata GmbH
online	29.01.2024	3	Cegos Integrata GmbH
online	28.02.2024	3	oose Innovative Informatik eG
ISAQB® Software Architecture – Foundation Level			
München	15.01.2024	4	Cegos Integrata GmbH
Hamburg	22.01.2024	4	oose Innovative Informatik eG
online	22.01.2024	3	Cegos Integrata GmbH
online	13.02.2024	4	Cegos Integrata GmbH
Hamburg	26.02.2024	4	oose Innovative Informatik eG
ISAQB® Software Architecture – Advanced Level			
online	18.12.2023	3	oose Innovative Informatik eG
online	16.01.2024	3	oose Innovative Informatik eG
online	24.01.2024	3	oose Innovative Informatik eG
Hamburg	14.02.2024	3	oose Innovative Informatik eG
Hamburg	26.02.2024	3	oose Innovative Informatik eG
online	26.02.2024	2	oose Innovative Informatik eG
ISTQB® CTFL Test Data Specialist (GTB)			
online	14.12.2023	2	CGI Deutschland B.V. & Co. KG/Cegos Integrata GmbH
ISTQB® Certified Tester – Acceptance Testing			
online	12.02.2024	2	QualityDojo - TestSolutions Academy
Berlin	12.02.2024	2	QualityDojo - TestSolutions Academy
München	12.02.2024	2	QualityDojo - TestSolutions Academy
UXQB® Certified Professional for Usability and User Experience – Foundation Level			
Heidelberg	29.11.2023	3	ProContext Consulting GmbH
Hamburg	29.01.2024	3	oose Innovative Informatik eG
online	29.01.2024	6 x 4 Stunden	artop - Institut an der Humboldt-Universität zu Berlin
Köln	26.02.2024	4	ProContext Consulting GmbH
online	29.02.2024	2	CGI Deutschland B.V. & Co. KG/Cegos Integrata GmbH
Heidelberg	20.06.2024	3	ProContext Consulting GmbH
Heidelberg	11.09.2024	3	ProContext Consulting GmbH
Köln	25.11.2024	3	ProContext Consulting GmbH
UXQB® Cerified Professional for Usability and User Experience – Essentials in UX and HCD Management			
Heidelberg	23.02.2024	2	ProContext Consulting GmbH
Heidelberg	03.06.2024	1	ProContext Consulting GmbH
UXQB® Cerified Professional for Usability and User Experience – Advanced Level „User Requirements Engineering“			
Köln	16.04.2024	3	ProContext Consulting GmbH
UXQB® Cerified Professional for Usability and User Experience – Advanced Level „Designing Solutions“			
Köln	13.05.2024	3	ProContext Consulting GmbH
Köln	18.11.2024	3	ProContext Consulting GmbH
UXQB® Cerified Professional for Usability and User Experience – Advanced Level „Usability Testing & Evaluation“			
Köln	23.09.2024	3	ProContext Consulting GmbH

AGILE **REQUIREMENTS ENGINEERING** **SECURITY** **MOBILE**

SOFTWARE TESTING **PROJECT MANAGEMENT** **USABILITY** **SPECIALISED**

PRODUCT MANAGER **SOFTWARE ARCHITECTURE** **WEITERE ANGEBOTE** **MEDICAL**

Seminare 2023/24

Dezember 2023

Das iSQI fungiert hier als Vermittler.
Ausfühliche Seminarbeschreibungen,
Preise und Anmeldeformular unter www.isqi.org.

Seminartitel	Ort	Startdatum	Tage	Anbieter
Design Thinking erleben und begreifen	online	18.12.2023	2	oose Innovative Informatik eG
Modellbasiertes Systems Engineering mit SysML inklusive Vorbereitung und Prüfung OCSMP-Model-User	Hamburg	15.01.2024	5	oose Innovative Informatik eG
Vorbereitungskurs Certified Systems Engineer Ebene C	Hamburg	15.01.2024	12	oose Innovative Informatik eG
Vorbereitungskurs Certified Systems Engineer Ebene B	Hamburg	16.01.2024	12	oose Innovative Informatik eG
Grundkurs Künstliche Intelligenz	online	18.01.2024	1	oose Innovative Informatik eG
Web-Authentifizierung	Hamburg	22.01.2024	3	oose Innovative Informatik eG
Agiles Software-Projektmanagement	Hamburg	05.02.2024	3	oose Innovative Informatik eG
Geschäftsprozesse mit BPMN 2.0 verständlich modellieren inkl. Zertifizierung OCEB2-F	Hamburg	05.02.2024	5	oose Innovative Informatik eG
OKR – Objectives and Key Results at Scale	Hamburg	08.02.2024	2	oose Innovative Informatik eG
Agile Moderationsmethoden – analog und digital	Hamburg	12.02.2024	2	oose Innovative Informatik eG
Der Quality Evangelist	online	12.02.2024	3	oose Innovative Informatik eG
Softwaretester:in inklusive ISTQB® Foundation Level	online/Videokonferenzsystem alfaview	08.01.2024	20	alfatraining Bildungszentrum GmbH
Softwaretester:in inklusive ISTQB® Foundation Level	online/Videokonferenzsystem alfaview	05.02.2024	20	alfatraining Bildungszentrum GmbH
Softwaretester:in/Agile Tester	online/Videokonferenzsystem alfaview	04.12.2023	20	alfatraining Bildungszentrum GmbH
Softwaretester:in/Agile Tester	online/Videokonferenzsystem alfaview	08.01.2024	20	alfatraining Bildungszentrum GmbH
Softwaretester:in/Agile Tester	online/Videokonferenzsystem alfaview	05.02.2024	20	alfatraining Bildungszentrum GmbH
Anforderungsmanager:in	online/Videokonferenzsystem alfaview	08.01.2024	20	alfatraining Bildungszentrum GmbH
Anforderungsmanager:in	online/Videokonferenzsystem alfaview	05.02.2024	20	alfatraining Bildungszentrum GmbH

STAND: Dezember 2023

Zusätzliche Schulungs- und Seminartermine finden Sie auf www.isqi.org!

Alle Termine auch als Inhouse-Angebot buchbar!

Irrtümer, Termin- und Preisänderungen vorbehalten. Es gelten die allgemeinen Geschäfts- und Preisbedingungen des jeweiligen Veranstalters.

Voltastraße 1
14482 Potsdam
Tel.: +49 331 231810-0
Fax: +49 331 231810-10

Ansprechpartnerin:
Nadia Diesend
Customer Success
+49 331 231810-33
exam@isqi.org

DIE WEIHNACHTSZEIT EINLÄUTEN: SOLIBASAR FÜR KINDER IN NOT

Die Weihnachtszeit rückt näher, und mit ihr auch die Gelegenheit, Freude zu schenken und an andere zu denken. In diesem Sinne führte iSQI auch in diesem Jahr seinen traditionellen Weihnachts-Solibasar durch, erstmals in Zusammenarbeit mit dem neuen Coworking- und Eventspace „1448zwo“ in Babelsberg. Die Mitarbeiterinnen und Mitarbeiter von iSQI trugen Sachspenden zusammen, backten und bastelten und stellten eine vielfältige Produktpalette für den Basar zur Verfügung. Die erzielten Erlöse des Basarverkaufs werden von iSQI verdoppelt und der Arbeiterwohlfahrt (AWO) Potsdam zur Verfügung gestellt. Die AWO Potsdam setzt diese Spenden ein, um Kindern aus sozial benachteiligten Familien kleine Weihnachtsgeschenke zu ermöglichen.

Der Weihnachtsbasar erwies sich als voller Erfolg, sowohl finanziell durch die beeindruckende Spendensumme als auch durch die aktive Teilnahme von Nachbarn und Anwohnern aus dem Stadtteil Potsdam-Babelsberg. Viele Besucher nutzten die Gelegenheit, die Räumlichkeiten der „Alten Post“ in Babelsberg kennenzulernen und gleichzeitig kleine Weihnachtsgeschenke auf dem Solibasar einzukaufen. Stephan Goericke, CEO von iSQI, freut sich über das gelungene Event: „Mit dem Weihnachtsbasar haben wir eine schöne Aktion gestartet, die wir auch in den folgenden Jahren fortsetzen möchten. Damit zeigen wir nicht nur aktive Präsenz in unserem Kiez, sondern fördern auch unseren Teamzusammenhalt und das soziale Engagement für Potsdam.“

Die erfolgreiche Zusammenarbeit von iSQI und 1448zwo mit Unterstützung der AWO Potsdam markiert den Auftakt zur festlichen Jahreszeit. iSQI freut sich auf weitere gemeinsame Initiativen und darauf, auch in Zukunft einen Beitrag zur Unterstützung der Region leisten zu können.

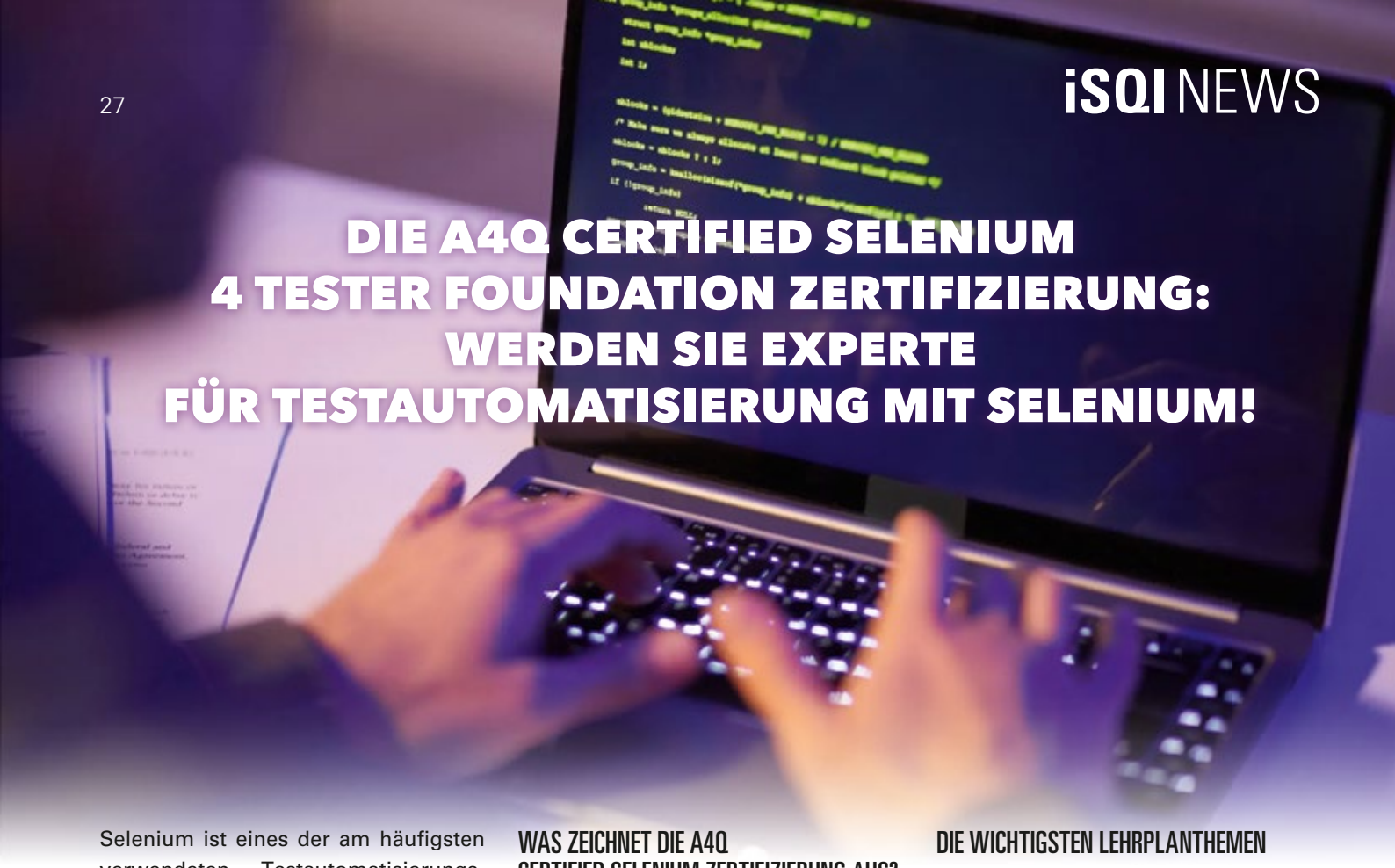
MEHR BEWUSSTSEIN FÜR REQUIREMENTS ENGINEERING: ISQI UND IREB AUF ROADSHOW DURCH OSTEUROPA

Wie können wir die Zukunft der IT-Branche gestalten, und welche Rolle nehmen dabei Qualitätssicherung und Requirements Engineering ein? Mit dieser Frage bereisten iSQI und das International Requirements Engineering Board (IREB) im November gemeinsam Osteuropa. Die Roadshow führte die beiden Organisationen durch die Städte Bukarest, Belgrad, Sarajevo und Riga. In jeder Stadt konnten sich die Teilnehmer der Veransatltung auf inspirierende Netzwerkabende und spannende Einblicke in die Themen Anforderungsmanagement und -gestaltung sowie die Bedeutung der Aus- und Weiterbildung in diesem Bereich freuen, immer mit dem Fokus auf der Verbesserung von Softwareprodukten.

Stan Bühne, Vorsitzender von IREB, äußerte sich begeistert über die Veranstaltung: „Die lebhaften Diskussionen, Reaktionen und die enge Verbindung zur Community zeigen, dass ‚Community' mehr ist als nur Likes in den sozialen Medien. Wir setzen uns aktiv dafür ein, Fachwissen, Kreativität und Unterstützung in den Bereichen Requirements Engineering und Digital Design zu fördern."

Stefan Goericke, CEO von iSQI, ist ebenso erfreut über den Erfolg der Veranstaltung: „Wir konnten zahlreiche Kontakte zu Kollegen in Osteuropa knüpfen und sind begeistert von den frischen Einblicken und den spannenden Meinungen zur Qualitätssicherung und zum Requirements Engineering in dieser Region. Der wertvolle Austausch wird unsere Arbeit auch in Zukunft fruchtbar weiterentwickeln."

IREB fördert seit 2006 das Berufsbild des Requirements Engineerings und hat ein internationales Weiterbildungsschema zur Erlangung international anerkannter Qualifikationen für Requirements Engineers und Business Analysten etabliert. Mit der Certified Professional of Requirements Engineering (CPRE) Zertifizierung setzt IREB den Standard für Requirements-Engineering-Wissen auf internationaler Ebene. Neu hinzugekommen ist das Schema des Digital Design Professionals, das sich mit der Gestaltung digitaler Materialien und dem ganzheitlichen Denken in Bezug auf digitale Lösungen beschäftigt. Das CPRE Ausbildungsprogramm wird mittlerweile in 68 Ländern angeboten. Die Osteuropa-Roadshow hat die Verbindungen in dieser Region erheblich gestärkt. ■



DIE A4Q CERTIFIED SELENIUM 4 TESTER FOUNDATION ZERTIFIZIERUNG: WERDEN SIE EXPERTE FÜR TESTAUTOMATISIERUNG MIT SELENIUM!

Selenium ist eines der am häufigsten verwendeten Testautomatisierungs-Frameworks weltweit und wird von Testern und Entwicklern gleichermaßen genutzt, um Webbrowser-Tests zu automatisieren.

Wenn Sie Ihre Tests noch effizienter gestalten möchten, dann ist die A4Q Certified Selenium 4 Tester Foundation Schulung genau das Richtige für Sie. In dieser Schulung erfahren Sie alles Wichtige über die Anwendung von Selenium 4 zur Testautomatisierung sowie bewährte Methoden für Ihre Selenium-Automatisierungsprojekte.

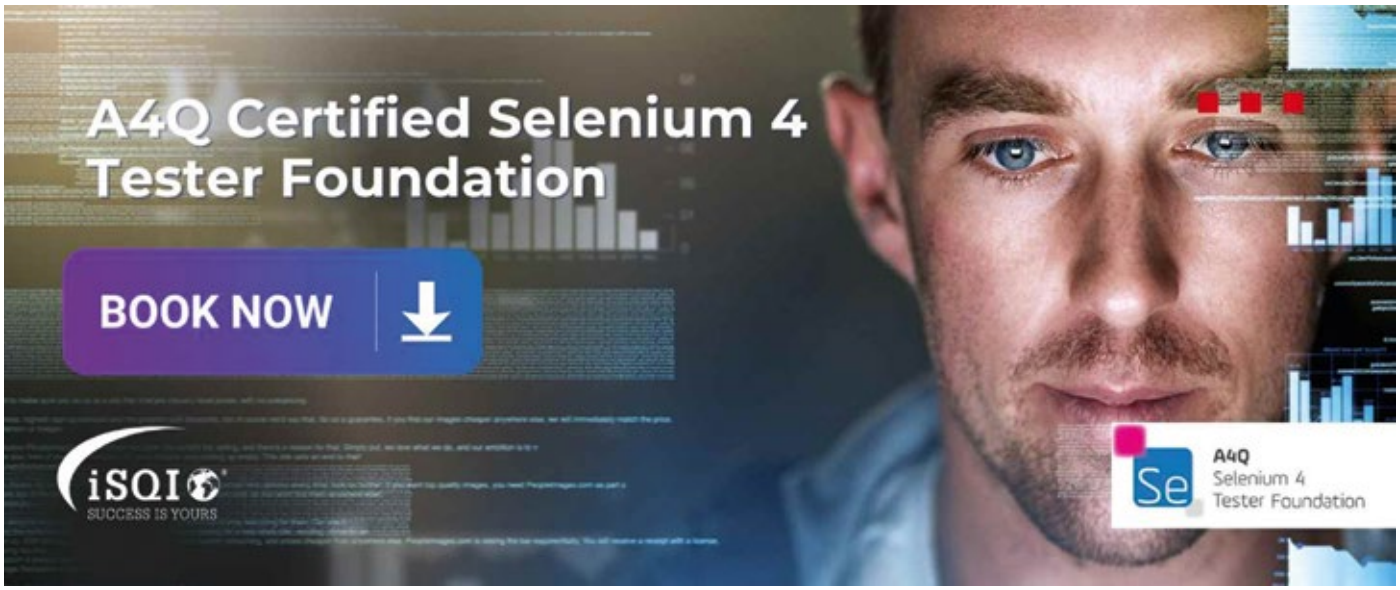
WAS ZEICHNET DIE A4Q CERTIFIED SELENIUM ZERTIFIZIERUNG AUS?

Die A4Q Certified Selenium 4 Tester Foundation Zertifizierung bringt theoretische Grundlagen der Testautomatisierung mit der praktischen Anwendung von Selenium zur Testautomatisierung für Webanwendungen zusammen! Diese Schulung richtet sich an Softwaretester und Softwareentwickler, die nicht nur theoretisch, sondern auch in der Praxis verstehen möchten, wie man automatisierte Tests mit Selenium konzipiert, umsetzt und pflegt.

DIE WICHTIGSTEN LEHRPLANTHEMEN

Einführung in die Testautomatisierung: Hier erhalten Sie eine grundlegende Einführung in die Testautomatisierung und lernen, wie sie in verschiedenen Testebenen angewendet wird.

Automatisierung von Webtechnologien: Sie erfahren, wie Webseiten und Webanwendungen aufgebaut sind, wie verschiedene Web-Technologien miteinander interagieren und wie Selenium Locators funktionieren, um Elemente auf einer Webseite zu identifizieren und darauf zuzugreifen.



Selenium Locators: Die richtige Verwendung von Selenium Locators ist entscheidend, um Interaktionen mit den Elementen einer Webseite zu automatisieren. Sie lernen die verschiedenen Arten von Selenium Locators kennen und für welchen Zweck sie eingesetzt werden.

Testautomatisierung mit Selenium: Sie lernen Schritt für Schritt, wie die Testautomatisierung mit Selenium funktioniert, welche Voraussetzungen erforderlich sind und welche Automatisierungstools im Selenium-Framework zur Verfügung stehen, wie Selenium IDE, Selenium WebDriver und Selenium Grid.

Selenium-Frameworks: Sie werden mit den verschiedenen Selenium-Frameworks und den unterstützten Programmiersprachen, nämlich Java und Python, vertraut gemacht.

Verwendung von Selenium WebDriver: Erfahren Sie, wie Sie den Selenium WebDriver effektiv nutzen können, um Tests auf verschiedenen Plattformen und Browsern auszuführen.

Implementierung von Testautomatisierung in einer Organisation: Lernen Sie bewährte Methoden zur Einführung von Testautomatisierung in Ihrem Unternehmen und zur Integration in den Entwicklungsprozess kennen.

Java oder Python? Sie haben die Wahl! Sie können zwischen Java und Python als Referenzsprache wählen und werden in beiden Sprachen geschult, um Ihre Flexibilität zu erhöhen.

Mit dieser Selenium-Schulung erwerben Sie nicht nur fundiertes Wissen über Selenium 4, sondern auch die praktischen Fähigkeiten, die Sie brauchen, um automatisierte Tests für Webanwendungen in Projekten durchzuführen. Mit dem A4Q Certified Selenium 4 Tester Foundation Zertifikat erhalten Sie einen international gültigen Nachweis für Ihre umfassende Expertise in der Testautomatisierung mit Selenium! ■

LOW-CODE: DIGITALISIERUNG EINFACH MACHEN

AUTOR: **GEORG REDEKOP**

Nach wie vor stellt die Digitalisierung viele Organisationen vor eine Herausforderung. Allein der Fachkräftebedarf macht es vielen unmöglich, die digitale Transformation voranzutreiben. So hat der IT-Verband Bitkom allein für das Jahr 2022 in Deutschland einen IT-Fachkräftebedarf in Höhe von 137.000 Personen ermittelt. Auf der anderen Seite hat Dr. Andrew Russell (Nvidia) in einem Forschungsprojekt herausgefunden, dass etwa 60 Prozent der Softwareentwicklungskosten ins Debugging von bestehender Software fließen. Der Rückschluss: Es braucht schon allein immer mehr Softwareentwickler:innen und -tester:innen, um die bestehende Software zu pflegen. So bleiben nur wenige Kapazitäten, um neue Apps zu entwickeln.

Neben dem Fachkräftebedarf spielt auch die Finanzierung eine wichtige Rolle. Derzeit sitzt das Geld in Deutschland nicht sehr locker. So lag der GFK-Konsumklima-Index im August 2023 bei einem Indexwert von -24,6 Punkten. Und auch im September soll es nicht besser aussehen. Die Prognose liegt bei -25,5 Punkten. Damit wäre er das 22. Mal in Folge negativ. Das merken natürlich auch die Unternehmen in Form von stagnierenden oder sogar rückläufigen Umsätzen, was zu vorsichtigen bzw. verhaltenen Investitionen in die Zukunft führt. Auf der anderen Seite besteht ein enormer Digitalisierungsdruck. Die Kundinnen und Kunden wollen digital konsumieren, z. B. mit ihren Smartphones einkaufen, Produkte bezahlen und ihre Päckchen nachverfolgen.

Vor diesem Hintergrund kann die Low-Code-Technologie Unternehmen dabei helfen, den Weg in die Digitalisierung zu vereinfachen, ohne zusätzliche Fachkräfte und kostenintensive Software. Die gar nicht so neue Idee ist, dass man Software nicht mehr Zeile für Zeile schreibt, sondern bereits vorkonfigurierte Bausteine nutzt: Anwendungen werden damit eher designt als geschrieben.

Es lässt sich gut mit einem Baukasten für Webseiten vergleichen. Im Kern bestehen nämlich auch Apps aus wiederkehrenden Elementen: Übersichten, Detailansichten, Buttons, Eingabefelder etc. Warum immer wieder einzeln programmieren, wenn man sie als vorkonfigurierte Elemente zur Verfügung stellen kann?

DER SIEGESZUG VON LOW-CODE

Der aktuelle Siegeszug hat wohl auch etwas mit dem griffigen Namen zu tun, den die Forrester Research Analysten Clay Richardson und John Rymer im Jahr 2014 prägten. Als diese auf die hier beschriebene Entwicklung stießen, hatte das Kind noch keinen Namen, obwohl die Ansätze bereits in den 1980er und 1990er verwendet wurden, um Prototypen zu entwickeln. Doch beim Prototyping ist es seitdem nicht geblieben. Das Marktforschungsinstitut Gartner schätzt, dass bis zum Jahr 2026 80 Prozent aller Apps mit Low-Code-Technologien entwickelt werden sollen. Die aktuelle Wachstumsprognose des Marktes für das Jahr 2023 beträgt 20 Prozent. Auch große Player wie Microsoft, Mendix (Siemens Tochter), Salesforce, ServiceNow und OutSystems sind auf den Trend aufgesprungen und vermarkten vehement ihre Lösungen. Damit ist Low-Code der ursprünglichen Idee, Citizen Developer zu befähigen, selbst ihre Apps zu schreiben, entwachsen. Low-Code ist nicht mehr „low“, sondern marktreif für die professionelle Softwareentwicklung.

Während das Geschäftsmodell der deutschen Industrie von hoher Kapitalbindung und niedrigen Margen geprägt ist, zeichnet sich das Geschäftsmodell der Digitalkonzerne aus dem Silicon Valley durch niedrige Kapitalbindung und hohe Margen aus. Dies wird möglich, weil die Unternehmen drei Dinge miteinander kombinieren: Software, Hardware und Plattform. Letzteres lässt die beiden anderen skalieren. Software und Hardware lassen sich auch als Synonyme für Services und Produkte verstehen. Damit stellt dies das Geschäftsmodell der digitalen Transformation im 21. Jahrhundert dar.

Eines der bekanntesten Beispiele hierfür ist wohl Apple. Das Unternehmen hat ein Betriebssystem (Software) für Smartphones entwickelt. Die selbst designten Smartphones (Hardware) lässt das Unternehmen günstig in Asien fertigen. Anstatt jedoch tausende Entwicklerinnen und Entwickler fest einzustellen, um all die

Apps für die Endnutzer:innen selbst zu entwickeln, hat das Unternehmen eine Plattform entwickelt, mit deren Hilfe externe Entwickler:innen aus der ganzen Welt Apps entwickeln können, an deren Erlösen Apple über die iOS-Plattform mitverdient. Mit Hilfe der Low-Code Technologie ist dieses Vorgehen nun auch für Unternehmen möglich, die nicht zwangsweise über die notwendige Finanzkraft und Fachkräfte verfügen.

HERAUSFORDERUNGEN BEI DER EINFÜHRUNG VON LOW-CODE

Doch trotz der Vorteile, die die Low-Code Technologie mit sich bringt, kann diese Technologie nicht mit einem Fingerschnipsen im Unternehmen etabliert werden. Eine der anfänglichen Herausforderungen stellt die Akzeptanz und das Verständnis für die Technologie dar. Wir Menschen tun uns bekanntermaßen schwer mit Veränderungen. Neue Dinge zu erlernen, kostet uns immer Energie. Insbesondere Mitarbeiter:innen, die bisher mit traditionellen Methoden und Technologien Software entwickelt haben, können sich mit Low-Code schwer tun und es als eine Entwicklungsumgebung für Laien als für Profis abtun. Diese und ähnliche Bedenken gilt es aufzudecken und aufzulösen, z. B. indem man für positive Referenzerlebnisse sorgt, die den Betroffenen den Mehrwert vermitteln. Mehrwert in Form von Vereinfachung, Mehrwert in Form von Zeitgewinn und schließlich Mehrwert in Form von zufriedenen Kunden, deren Projekte zeitnah umgesetzt werden.

Eine weitere Herausforderung stellt die Auswahl der passenden Low-Code-Plattform dar. Zum Leidwesen der Anwender:innen gibt es mittlerweile Low-Code-Plattformen wie Sand am Meer. Daher kommt man um eine klassische Bedarfsanalyse nicht herum. Was soll die Plattform leisten können? Mit welchen Systemen soll sie interagieren können? Handelt es sich um eine strategische Plattform, die die Altsysteme ersetzen soll? Oder sollen die bestehenden Lücken geschlossen werden? Unternehmen sollten sicherstellen, dass die Plattform gut mit den vorhandenen Anwendungen kommunizieren kann, um einen reibungslosen Datenfluss und eine effektive Zusammenarbeit zwischen Low-Code-Anwendungen und bestehenden Systemen zu gewährleisten. Ein durchdachtes Migrationskonzept und klare Integrationsstrategien sind hierbei unerlässlich.

Und dann wäre da natürlich das Thema der „IT-Sicherheit“, das wie ein Elefant im Porzellanladen vor einem steht. Es ist nicht zu leugnen, dass die großen Player aus dem angelsächsischen Raum kommen. Diese Tatsache und ein damit verbundener zweifelhafter Umgang mit (personengebundenen) Daten stellen insbesondere Organisationen der öffentlichen Verwaltung vor eine Herausforderung, sich eine Low-Code-Plattform ins Haus zu holen. Abhilfe kann hier die aufstrebende deutsche Low-Code-Bewegung schaffen. Auf dem diesjährigen German Low-Code Day in Hannover kamen insbesondere die deutschen Anbieter zusammen und zeigten eindrucksvoll, dass die Technologie und die dahinterstehenden Ideen auch in Deutschland angekommen sind und gelebt werden.

AUSBLICK: SO SIEHT DIE ZUKUNFT AUS

Und auch die Zukunft lässt nicht lange auf sich warten. Aktuell sind zahlreiche Low-Code Anbieter dabei, KI-Anwendungen wie ChatGPT in ihre Plattformen zu integrieren. Die Idee ist, dass der Entwicklungsprozess sich damit weiter beschleunigen lässt. Das beste Beispiel hierfür ist der Copilot von Microsoft. Der auf einem generativen Sprachmodell basierende Assistent wird aktuell in den USA ausgerollt. Mit seiner Hilfe lassen sich z. B. Workflows in Power Automate gänzlich mit menschlicher Sprache erstellen. Man beschreibt lediglich das Endergebnis und den Rest macht der Computer. ■

DER AUTOR:

GEORG REDEKOP ist Wirtschaftsingenieur für Elektrotechnik und Experte für digitale Transformation. Mit seinen Impulsen setzt er sich aktiv dafür ein, Menschen für die Möglichkeiten der Digitalisierung zu begeistern. Dabei legt er besonderen Wert auf technologische Trends wie Low-Code und Künstliche Intelligenz sowie digitale Geschäftsmodelle und deren Auswirkungen auf die Arbeitswelt. Sein Motto: „Lasst uns spielen, um zu gewinnen.“ Er ist Teil der Digitalagentur-Niedersachsen, einer Initiative des Innovationszentrums Niedersachsen.

LOW-CODE IN FERTIGUNG UND QUALITÄTSSICHERUNG

ANWENDERORIENTIERTE UND OPTIMIERTE PROZESSLÖSUNGEN FÜR DIE PRODUKTION MIT LOW-CODE NEU DENKEN

AUTORIN: KERSTIN STIER

Quelle: Foto von Michael Dziedzic auf Unsplash

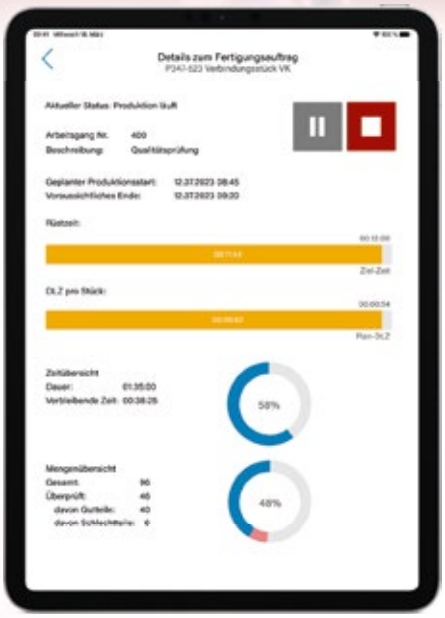
In produzierenden Unternehmen steht die Fertigung in aller Regel im Zentrum der Wertschöpfung. Meist über viele Jahre hinweg haben diese Unternehmen ihre Fertigungsprozesse individuell entwickelt und perfektioniert und die Basis dafür geschaffen, Wettbewerbsvorteile auf globalisierten Märkten zu erlangen und zu sichern. Im Zuge der Digitalisierung gilt es, diese Prozesse in ihrer Einzigartigkeit auch systemseitig abzubilden – mit dem Ziel, Potenziale zur Effizienz- und Produktivitätssteigerung zu heben und zugleich die individuellen Verfahrens- und Produktvorteile beizubehalten.

DIGITALISIERUNG IN MITTELSTAND UND INDUSTRIE - DER STATUS QUO

Durch die Einführung von Standardsoftwaretools und -systemen, wie ERP-, MES- oder QS-Systemen, haben viele Unternehmen bereits Fortschritte in Richtung Digitalisierung gemacht. Bei individuellen Prozessen stoßen diese Systeme jedoch oft an ihre Grenzen. Es ist zwar grundsätzlich möglich, individuelle Anforderungen in einer Standardsoftware abzubilden, jedoch ist dies meist mit erheblichem zeitlichen und finanziellen Aufwand verbunden. Die Realisierung individueller Lösungen führt nicht selten zu Mammutprojekten, die sich über Monate oder sogar Jahre erstrecken und die Organisation eher hemmen als voranbringen. Vor allem im Mittelstand führt die Digitalisierung dazu, dass IT- und Fachabteilungen mit erheblichen und oft schwer zu bewältigenden Herausforderungen konfrontiert werden.

HERAUSFORDERUNGEN IN DER DIGITALISIERUNG VON PRODUKTIONSPROZESSEN

Insbesondere im Bereich der Fertigung treten neben den höchst individuellen Prozessen weitere Herausforderungen auf: Trotz fortgeschrittener Digitalisierung existieren oftmals noch papierbasierte Prozesse, die in verschiedenen



Quelle: engomo GmbH

Alles auf einen Blick: Mit einer individuellen App auf einem mobilen Endgerät wickeln Fertigungsmitarbeiter die BDE ab und haben jederzeit den Überblick über Fertigungsaufträge und -status.

Bereichen anzutreffen sind und deren systemseitige Abbildung nicht vorgesehen ist. Das betrifft beispielsweise die Werkzeugverwaltung oder die

Aktualisierung von Stammdaten im Fertigungsprozess.

Überall dort, wo Papier und Stift eingesetzt werden, sind Ineffizienzen und Fehlerpotenziale nicht weit entfernt. Die manuelle Datenübertragung ist nicht nur arbeitsintensiv, sondern birgt auch das Risiko, dass ungenaue Daten in die Systeme gelangen oder – im Worst Case – Daten gar nicht in den Backends erfasst werden.

Eine weitere Herausforderung, mit der sich Produktionsbereiche in der Regel konfrontiert sehen, ist die Vielfalt der dort eingesetzten Systeme. Mit der Fortschreitung der Digitalisierung nimmt auch die Anzahl der verwendeten Softwarelösungen zu – angefangen beim ERP-System, über Produktionsplanungssoftware (PPS), QS-Systeme, MES-Anwendungen (Manufacturing Execution Systeme) bis hin zum PIM (Produktinformationsmanagement). Die erforderliche Integration dieser Anwendungen ist oft komplex, kostspielig und manchmal auch nicht umsetzbar, was zur Entstehung zahlreicher Insellösungen führt, die nicht miteinander kommunizieren können.

Quelle: Shutterstock | 1412369838 | Drazen Zigic

Individuelle, benutzerfreundliche Apps unterstützen Fertigungsmitarbeiter optimal bei der Arbeitserledigung.

Darüber hinaus ist die Landschaft dieser Systeme meist nicht benutzerfreundlich gestaltet. Die Mitarbeiter in der Produktion verfügen in der Regel über Prozesskenntnisse, aber nicht notwendigerweise über das Know-how zur Bedienung komplexer Softwareoberflächen. In Zeiten, in denen Fachkräfte rar sind, ist es essenziell, die Nutzer durch benutzerfreundliche Apps effektiv und bedarfsgerecht zu unterstützen. Damit können begrenzte Ressourcen optimal genutzt, neue Mitarbeiter:innen schneller eingearbeitet und die Mitarbeitermotivation durch die Nutzung angenehmer und benutzerfreundlicher Anwendungen gefördert werden.

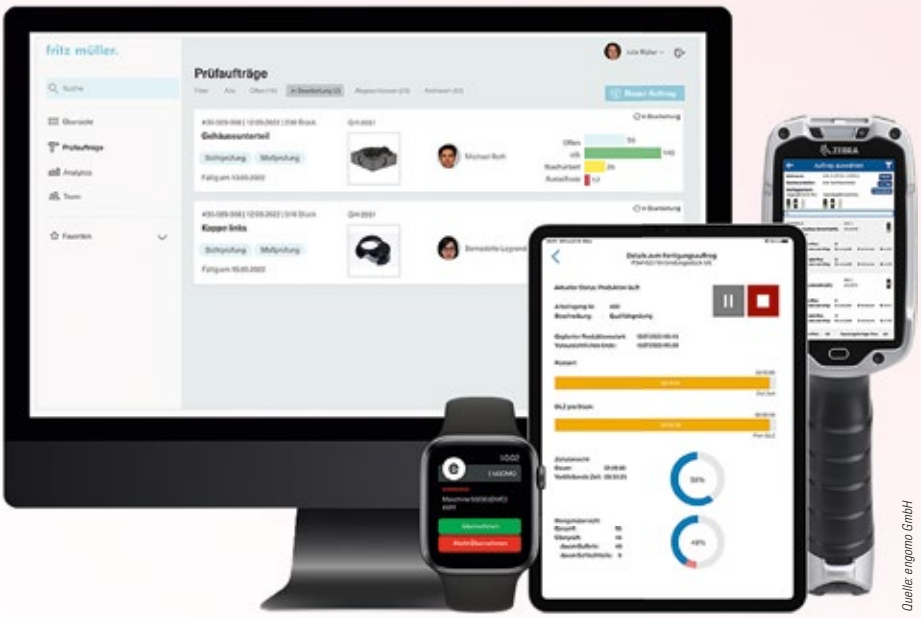
Anwendungsentwicklung für die Produktion: mit Low-Code neu gedacht

An diese vielfältigen Herausforderungen setzt das Prinzip von Low-Code an. Die Technologie denkt mit ihrem Ansatz die Anwendungsentwicklung neu: Anstelle von manuellem Coding tritt bei Low-Code ein Baukasten mit vorgefertigten Möglichkeiten, um Anwendungen individuell zu konfigurieren. Es fällt nur geringer Programmieraufwand an, um Lösungen zu erstellen und bereitzustellen: Benutzeroberflächen werden per Drag-and-drop erstellt, vorgefertigte Funktionsbausteine ermöglichen das schnelle Vorankommen. So können Elemente einfach auf User Interfaces platziert, Datenfelder hinzugefügt oder entfernt und Workflows intuitiv und anwenderorientiert abgebildet werden. So wird gewährleistet, dass nur die im Prozess wirklich benötigten Informationen dargestellt und manuelle Dateneingaben minimiert werden.

Funktionsbausteine wie Barcode-Scanning, Fotodokumentation per Kamera und das Auslesen von NFC- oder RFID-Chips erleichtern zudem die direkte Datenerfassung vor Ort. Dort, wo komplexe Logik eingebracht werden soll oder besondere Funktionen erforderlich sind, bieten Low-Code-Plattformen in der Regel die Möglichkeit, dies mittels Coding zu ergänzen.

Auf der Backend-Ebene ermöglicht Low-Code eine schnelle und effiziente Integration verschiedener Systeme. Dabei kann eine Low-Code-Plattform wie eine Middleware an vorhandene Systeme angedockt werden. In der Regel stehen hierfür Standardschnittstellen wie REST und SOAP sowie systemspezifische Plug-ins und Konnektoren zur Verfügung, die einen direkten Datenaustausch in Echtzeit ermöglichen. Auch die Anbindung von Maschinen und Anlagen ist möglich, etwa per MQTT und OPC/UA, sodass auch diese in Prozessanwendungen nahtlos eingebunden werden können.

Schließlich bieten Low-Code-Plattformen eine hohe Flexibilität, um die Anwendung auf der Plattform bzw. einem Endgerät zu nutzen, das optimal zum Prozess passt. Gerade in der Produktion sind es meist mobile Endgeräte, die es ermöglichen, dass Informationen an Ort und Stelle im Prozess abgerufen und erfasst werden. Mithilfe von Low-Code-Plattformen sind sowohl Anwendungen für die Desktop-Nutzung als auch für die mobile Nutzung auf Tablet,



Quelle: engomo GmbH

Ob als Web- bzw. Desktop-Anwendung oder als mobile App für Smartphone, Smartwatch und Spezialgeräte: Mit Low-Code profitieren Unternehmen von maximaler Flexibilität bei der Umsetzung prozessoptimierter Lösungen.

Smartphone oder auch auf Industriescannern und Smartwatches möglich – je nach Low-Code-Anbieter sogar als native Apps für die jeweiligen Plattformen; das macht weitere Vorteile nutzbar, wie bessere Usability und einfacheres Handling.

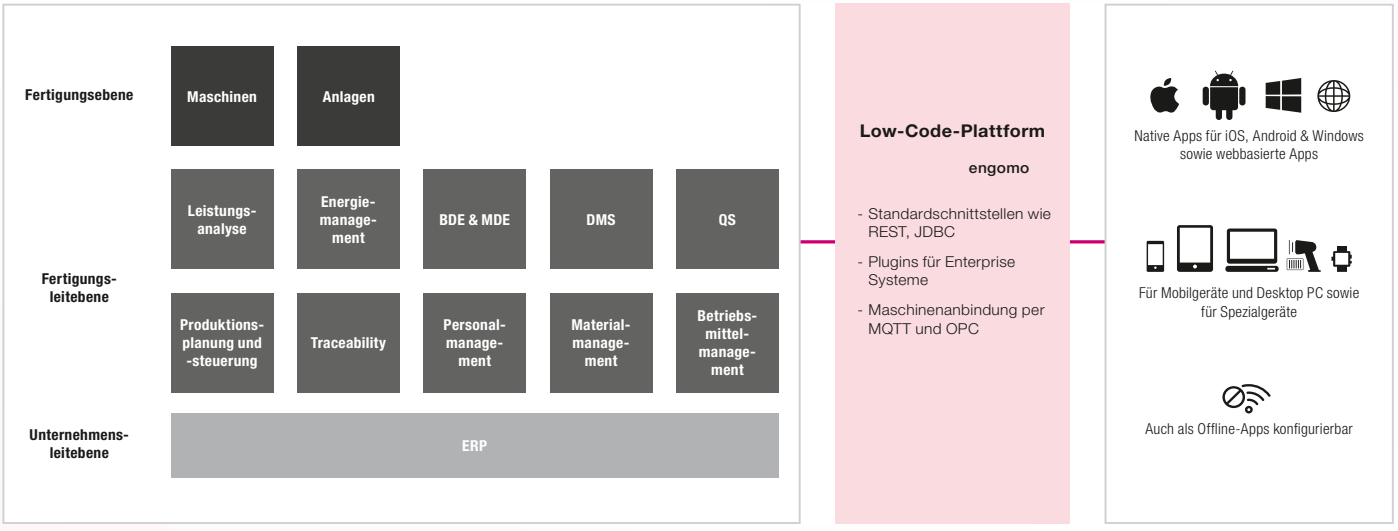
LOW-CODE-ANWENDUNGEN IN DER FERTIGUNGSPRAXIS

In der Praxis gibt es vielfältige Beispiele dafür, wie Low-Code-Apps die Digitalisierung der Produktionsprozesse in Unternehmen voranbringen können:

Ein Unternehmen, das Prüfmittel für die Elektronikindustrie fertigt, verwirklichte mit Low-Code eine umfassende Lösung für die papierarme Produktion – „papierarm" deshalb, weil nach wie vor am Ende des Prozesses Etiketten gedruckt werden, wobei der Etikettendruck dabei auch mobil per Fertigungs-App angestoßen wird. Die umfassende Lösung liefert den Mitarbeitenden in der Produktion mithilfe einer mobilen App alle relevanten Informationen für ihren Arbeitstag: vom Arbeitsvorrat über Fertigungsstatus und Rüstpläne bis hin zur Betriebsdatenerfassung ist die App ihr zentrales Arbeitstool, das auf einem Tablet läuft. Auch die Rückmeldungen zu den Qualitätsprüfungen werden über die Anwendung getätigt, Prüfpläne sind einsehbar und es wird automatisiert gewährleistet, dass in regelmäßigen Abständen QS-Prüfungen durchgeführt werden. Alle erfassten Daten werden automatisiert und in Echtzeit in die jeweiligen, angebundenen Backend-Systeme übertragen.

Ein weiteres Anwendungsbeispiel ist die Realisierung von IoT-Szenarien mit Low-Code-Technologie, wie etwa die Anbindung von Fertigungsanlagen an die Low-Code-Plattform. Per MQTT (Message Queuing Telemetry Transport) können sie per App gesteuert werden, was in der praktischen Anwendung bedeutet, dass ein spezieller Validierungsprozess abgebildet werden konnte, den die Werker vor dem Auftragsstart durchlaufen: Zunächst werden mithilfe der Tablet-Kamera die Barcodes auf den Einsatzmaterialien gescannt und von der Low-Code-App gegen das Backend-System validiert. Nachdem alle zu verwendenden Materialien korrekt gescannt wurden, kann die Maschine dann direkt per Tablet gestartet werden.

Ein anderes, eindrückliches Beispiel für die Realisierung von Prozessoptimierungen mit Low-Code durch System- und Anlagenintegration



Quelle: engomo GmbH

Low-Code ermöglicht die Integration aller Ebenen die in Fertigung und Produktion wirken, und schafft bedarfsgerechte Lösungen für die Anwender im Shopfloor.

stammt aus der Nahrungsmittelindustrie: In einem Betrieb, in dem Nahrungsmittel wie Reformkost und Müsli nach vorgegebenen Rezepturen gefertigt werden, wurde eine Low-Code-App realisiert, die direkt an die verwendeten Industriewaagen sowie an das ERP-System angebunden ist. Die Anwendung führt die Mitarbeitenden anhand der Stückliste durch die Rezeptur, sodass in jedem Schritt erkennbar ist, welche Zutat in welcher Menge benötigt wird.

Beim Abwiegen der Zutaten übermittelt die angebundene Industriewaage das Gewicht in Echtzeit in die Anwendung, sodass dort lediglich noch die Menge bestätigt werden muss. Daraufhin erfolgt automatisch die Materialbuchung im ERP-System. Das Unternehmen berichtet, dass es mit dieser Lösung einen erheblichen Effizienzgewinn erzielen konnte, indem die manuellen Datenübertragungen und Materialbuchungen komplett entfielen.

Schließlich sind auch im Bereich der Qualitätssicherung erhebliche Verbesserungen möglich: Ein Unternehmen im Bereich der Fertigung von Pumpensystemen realisierte mithilfe von Low-Code-Technologie eine papierlose Prüfdatenerfassung: Anstelle der bisherigen Excel-basierten Dateneingabe werden Prüfmerkmale von den Mitarbeitenden per App auf dem Tablet erfasst. Zusätzlich sind einige Messgeräte direkt mit der Low-Code-Plattform gekoppelt, sodass diese Messwerte – wie beispielsweise Oszillographen – direkt an die App übermittelt werden. Alle erfassten Informationen landen in Echtzeit im angebundenen ERP-System, und es wird parallel auch automatisiert das Prüfdaten-PDF generiert.

FAZIT: LOW-CODE FÖRDERT DIE DIGITALE TRANSFORMATION

Die aufgeführten Beispiele zeigen, wie vielfältig die möglichen Lösungen von Low-Code-Plattformen in der Fertigung und Qualitätssicherung sind. Von vollumfänglichen Prozesslösungen bis hin zu schlanken Apps für kleine Teilprozesse können Unternehmen bedarfsgerechte Lösungen realisieren, die ihren Erfordernissen exakt entsprechen. Ein Vorteil ist, dass Low-Code-Plattformen so konzipiert sind, dass auch Personen ohne tiefgehende Programmierkenntnisse Anwendungen (teil-)realisieren können. Das öffnet die Tür für eine engere Zusammenarbeit zwischen IT-Abteilungen und den eigentlichen Anwendern aus den Fachbereichen. Damit können alle Beteiligten gemeinsam Lösungen erarbeiten, die exakt den Anforderungen des Geschäftsalltags entsprechen.

Speziell in mittelständischen Unternehmen, die von begrenzten Ressourcen und Fachkräftemangel geprägt sind, ermöglichen Low-Code-Lösungen einen praxisorientierten und pragmatischen Weg in Richtung Industrie 4.0. Sie erlauben die schnelle Realisierung integrierter und individueller Lösungen und machen IT-Projekte weniger langwierig. Die erstellten Apps sind benutzerfreundlicher als Standardsoftware und können bei Bedarf und bei sich ändernden Prozessanforderungen jederzeit flexibel angepasst werden. Damit stellen sie eine wichtige Ergänzung in der Systemlandschaft produzierender Unternehmen dar, um die Digitalisierung voranzutreiben und Wettbewerbsvorteile zu behaupten. ■

DIE AUTORIN

KERSTIN STIER ist Mitgründerin der engomo GmbH, einer Low-Code-Plattform mit Schwerpunkt im Bereich ERP-Prozesse und Industrie 4.0. Sie ist seit 2012 im Bereich Low-Code spezialisiert und bei engomo für die Bereiche Strategie, Marketing, People & Culture sowie Finanzen verantwortlich. Nach ihrem Studium in International Management war Kerstin Stier zunächst im IT-Projektmanagement bei der Daimler AG tätig und begleitete dann über 8 Jahre hinweg in verschiedenen Rollen ein Software-Unternehmen im Bereich Marketing Automation im Skalierungs- und Internationalisierungsprozess.

DIE FUNKTION VON LOW-CODE AM BEISPIEL DER ERP-ENTWICKLUNG

DER LOW-CODE-ANSATZ UND SEINE ROLLE IN DER DIGITALEN TRANSFORMATION

AUTOR: **UDO HENSEN**

Jahrzehntelang waren Softwarelösungen das Ergebnis intensiver Arbeit qualifizierter Programmierer:innen, die eine Vielzahl von Programmiersprachen nutzten. Die Entstehung komplexer Systeme, wie ERP-Software, war ein langwieriger und kostenintensiver Prozess, in dem hochqualifizierte und gut bezahlte Entwickler:innen eingesetzt wurden, um oft mehrere Jahre an einem Projekt zu arbeiten. Um nach Fertigstellung der ERP-Software viele verschiedene Kunden bedienen zu können, verfolgten die Hersteller die Strategie der maximalen Standardisierung. Dazu wurden möglichst allgemeingültige Prozesse definiert, die auf den ersten Blick für die meisten Unternehmen gleich sind. Lediglich für unterschiedliche Branchen wurden Individualisierungen eingebaut.

Blickt man aber ins einzelne Unternehmen hinein, so zeigt sich, dass sich darin viele einzigartige und teilweise organisch gewachsene Prozesse entwickelt haben. Eine Individualisierung und Anpassung an spezifische Prozesse ist nur über Sonderprogrammierung möglich, bei der das Entwicklungsteam des Herstellers gefragt ist. Ist ein Produkt einmal fertig, ist das Geschäft für die Hersteller einfach. Die Standardsoftware wird durch Updates, Ergänzungen und Erweiterungen lediglich gepflegt, was die Lösungen immer mächtiger, komplexer und anfälliger werden lässt.

Alle großen, bekannten ERP-Lösungen sind Standardlösungen mit einer riesigen Funktionspalette, von der einzelne Unternehmen nur einen Bruchteil tatsächlich benötigen. Teilweise dauern ERP-Projekte mehrere Jahre, weil die Unternehmensprozesse an die Software angepasst werden müssen.

Eine Alternative zu den Aufwänden der klassischen Softwareentwicklung bei ERP-Systemen, stellt nun die Low-Code-Entwicklung dar.

LOW-CODE-PLATTFORMEN

Wer Low-Code entwickeln möchte, nutzt dazu eine Low-Code-Plattform. Die meisten Low-Code-Plattformen werden in der Cloud betrieben und sind als „Software-as-a-Service“ (SaaS) konzipiert. Als Web- oder Cloud-Dienste können diese Plattformen sämtliche Schritte des Low-Code-Development abwickeln: Darstellung, Logik, Business-Integration, Datenverarbeitung und User-Authentifizierung und Testverfahren. Es gibt jedoch auch Low-Code-Plattformen, die On-Premise – also lokal im Rechenzentrum – installiert werden können.

Interessant dabei ist besonders die Art und Weise, wie der Programmcode in diesen Low-Code-Plattformen erzeugt wird.

Fertiger Code „wie von Geisterhand“

Bei einigen Systemen wird im Hintergrund von einem Algorithmus „wie von Geisterhand“ ein Code geschrieben, der kompiliert wird und dann ausführbar ist. Der jeweilige Algorithmus wird dazu einmal entwickelt und bildet danach das Herzstück der Plattform. Hier steckt die Entwicklerarbeit des jeweiligen Herstellers drin.

Rendering on the fly

Hier wird zunächst kein ausführbarer Code generiert, stattdessen werden alle erforderlichen Informationen in Textdateien (zum Beispiel JSON) und einer SQL-Datenbank abgelegt. Wenn die Endnutzer:innen die Webseite aufrufen, interpretiert der Webserver die Informationen und rendert den Code für die Webseite „on the fly“ und sendet diese an den Client zur Darstellung im Browser.

Quelle: GEBRA-IT GmbH

Softwareentwicklung per Drag & Drop

Applikationen bauen per Drag & Drop

Um eine Softwareanwendung zu „bauen“, bedient man sich fertiger Software-Bausteine, die die Low-Code-Plattform bereitstellt. Über eine grafische Oberfläche klicken sich die Anwender:innen aus diesen vielen verschiedenen Bausteinen über Pull-Down-Menüs und per „Drag & Drop“ ihre Anwendung zusammen. Die Entwicklung einer ERP-Lösung nach dieser Methode ist bis zu 10 Mal schneller als die Anpassung einer Standardsoftware. Üblicherweise ist eine erste Version in wenigen Wochen fertiggestellt – wohlgemerkt komplett auf das jeweilige Unternehmen zugeschnitten.

PRAXISBEISPIEL: DIE ENTWICKLUNG VON ERP-LÖSUNGEN MIT LOW-CODE

Low Code Plattformen für den Aufbau von ERP Lösungen einzusetzen, ist noch jung und nicht weit verbreitet. Das mag daran liegen, dass der Markt für ERP-Systeme sehr vielfältig ist und einige Hersteller schon lange am Markt sind. Gleichzeitig sind ERP-Lösungen sehr komplex.

Wie eingangs beschrieben, wurden ihre Lösungen kontinuierlich erweitert, angepasst und dadurch immer komplexer und schwerfälliger. Gängige ERP-Systeme decken heute nahezu alle Standardprozesse im Unternehmen ab, können aber nur schwer „customized" werden. Sie schleppen durch den jahrelangen Aufbau einen Ballast an Code und Funktionen mit, von dem im Einzelfall nur 20 % im Kundenprojekt benötigt werden.

Die Digitalisierung setzt Unternehmen aller Größen und Branchen unter Druck, in einem immer rascheren Tempo auf Marktveränderungen und veränderte Kundenwünsche zu reagieren. Ein modernes, schlankes, flexibles und schnell anpassbares ERP-System ist erforderlich. Hier stoßen viele etablierte Systeme an ihre Grenzen.

MIT LOW-CODE DEN ENTWICKLUNGS-PROZESS NEU DENKEN

Mithilfe moderner, cloudbasierter Low-Code-Entwicklungsplattformen sind Unternehmen jetzt in der Lage, den gesamten Entwicklungsprozess einer ERP-Lösung neu aufzusetzen. Aus einer starren Softwarearchitektur wird so ein modernes und schlankes Baukastensystem. Das Gerüst dieses Baukastens stellen Grundmodule dar, die in jeder ERP-Lösung wiederkehren. Solche Grundmodule sind kleinste Bausteine einer ERP-Software, die man logisch zu einem Prozess verknüpft. Die Umsetzung erfolgt auf Basis eines Grundsystems („Base Template"), das alle grundlegenden Prozesse schon bereithält.

Um die ERP-Software an die einzigartigen Prozesse eines Unternehmens anzupassen, kommen zu den Grundmodulen maßgeschneiderte Bausteine hinzu, die in enger Abstimmung von ERP-Beratern und Kunden zu einer individuellen Lösung zusammengesetzt werden. Wichtig dafür ist das vollständige Verständnis der vorhandenen und der neu zu definierenden Prozesse sowie der zu erfüllenden Wünsche und Erwartungen. Die Anforderungen an das neue ERP-System werden anschließend festgeschrieben und darauf aufbauend die neuen Prozesse vorbereitet. Es werden nur die Dinge angepasst, die genau dieses Unternehmen von anderen unterscheidet. Dabei werden Veränderungen und Ergänzungen durch den Low-Code-Ansatz in kürzester Zeit umgesetzt. Am Ende steht eine dem Kunden sprichwörtlich „auf den Leib geschneiderte" ERP-Lösung zur Verfügung

NACHTEILE VON ERP AUF LOW-CODE

Selbstverständlich ist Low-Code nicht das Allheilmittel. Die neu gewonnene „Freiheit" bei der Entwicklung von Applikationen bedeutet auch Verantwortung. Wer völlig flexibel eine Lösung konstruieren kann, dem kommen immer neue Anforderungen in den Sinn. Es besteht die Gefahr, zu vielen Kundenwünschen gerecht werden zu wollen. Das Beraterteam muss über Führungsqualitäten sowie umfangreiche Prozess- und IT-Kenntnisse verfügen und es ist wichtig, dass zu Beginn eines Projekts die Anforderungen, Ziele und Meilensteine präzise in einem Pflichtenheft festgelegt werden, um das Risiko der Zerstreuung und Ineffizienz zu vermeiden. Für jedes ERP-Projekt einer bestimmten Größe sollte also stets ein ausführliches Lastenheft vorhanden sein.

FAZIT

Low-Code ist ein spannender und vielversprechender Ansatz, um zukünftig zentrale Unternehmensanwendungen schneller und flexibler zu entwickeln als bisher. Gleichzeitig wird Low-Code den klassischen Entwicklungsweg auch nicht gänzlich ersetzen können.

Für Unternehmen, die in neue Anwendungen wie ERP-Software investieren wollen/müssen, stellt der Low-Code-Ansatz eine ernst zu nehmende Alternative zu den klassischen Lösungen dar. Die Vorteile der Low-Code-Entwicklung gegenüber der klassischen ERP-Entwicklung wiegen schwer, die Nachteile sind gut beherrschbar. ■

DER AUTOR

UDO HENSEN, Diplom-Betriebswirt Udo Hensen ist Mitgründer und Geschäftsführer der GEBRA-IT GmbH in Aachen. Mit über 30 Jahren Erfahrung in IT, ERP und Beratung hat er die Firma mit der eigenen Low-Code-Plattform GEBRA-Suite zu einem Vorreiter für flexible ERP-Systeme für KMU und Mittelstand gemacht.

software quality days

EXPERIENCE THE VALUE OF QUALITY

24. - 25. APRIL 2024, WIEN

Europas führende Konferenz für Software Qualität!

- **400+ TEILNEHMER**
- **2 TAGE**
- **5 TRACKS**
- **3 HOCHKARÄTIGE KEYNOTES**
- **70+ VORTRÄGE**

Besuchen Sie die Software Quality Days als Teilnehmer
oder präsentieren Sie Ihr Unternehmen 2024
als Aussteller oder Sponsor!

www.software-quality-days.com

VON DER VISION ZUR UMSETZUNG

REQUIREMENTS ENGINEERING IM ZEITALTER DER LOW-CODE-ENTWICKLUNG

Low-Code-Technologien verändern die Art und Weise, wie wir in Zukunft Anwendungen entwickeln werden. Dank ihnen können auch Personen ohne tiefergehende Programmierkenntnisse ihre Ideen in die Tat umsetzen. Damit dieser Weg von der Vision zum digitalen Produkt gelingen kann, braucht es Menschen, die die Ziele und Anforderungen an das Produkt genau kennen. Welche Rolle wird das Requirements Engineering im Kontext der Low-Code-Entwicklung in Zukunft spielen? Und was ist notwendig, um erfolgreich mit Low-Code zu arbeiten? Darüber hat das SQ Magazin mit Kim Lauenroth und Stan Bühne vom International Requirements Enegineering Board (IREB) gesprochen.

Wie beurteilen Sie die Low-Code-Entwicklung aus der Sicht des Requirements Engineering und des Digital Designs?

Stan Bühne: Grundsätzlich macht es für mich zunächst keinen Unterschied, ob ich Requirements Engineering für ein codebasiertes System mache oder eines, das auf No-Code oder Low-Code baut. Ich muss mich am Anfang genauso fragen, was will ich überhaupt haben? Das heißt für mich aus RE-Sicht: Scope abgrenzen, Stakeholder abholen, Use Cases klären, Qualitätsanforderungen abstimmen. Es wird etwas anders, wenn ich mich im Vorfeld für eine No-Code oder Low-Code Plattform entscheide, denn da habe ich gewisse Einschränkungen, die die Plattform mit sich bringt.

Kim Lauenroth: Ich finde das Thema sehr spannend, weil es eine andere Herangehensweise an die Softwareentwicklung und -gestaltung bietet. Wenn ich eine App entwerfe oder eine betriebliche Anwendung baue, dann wird sie einfach realisiert und umgesetzt. Bei der Low-Code und No-Code Entwicklung denke ich immer auch daran, dass ich meine Kunden befähige, Lösungen weiterzuentwickeln, umzugestalten oder an spezifische Herausforderungen anzupassen. Wenn man sich anschaut, wie viele Unternehmen heute noch mit Excel Probleme lösen, dann könnte das der nächste Schritt sein, das auf eine professionellere Ebene zu heben.

Welche Risiken sehen Sie im Hinblick auf die No-Code bzw. Low-Code Entwicklung?

Kim Lauenroth: Ich habe es sehr häufig erlebt, dass solche Technologien als Ausrede benutzt wurden, die Komplexität auf Seiten der Kunden und Nutzer zu schieben. Schon in den 90er Jahren gab es die Idee der modellbasierten Entwicklung, da besteht immer der Traum, dass Kunden und Nutzer aktiv in die Software eingreifen, sie im Idealfall sogar selbst programmieren und weiterentwickeln können. Aber es besteht sehr häufig ein Missverständnis darin, was Kunden und Benutzer wirklich leisten können und was technisch möglich ist. Da ergibt sich ein Spannungsfeld und das ist eine schöne Brücke zu den Anforderungen: Man muss sich auch im Klaren darüber sein, welche Ansprüche die Nutzer:innen überhaupt an Flexibilität und Anpassbarkeit stellen.

Stan Bühne: Diese Plattformen bieten ein riesiges Universum an Möglichkeiten, aber gleichzeitig schränken sie uns auf das ein, was sie selbst an Funktionalitäten zu bieten haben. Low-Code Anbieter werben oft mit dem Thema der „Citizen Developer“, also dass der Fachbereich sich selbst seine Lösung zusammenbauen kann und dass es IT-Abteilungen zukünftig nicht mehr braucht. Ich weiß nicht, ob das tatsächlich alle leisten können. Einfach zu sagen, hier gibt es diese Wunderwaffe, eine große Box, aus der du dich bedienen und deine Lösung entwickeln kannst, das wird nicht funktionieren.

Kim Lauenroth: Das Spannende ist ja, egal ob man Programmiersprachen bedient oder mit Low-Code arbeitet, man muss über ein gewisses Kompetenzspektrum verfügen. Dafür gibt es in der Bildung mittlerweile den Begriff

„Computational thinking“ – wie ein Computer in Daten und Algorithmen zu denken. Und das ist genau das, was man auch bei Low-Code braucht.

Wie schätzen Sie die Möglichkeiten von Low-Code ein, wenn sich die Anforderungen an ein Projekt ändern?

Stan Bühne: Im Grunde sehe ich das nicht groß anders als bei der traditionellen Entwicklung. Wenn sich Anforderungen ändern und ich zum Beispiel neue Bezahlmöglichkeiten – z. B. Kreditkarte statt PayPal – einführen möchte und in der Low-Code Plattform einen Adapter A gegen einen Adapter B tauschen kann, dann ist das kein Problem. Es kann aber auch durchaus sein, dass die Plattform das nicht hergibt. Dann wird es problematisch, denn dann muss ich anfangen, parallel daneben zu entwickeln oder den Anbieter zu wechseln und das ganze System neu zu bauen. Es kann durchaus einfacher sein, mit Low-Code zu arbeiten, wenn die Plattform die gewünschten Möglichkeiten bietet. Ansonsten sehe ich es ähnlich wie in der klassischen Entwicklung, man sollte sich über Abhängigkeiten von Anforderungen und die Zeitplanung im Vorfeld Gedanken machen. Genauso, wenn Anforderungen oder Tests angepasst werden müssen, egal ob ich nun codiere oder Kästchen schiebe.

Wie sehen Sie die Zukunft von No-Code und Low-Code?

Stan Bühne: Ich denke, dass Low-Code Anwendungen ihren Platz haben werden. Gerade im Bereich Web- und App-Entwicklung sind sie schon weit verbreitet. Auch beim Business Support finde ich Low-Code sehr sinnvoll. Requirements Engineering wird sich in seiner Idee durch Low-Code nicht verändern: Es geht ja darum, herauszufinden, was der Kunde möchte und wie man aus der Vision des Kunden seine echten Bedürfnisse herausfiltert. Was ist wirklich das Problem und wie kann ich für das Problem eine Lösung definieren? Das ist im Wesentlichen, was Requirements Engineering ausmacht, egal wie die Lösung schließlich umgesetzt wird. Low-Code bringt mir vielleicht einen Vorteil, weil ich schnell einen Prototypen erstellen und diesen gemeinsam mit dem Kunden ausprobieren lassen kann. Ansonsten sehe ich für den RE-Bereich tatsächlich keine großen Veränderungen.

Kim Lauenroth: Man muss Low-Code als Material kennen und wissen, dass man Anwender:innen die Möglichkeit geben kann, in das Softwaresystem einzugreifen. In gewissen Größenordnungen, z. B. bei Smart Home finde ich, ist das sinnvoll. Man kann nicht jedes Haus in der Software abbilden und deshalb macht es Sinn, auf solche Lösung zurückzugreifen. Andererseits ist es eine gefährliche Technologie. Man kann unnötig Komplexität an der falschen Stelle auslagern. Das sollte man berücksichtigen. Uns geht es ja auch immer darum, möglichst zügig Feedback von Kunden und Nutzern zu erhalten. Wenn ich ein Instrument habe, mit dem ich einen relativ schnell einen gut funktionierenden Prototyp bauen kann, dann ist das eine tolle Möglichkeit und hilft mir dabei, zügig mit einer Lösung an den Markt gehen zu können. Man sollte die Technologie aber mit Bedacht einsetzen, da sie noch mehr Risiken – oder andere Risiken – hat als die klassische Softwareentwicklung. ■

INTERVIEW-PARTNER:

KIM LAUENROTH (Jahrgang 1977) lehrt und forscht zu Digital Design an der Fachhochschule Dortmund. Bis Ende 2021 war er Leiter des Competence Center für Requirements Engineering bei der adesso SE. Kim hat Informatik, BWL und Psychologie an der TU Dortmund studiert und im Bereich Requirements Engineering für Produktlinien an der Universität Duisburg-Essen promoviert. Im IREB engagiert sich Kim für die Aus- und Weiterbildung im Requirements Engineering und im Digital Design. Weiterhin engagiert er sich im Branchenverband der deutschen Informations- und Telekommunikationsbranche Bitkom für die Etablierung von Digital Design als Gestaltungsprofession der Digitalisierung.

STAN BÜHNE ist Geschäftsführer der IREB GmbH, wo er für den Geschäftsbereich Digital Design verantwortlich ist. In 15 Jahren IT-Beratung, konnte er umfangreiche Erfahrungen in internationalen Digitalisierungsprojekten und in der Telekommunikationsbranche gewinnen. Thematisch beschäftigt sich Stan Bühne seit mehr als 20 Jahren mit den Themen Requirements Engineering und Projektmanagement.

DAS POTENZIAL VON LOW-CODE-PLATTFORMEN: ZWISCHEN MAGIE UND REALITÄT

AUTOR: **ZIED CHTIOUI**

Stellen Sie sich eine Welt vor, in der Softwareentwicklung einfacher als je zuvor ist. Menschen mit unterschiedlichen Hintergründen entwickeln Anwendungen – ganz demokratisch und ohne traditionelle Programmierkenntnisse. Low-Code-Plattformen versprechen, der Schlüssel zu dieser Welt zu sein. Doch sobald die anfängliche Euphorie verfliegt und die pragmatische Realität in den Vordergrund rückt, stellt sich die Frage: Erfüllen diese Plattformen wirklich ihre Versprechen?

Auf den ersten Blick wirkt es wie Magie: Low-Code-Plattformen ermöglichen Nutzern, schnell und einfach Anwendungen zu erstellen. Ein paar Klicks, und schon ist die Anwendung fertig. Doch schaut man genauer hin, stößt man selbst bei hoch entwickelten Low-Code- und No-Code-Anwendungen auf Herausforderungen. Die Probleme reichen von wenig anpassungsfähiger Software bis hin zu unhandlichen Plattformen. Ohne ein grundlegendes Verständnis für Codierung sind Unternehmen an bestimmte Anbieter gebunden oder auf Plattformen festgelegt, was ihre Innovationsfähigkeit einschränkt. Dieses Szenario spiegelt das Konzept der 80/20-Regel wider. Demnach sind etwa 80 % aller Auswirkungen in 20 % der Ursachen begründet. Im Zusammenhang mit Low-Code-Plattformen lassen sich leicht Parallelen ziehen: Die Plattformen können eine schnelle Entwicklung ermöglichen, die einen erheblichen Teil der typischen Anwendungsanforderungen abdeckt, was den 80 % entspricht. Die verbleibenden 20 % der benutzerdefinierten und komplizierten Funktionen erfordern jedoch oft die Fähigkeiten erfahrener Entwickler:innen, ein durchdachtes Design und eine strategische Architektur.

Die Softwarearchitektur dient in diesem Zusammenhang als Brücke zwischen beiden Bereichen. Eine robuste Architektur kann so entworfen werden, dass sie mit Low-Code-Plattformen harmoniert und eine Basis für effiziente, anpassbare Anwendungen bildet. Selbst in einer Welt technologisch fortschrittlicher Low-Code-Plattformen können aber nicht alle Qualitätsanforderungen gleichzeitig erfüllt werden, da für jede Plattform ein anderer Architekturansatz erforderlich ist.

Da die anfängliche Begeisterung für Low-Code-Plattformen nachlässt und an der ein oder anderen Stelle Kritik laut wird, ist es wichtig mit einem differenzierten Blick auf die Technologie zu schauen: Ja, Low-Code vereinfacht die Softwareentwicklung, beschleunigt die Markteinführung und steigert die Effizienz. Wenn Sie jedoch wichtige Teile in Ihrer Software selbst entwickeln, bleiben Sie flexibel und können jederzeit optimieren. Mit diesem Ansatz minimieren Sie auch die Risiken, die mit Low-Code in Verbindung stehen und gehen auf die spezifischen Anforderungen Ihres Unternehmens und Marktes ein, während gleichzeitig eine starke, anpassungsfähige Architektur erhalten bleibt.

Seien Sie daher kritisch gegenüber Versprechungen von „einfachen Lösungen“ in der Technologie, selbst wenn Low-Code- und No-Code-Plattformen heute hohe Entwicklungsstufen erreicht haben. Nutzen Sie das Potenzial dieser zur Unterstützung Ihrer Ziele, aber tun Sie es in Maßen und behalten Sie die Beherrschung der Grundlagen bei. Mit einem soliden Fundament aus Programmierkenntnissen, architektonischer Weitsicht und zielgerichteten Entscheidungen können Sie auch in der sich schnell entwickelnden Technologielandschaft dauerhaft erfolgreich sein.

DER AUTOR

ZIED CHTIOUI ist ein leidenschaftlicher, moderner und kreativer Softwarearchitekt, Projektmanager und iSAQB Accredited Trainer mit langjähriger Erfahrung in den Bereichen Automatisierung von Fondsbuchhaltungsprozessen, Fonds-Compliance, Spitzentechnologien und Produktinnovation/Transformationsstrategien.

KOLUMNE: QUALITÄT ALS HALTUNG

LOW/NO-CODE – BRAUCHT NOCH WER ´NEN TESTER?

„No-Code verlagert nur das Thema Testen. Qualitäts-sicherung bleibt notwendig!“

DER AUTOR:

RICHARD SEIDL ist Software-Test-Experte, Agile Quality Coach und Autor. Er hat in seiner beruflichen Laufbahn schon viel Software gesehen: gute und schlechte, große und kleine, neue und alte. Software so schön, dass man weinen könnte und auch solche wo es Fußnägel aufrollt. Für ihn ist klar: Wer heute exzellente Software kreieren möchte, denkt den Entwicklungsprozess ganzheitlich: Menschen, Kontext, Methoden und Tools – erst wenn alles zusammenspielt, entsteht ein Mindset für Potentialentfaltung und Innovation.

Low- und No-Code-Plattformen werden seit einigen Jahren immer populärer. Zumindest in den Medien. Die Meinungen darüber sind breit gefächert: Von „Alter Wein in neuen Schläuchen“ bis „Die Software-Revolution“ ist alles dabei. Die Lösungsversprechen sind irgendwie die gleichen wie auch schon bei allen anderen Software-Revolutionen: Kosteneffizienz, schnelle Markteinführung, Flexibilität und einfache Umsetzung.

Die Frage nach der Sinnhaftigkeit für den Einsatz von Low/No-Code lässt sich mit dem meist-verwendeten Berater-Zitat klären: „Kommt drauf an.“ Es hängt von vielen Faktoren ab. Das müssen wir genau prüfen! Denn wo Licht ist, ist auch Schatten. Und natürlich wünschen wir uns für unsere mittlerweile fachlich und technisch hochkomplexe Software-Welt einfache Lösungen und Abstraktionen. Wir wollen Komplexität beherrschbar machen.

WIE SIEHT ES MIT DER QUALITÄTSSICHERUNG AUS?

Aber wie sieht es denn mit Testing und Qualität aus? Ich hörte schon mal, dass man bei No-Code nicht testen muss. Nun, das sehe ich anders. Für die unteren Teststufen können wir uns ja auf die Hersteller der Plattform verlassen (räusper, hüstel). Aber die Qualitätssicherung auf den höheren Teststufen bleibt. Und hier vor allem Fragen wie:

- Ist eine passende Architektur für die Prozesse/Oberflächen gewählt?
- Erfüllen die Abläufe die fachlichen Anforderungen?
- Sind die Module sinnvoll genutzt?

Hä? Das sind doch die gleichen Fragen wie immer. Ja. Denn wir verlagern hier ja die „Entwicklung“ nur auf eine höhere Abstraktion. Statt Code basteln wir Module, Oberflächen, Prozesse zusammen. Und auch hier gibt es eine Architektur, eine Entwicklung und einen Test. Meiner Meinung nach spricht sogar vieles hier für einen sehr ordentlichen Test. Denn der „Citizen developer“ kann ja jeder sein. Und dadurch ändern sich oft auch die Rollen. Ein Beispiel: ein Business Analyst, der jetzt, statt die Eigenentwicklung zu testen und abzunehmen, selbst beginnt, seine Applikationen zusammen zu klicken. Ist ja praktisch, er hat ja das fachliche Know-How. Aber ich unterstelle hier mal einfach ein paar nicht so tief ausgeprägte Skills:

- Der BA ist kein Software-Architekt. Er wird nicht im Raum von Software-Design, Entitäten, Datenflüssen und Wiederverwendbarkeit denken, sondern eher nach Gutdünken, Module, Oberflächen und Sonstiges zusammenstellen.
- Er ist auch kein Softwareentwickler. Ihm fehlen wahrscheinlich die Paradigmen, Pattern und Best Practices wie ein Software-Fluss bestmöglich funktioniert
- Außerdem fällt er als Tester für seine eigene Umsetzung raus. Sprich Tests und die fachlichen Abnahmen müssen anderswo stattfinden.

Software ist Software. Jede anders. Low-Code heißt nicht Low-Test und No-Code schon gar nicht No-Test. Aber halt anders.

Ihr Richard Seidl

DIE KUNST DER AGILEN ENTWICKLUNG

Das Buch von James Shore war bereits in der ersten Auflage ein umfassendes Standardwerk zum Thema. Die zweite Auflage ist noch umfangreicher und basiert auf weiteren 14 Jahren Erfahrung des Autors und der Weiterentwicklung des agilen Arbeitens. James Shore hat das Werk um neue Themen erweitert sowie die bereits in der ersten Auflage enthaltenen Praktiken überarbeitet und auf den aktuellen Stand gebracht.

Daten zum Buch:
James Shore
Die Kunst der agilen Entwicklung
Grundlagen, Methoden und Praktiken

2023, Übersetzung der 2., vollständig überarbeiteten Auflage, dpunkt.verlag, 744 Seiten
ISBN 978-3-86490-860-6

Der Anspruch und die gewünschte Wirkung des Buches sind gleich geblieben: Es will nicht nur den aktuellen Stand der agilen Entwicklung vermitteln, sondern die Leser:innen in die Lage versetzen, selbstständig agile Praktiken in ihre Teams einzuführen oder diese zu verbessern – auch im Hinblick auf die das Team umgebende Organisation.

Es gehört einiges dazu, die „Kunst der agilen Entwicklung" zu meistern. Von den technischen Details der testgetriebenen Entwicklung (mit Codebeispielen) bis hin zur Entwicklung einer agilen Unternehmenskultur deckt das Buch alle Themenkomplexe und notwendigen Fähigkeiten ab.

Dabei hat der Inhalt eine unübertroffene Qualität in den Details. Das liegt daran, dass der Autor das Buch inkrementell entwickelt hat. Während des Schreibprozesses hat er die vorläufig fertigen Kapitel veröffentlicht und seine Community um Feedback gebeten. Da James Shore zur ersten Generation der agilen Vordenker gehört, besteht seine Community aus vielen Menschen, die das Thema agile Entwicklung entscheidend geprägt haben und noch immer prägen. Dieses wertvolle Feedback hat er in die Kapitel einfließen lassen. Das Ergebnis ist ein Kompendium, das multiperspektivisch alle Aspekte der agilen Entwicklung beleuchtet. Kein Wunder, dass hier ein „dicker Schinken" entstanden ist.

Ein großer Mehrwert der zweiten Auflage ist die Integration des „Agile Fluency"-Modells. Dahinter verbirgt sich, kurz gesagt, die Erkenntnis, dass die Art und Weise, wie tief agiles Arbeiten in einem Team oder einer Organisation etabliert wird, davon abhängen sollte, was die Organisation benötigt, um in ihrem Kontext erfolgreich zu sein. Das Modell gibt dem Buch seine Struktur. Die zahlreichen agilen Praktiken sind den unterschiedlichen Zonen des Modells (Fokus auf den Geschäftswert, zuverlässige Lieferung und Optimierung der Ergebnisse) zugeordnet. Durch Querverweise werden die Themen zueinander in sinnvolle Beziehung gebracht. So ist es möglich, mit der Lektüre an einer beliebigen Stelle zu beginnen und sich dann über die Querverweise weiterleiten zu lassen. Beim Lesen des Abschnitts „Aufgaben anlegen" verweist das Buch beispielsweise auf die verwandten Themen „Kapazität", „Visuelles Planen" und „Das Planning Game".

Es lohnt sich, das Buch von vorne zu lesen. Hier wird der große Bogen gespannt, was der Auslöser für die agile Bewegung war und warum es ihr gelungen ist, die Softwarewelt zu erobern. Agiles Arbeiten ist heute Standard, aber nur wenige wissen, wie es richtig funktioniert. Wer dieses Buch liest, kommt dem ein großes Stück näher. ■

DER AUTOR:

MARCO BONK ist Berater und Trainer für agile Wertschöpfung. Tiefe Erfahrung mit Produktentwicklung und Projektmanagement hat er bei einem Verlag gesammelt, wo er im Zuge der Digitalisierung damit konfrontiert war, neue Lösungen für neue Probleme zu finden. Dabei ist seine Leidenschaft für Teamarbeit entstanden. Komplexe Herausforderungen mit einem motivierten selbstorganisierten Team zu meistern ist seine Berufung. Seit 2015 beschäftigt er sich intensiv mit agilem Arbeiten, zunächst als Scrum Master in der Logistikbranche, seit 2018 als Berater bei it-agile.

Das SQ-Quiz

GESPONSERT VOM dpunkt.verlag

In Kooperation mit www.Sudoku-Aktuell.de

5		7	3				8	
		1	5				6	
		2	7			3	4	
				2				8
	1	8		6				2
	5							
					3	6		
7						1		
3			2		9			

Buchstaben: **1=S, 2=I, 3=D, 4=R, 5=P, 6=U, 7=O, 8=T, 9=N**

LÖSUNGSWORT

Sudoku

Die Lösung des letzten Sudokus lautete:

GENERATIVE AI

Die Gewinnerinnen und Gewinner der Auslosung Sudoku SQ Magazin 66 sind:

Anne Kramer, Erlangen
Andreas Koch, Rottweil
Moritz Salein, Eschborn
Hans Greiner, Berlin
Daniel Siegrist, Bruchsal

Zu gewinnen!

„Toolkit für Agile Coaches und Scrum Master“
von Charlotte Bendermacher/Cleo Kampschuur/Rini van Solingen

Viele Organisationen gehen zu einer agilen Arbeitsweise über. Oft werden dazu Scrum, Kanban & Co. eingeführt. Diese Frameworks bringen jedoch nicht automatisch Agilität mit sich, denn den Kern des agilen Arbeitens bilden immer die Menschen und ihre Interaktionen. Doch welche Herausforderungen erwachsen daraus in der Praxis? Und wie geht man am besten damit um? Dieses Arbeitsbuch für Agile Coaches und Scrum Master enthält 99 praxiserprobte Übungen und Workshop-Formate für die vielfältigen Aspekte agilen Arbeitens.
Kurze Interventionen wie verschiedene Energizer- und Kennenlernübungen, konkrete Praktiken, Formate und Simulationen, deren Ablauf jeweils kurz und prägnant Schritt für Schritt vorgestellt wird, helfen dabei, schnell und gut vorbereitet loszulegen.
Das Buch ist Werkzeugkasten und Nachschlagewerk zugleich.

Dieser Buchpreis wird gesponsert vom **dpunkt.Verlag**
Senden Sie bis zum 1. Februar 2024 das Lösungswort aus dem SQ-Quiz an *info@asqf.de* und gewinnen Sie eines von 5 Büchern.*

*Der Rechtsweg ist wie immer ausgeschlossen. Die MitarbeiterInnen der iSQI GmbH und des ASQF e.V. sowie sämtliche am Gewinnspiel beteiligten Personen sind von der Teilnahme ausgeschlossen. Teilnehmende erklären sich mit der Veröffentlichung ihres Namens in der Folgeausgabe einverstanden.

SQ MAGAZIN №68

Erscheint im März 2024

SQ №68
Thema: Neue Perspektiven im Projekt- und Prozessmanagement
Anzeigenschluss: 1. Februar 2023
Redaktionsschluss: 1 Februar 2023

Im nächsten Heft:

EFFIZIENZ IM FOKUS: NEUE PERSPEKTIVEN IM PROJEKT- UND PROZESSMANAGEMENT

Um Projekte erfolgreich zu realisieren, braucht man optimale Prozesse, die effizient und leicht implementierbar sind. In der Softwareentwicklung hat sich die agile Entwicklung seit einigen Jahren als Standard etabliert, aber auch das sequentielle Modell erfreut sich nach wie vor großer Beliebtheit. Jedes dieser Modelle hat seine eigenen Stärken und Schwächen.

In der nächsten Ausgabe des SQ Magazins beschäftigen wir uns mit der Frage, **wie effektives Projekt- und Prozessmanagement in der Praxis gelingen kann**. Wie können schlanke, wirkungsvolle Prozesse etabliert werden, die den Produktionsablauf erfolgreich machen? Welche Werkzeuge können dabei helfen? Wir laden Sie ein, Ihre Erfahrungen zu teilen – senden Sie uns Ihre Beiträge bis zum 01. Februar 2024* an **redaktion@sq-magazin.de**

***Um eine Voranmeldung der Artikel mit ungefährem Umfang wird bis 15.01.2024 gebeten.**

IMPRESSUM

HERAUSGEBER
iSQI GmbH
Voltastraße 1, 14482 Potsdam
TEL +49 331 231810-0
FAX +49 331 231810-10
info@asqf.de, www.isqi.org

REDAKTION
V.i.S.d.P.: Stephan Goericke CEO
Chefredaktion: Eva Hofmann
Redaktion: Anna-Christina Feldhusen

Voltastraße 1, 14482 Potsdam
TEL +49 331 231810-18
FAX +49 331 231810-10
redaktion@sq-magazin.de,
www.sq-magazin.de

SATZ/LAYOUT
FORMAT Druck und Medienservice GmbH
www.arnoldgroup.de

FOTOS: ASQF e.V. und iSQI GmbH
Editorial-Bild: Silke Weinsheimer
Titelseite: pixabay
Alle Portraits und Grafiken mit freundlicher Genehmigung der Autoren.

DRUCK: ARNOLD group
DRUCKAUFLAGE: 4.000 Stück
INTERNETAUSGABE: www.sq-magazin.de

MEDIADATEN
Gern senden wir Ihnen unsere Mediadaten zu. Richten Sie Ihre Anfrage an werben@sq-magazin.de

Namentlich gekennzeichnete Beiträge müssen nicht mit der Meinung der Redaktion übereinstimmen. Die Redaktion behält sich das Recht auf sinngerechte Kürzung und Bearbeitung eingereichter Manuskripte vor. Wir machen darauf aufmerksam, dass Daten nicht an Dritte weitergegeben und ausschließlich zur internen Auswertung herangezogen werden können.

Die Redaktion achtet darauf, alle Geschlechter gleichermaßen auf faire Weise anzusprechen.

A4Q Certified Selenium 4 Tester Foundation

Mach dich fit für die Testautomatisierung mit Selenium!

Jetzt kostenlos registrieren und Online-Probeprüfung ablegen!

iSQI®
SUCCESS IS YOURS