

Abo Friederike Krieger

Qualitätsjournalismus kostet Geld. Mit Ihrem Abo sorgen Sie dafür, dass unsere Berichterstattung unabhängig bleibt.

Copyright © 2020 Versicherungsmonitor. All rights reserved.

Alleskönner oder alter Wein in neuen Schläuchen?

geschrieben von [Sascha Däsler](#) und [Martin Rybicki](#) am 22. Oktober 2020 in [Abo](#), [Allgemein](#), [Kommentare](#), [The Long View](#), [Top News](#) | [Keine Kommentare](#) | [Drucken](#)

The Long View – Der Hintergrund Low-Code- und No-Code-Lösungen, die auf einen visuellen Entwicklungsansatz statt auf traditionelles Programmieren setzen, sind in aller Munde. Ist dies nun endlich die Technologie, die den komplexen Herausforderungen der Versicherungsbranche gewachsen ist, mit beeindruckender Effizienz und ohne funktionale Einschränkungen? Der erste Blick besticht: Statt individueller Programmcodes kommen nun standardisierte Softwaremodule zum Einsatz. Low-Code- und No-Code-Lösungen allein sind aber nicht die Antwort auf alles.



Sascha Däsler ist Verantwortlicher Partner bei PPI.X

© PPI

großen Stück hinter sich gelassen. Inzwischen setzen sie entweder auf Kaufprodukte oder einen Mix aus Kauflösung und Eigenentwicklung. Bei den Kernsystemen liegen die Vorteile von Standards auf der Hand: Verteilung der Weiterentwicklungsaufwände auf mehrere Schultern, günstige Wartung, einfache Modularisierung und Skalierung. Wozu das Rad neu erfinden?

Dagegen bleibt die Kundenschnittstelle immer eine besondere, individuelle, exakt auf die Bedürfnisse des Versicherers zugeschnittene Komponente. Und die ließ bislang keine Standards zu. Versicherer fragen sich zu Recht: Ist die Individualität im Austausch mit dem Kunden der erfolgskritischste Faktor? Wie sehr muss dabei auf die Kundenbedürfnisse eingegangen werden? Welche Aufwände sind gerechtfertigt? Eine pauschale Antwort zu erwarten, wäre vermessen. Fakt ist, die Schnittstelle zum Kunden ist ein Fokusthema, das entsprechende Aufmerksamkeit verlangt. Konkret benannt: Kernsysteme wie die Bestandsverwaltung lassen sich gut hoch automatisiert abbilden. Anders ist die Situation beim Kundenportal, etwa im Fall der Antragsstrecke. Diese Komponente gilt es, genau auf die Bedürfnisse der Kunden beziehungsweise das Alleinstellungsmerkmal (Unique Selling Point, USP) des Versicherers abzustimmen.

Ausgereifte Lösungen – tauglich für alle Fälle?

So ein Heilsbringer kommt sehr gelegen. Die Versicherungsbranche ist in einer intensiven Investitionsphase, die durch die Covid-19-Pandemie noch befeuert wird. Der aktuelle Markt verlangt, Kunden digital, individuell und mit höchstmöglichem Wiedererkennungswert über die unterschiedlichsten Kanäle zu erreichen. Gleichzeitig steigen der Kostendruck sowie der Bedarf nach schlanken, standardisierten unternehmensinternen Prozessen.

Einige Versicherer haben ihre alten Mainframe-Systeme schon zu einem

Hersteller wie Microsoft, Amazon und Co. haben ihre Hausaufgaben gemacht. Sie bieten heute im Low-Code-/No-Code-Segment neben nahezu unbegrenzter Skalierbarkeit, der Integration von künstlicher Intelligenz und einer unglaublichen Time-to-Market verschiedenste weitere Vorteile. Dazu zählen eine einfache und fachgerechte Modellierung von Daten, abgeleitet von Standardmodellen, ebenso wie eine gewisse Unabhängigkeit von der IT. Das alles sind starke Argumente. Selbstverständlich hat jeder Ansatz aber auch seine Schwachpunkte. Daher sind individuelle Bewertungen des Vorhabens, die Betrachtung des Einsatzfeldes und die Art der Lösung eine grundlegende Voraussetzung für den Projekterfolg.

Den Vorteilen von Low-Code-/No-Code-Apps stehen aber auch Einschränkungen gegenüber, etwa wenn die IT-Verantwortlichen bei Fragen angefragt werden wie:

- Wer wartet die Applikation?
- Wer ist Ansprechpartner beim Auftreten von Fehlern?
- Wie integriert sich die Applikation mit den vorhandenen Stammsystemen?
- Wer optimiert und kontrolliert den Code?
- Funktionierte die Anwendung beim nächsten automatischen Update immer noch wie gewohnt und sieht sie auch noch so aus?
- An wen kann ich mich wenden, wenn auf einmal nichts mehr geht?

Diese Punkte sind nicht aus der Luft gegriffen. Schon vor Jahren hat die PPI AG Erfahrungen mit Low-Code-/No-Code-Plattformen gemacht. Namhafte Hersteller wie Oracle haben versucht, uns das Leben zu erleichtern, indem sie einen fertigen Baukasten mit schicken Funktionen zur Verfügung gestellt haben. Alles getrieben vom Versprechen, mit nur wenigen Klicks Anwendungen live zu bringen. Im Fokus stand beispielsweise der Einsatz für einfachste Datenvisualisierungen, Eingabehilfen und Formulare. Aber: Erforderte die Projektaufgabe, sich



Martin Rybicki ist Leiter Softwareentwicklung bei PPI.X

© PPI

vom Standard zu entfernen, etwa weil Logos ausgetauscht oder PDFs generiert werden sollten, wurden für diese Low-Code-/No-Code-Apps letztendlich doch Programmierexperten benötigt.

Bei Fehlern sind Experten gefragt

Doch warum erfährt die altbekannte Baukastentechnik gerade heute eine solche Renaissance? Hat sich etwas Entscheidendes geändert oder geht es eher um alten Wein in neuen Schläuchen? Stellen wir uns ein kleines Szenario vor: Sie haben den Auftrag, einen Tarifrechner auf Ihrer Portallösung zur Verfügung zu stellen. Ihre Produktmaschine befindet sich in Ihrem Backend, und die Eingaben, die zur Preisberechnung nötig sind, werden über eine Webseite abgefragt. Die Darstellung der Oberfläche ist schnell mit Low-Code-/No-Code-Apps gemacht. Nun gilt es, die Informationen, wie es der Service erwartet, in die Produktmaschine zu transportieren. Und die Antwort muss das System entsprechend darstellen, damit der Kunde auch den richtigen Preis bekommt. Hier wird es schon schwieriger. Spätestens, wenn die ersten Fehler (Exceptions) auftauchen, weil zum Beispiel Feldlängen überschritten wurden oder Authentifizierungen nicht geklappt haben, sind Experten gefragt, die solche Funktionalitäten integrieren können.

Gibt es nun aber einen systematischen Weg, die Vorteile der Low-Code- und No-Code-Anwendungen mit den Vorzügen der Individualentwicklung zu verheiraten? Aus unserer Sicht ja. Die Lösung: Framework Driven Individual Development.

Das Zusammenspiel zweier Welten

Sie bietet Individualität bei schneller Time-to-Market. Die Skalierbarkeit wird dabei durch Cloud-Ansatz – also Einbindung von Diensten aus der Cloud – gewährleistet. Dieser Entwicklungsansatz verbindet die Flexibilität von individuellem Code mit den Vorteilen eines Frameworks wie der beliebigen Wiederverwendung vorprogrammierter Module oder genormter Schnittstellen. Framework Driven Individual Development ist sowohl in die bestehende Systemlandschaft als

auch in Servicearchitektur und Geschäftsprozesse (Operations) hervorragend integrierbar. Die vorgefertigten Module beruhen auf bekannter Middleware, also Programmen, die zwischen Anwendungen vermitteln, und garantieren einen modernen und effizienten Ansatz sowie einen schnellstmöglichen Rendite auf das eingesetzte Kapital, der auch über Jahre beibehalten werden kann.

Launchpad realisiert Framework Driven Individual Entwicklung

In der Praxis liefert ein Launchpad, also eine Software-Collaboration-Plattform, die ideale Basis, um Initialisierungsaufwände zu minimieren und technische wie fachliche Features sofort nutzbar zu machen. Im praktischen Einsatz liegt die erste Aktivität eines Entwicklers darin, in seinem Projekt die Fachlogik zu implementieren. Dabei kann er in hohem Maß auf modulare Komponenten zurückgreifen. Das reicht von der Architektur über Standard-Features bis zur Testumgebung.

Im Projektalltag lassen sich mit einem Launchpad kleine wie große IT-Vorhaben stemmen, und das mit geringen Rüstzeiten. Das komplette Projekt-Setup steht automatisiert auf Knopfdruck zur Verfügung. Das Risiko der alleinigen Bindung an bestimmte Hersteller – sogenanntes Vendor Lock-in – wird durch die Nutzung von etablierter Open-Source-Technologie vermieden. Der Austausch von Komponenten auch von unterschiedlichen Herstellern bleibt möglich. Dies ist sinnvoll, da der Werkzeugkasten (Tech-Stack) des Launchpads nicht statisch ist, sondern sich mit der Weiterentwicklung der Technik wandelt. Anwender haben so die Sicherheit, stets auf aktuelle Werkzeuge wie Programmiersprachen, Qualitätssicherungs-Tools, Datenbanken und Frontend-Werkzeuge zurückgreifen zu können. Schließlich sichert ein Launchpad die Codequalität durch Kontrolle von Beginn an, inklusive automatisch generierter Testreports.

Hybrider Ansatz ist das Optimum

Die Verbindung des Besten aus den Welten der Individualplattformen und der Low-Code-/No-Code-Ansätze durch Einsatz eines Launchpads ist angesichts der Anforderungen an Versicherungsunternehmen nach unserer Meinung unvermeidlich. Gerade bei der in der Branche so häufig benannten und unbedingt notwendigen individuellen Kundenschnittstelle kristallisiert sich ein hybrider Ansatz als Optimum heraus: Die Eigenentwicklung mit all ihren Vorteilen nutzen, dabei auf bestehende, standardisierte Komponenten zurückgreifen.

Effizienzorientiert und individuell sollte das Credo sein. Low-Code/No-Code ist folgerichtig nicht die Antwort auf alles. An vielen Stellen kann und wird die Baukastenidee individuell geschriebene Software nicht verdrängen. Die Lösung ist ein hybrider Ansatz: Wenn individuell, dann bitte auch effizient mit Framework-Ansätzen.

Sascha Däsler ist Verantwortlicher Partner bei PPI.X und Martin Rybicki Leiter Softwareentwicklung. PPI.X ist das Collaboration-Lab des Hamburger Beratungs- und Softwarehauses PPI AG.

In der Kolumne „The Long View“ laden wir Branchenexperten ein, über Hintergründe zur Lage der Branche zu schreiben.

Dieser Text ist nur für Abonnenten von Herbert Frommes Versicherungsmonitor Premium persönlich bestimmt. Das Weiterleiten der Inhalte – auch an Kollegen – ist nicht gestattet. Bitte bedenken Sie: Mit einer von uns nicht autorisierten Weitergabe brechen Sie nicht nur das Gesetz, sondern sehr wahrscheinlich auch Compliance-Vorschriften Ihres Unternehmens.

Beitrag gedruckt von Herbert Frommes Versicherungsmonitor: <https://versicherungsmonitor.de>

URL zum Beitrag: <https://versicherungsmonitor.de/2020/10/22/alleskoenner-oder-alter-wein-in-neuen-schlaeuchen/>