

# Apps entwickeln mit SAP Build Apps

Das Praxishandbuch

» Hier geht's  
direkt  
zum Buch

# DAS VORWORT

# Einleitung

Unter No-Code-Entwicklung versteht man die Erstellung von Softwareanwendungen ohne das Schreiben von Code. Stattdessen können die Benutzer\*innen Software mithilfe visueller Drag-&Drop-Schnittstellen und vorgefertigter Vorlagen erstellen. Dies verkürzt die Zeit, die für die Entwicklung benötigt wird, erheblich und ermöglicht gleichzeitig auch Personen ohne Programmiererfahrung, einfache Softwareanwendungen zu erstellen. Die No-Code-Entwicklung kann Unternehmen dabei helfen, effizienter und kostengünstiger zu werden und schnell auf sich ändernde Geschäftsanforderungen zu reagieren.

No-Code-  
Entwicklung

Die No-Code-Entwicklung findet mittlerweile in vielen Unternehmen Einzug. Dabei wird Ihnen häufig der Begriff *Citizen Developer* begegnen. Citizen Developer sind nicht-professionelle Softwareentwickler\*innen, die Softwareanwendungen mithilfe von Low-Code- oder No-Code-Entwicklungsplattformen erstellen oder anpassen. Wie der Name es schon suggeriert, können diese Plattformen genutzt werden, ohne selbst Code schreiben zu müssen, so dass auch Personen ohne Programmiererfahrung Softwareanwendungen entwickeln können. Citizen Developer sind in der Regel Geschäftsanwender\*innen aus dem Fachbereich oder IT-Fachleute, die einen Bedarf an Software haben, der durch Standardlösungen nicht gedeckt wird.

Citizen Developer verfügen meist über bereichsspezifisches Wissen, sogenanntes Domänenwissen, das sie dazu befähigt, Software zu entwickeln, die den spezifischen Anforderungen ihres Unternehmens entspricht. Sie können Lösungen entwickeln, die Geschäftsprozesse automatisieren, in bestehende Systeme integrieren und den Datenfluss verbessern. Dies macht sie zu einem wertvollen Aktivposten für Unternehmen, da sie dazu beitragen können, die Effizienz zu steigern und Prozesse zu rationalisieren.

Citizen Developer werden immer wichtiger, da Unternehmen nach Möglichkeiten suchen, auf sich ändernde Bedingungen schnell zu reagieren und dabei leistungsfähig und kostengünstig zu bleiben. Indem sie bürgerliche Entwickler\*innen zur Erstellung von Software befähigen, können Unternehmen ihre Abhängigkeit von professionellen Entwicklern verringern, die Entwicklungszeiten verkürzen und die Abstimmung zwischen der IT und den Fachabteilungen verbessern.

Wir richten uns mit diesem Titel bewusst an Citizen Developer. Sie benötigen keine Programmiererfahrung, um dieses Buch zu lesen, die Inhalte zu verstehen und die Übungen nachzubauen. Sie sollten jedoch ein technisches Grundverständnis mitbringen.

Zielgruppe

Eine der beliebtesten No-Code-Entwicklungsplattformen ist *SAP AppGyver*, die unter dem Namen *SAP Build Apps* mittlerweile fester Bestandteil des SAP-Universums ist.

## Aufbau des Buches

**Teil I** Dieses Buch ist in drei Teile gegliedert. In **Teil I**, »No-Code-Entwicklung mit SAP Build Apps«, beschäftigen wir uns mit der No-Code-Entwicklung und mit SAP Build Apps im Allgemeinen. Wir starten in **Kapitel 1**, »Überblick«, mit einer allgemeinen Einführung in das Thema No-Code-Entwicklung. Danach gehen wir auf die Anforderungen an No-Code-Entwickler\*innen ein und erklären Ihnen anschließend, welche Rolle SAP Build Apps in diesem Zusammenhang spielt. Zusätzlich grenzen wir die Low-Code-Entwicklung auf Basis von SAP Fiori Elements von der No-Code-Entwicklung ab.

Nachdem Sie als Citizen Developer die Entwicklerwerkzeuge kennen sollten, erhalten Sie in **Kapitel 2**, »Einführung in die Entwicklertools von SAP Build Apps«, eine Einführung in die Entwicklertools der Lösung. SAP hat sich hier vor kurzem von der SAP-AppGyver-Plattform verabschiedet und die Werkzeuge voll in die SAP BTP integriert.

Danach werden Sie die Entwicklerwerkzeuge in **Kapitel 3**, »Ihre erste Business-App«, erstmals in Aktion sehen und Ihre erste Business-App entwickeln. Nachdem Sie dieses Kapitel gelesen und gegebenenfalls die App erstellt haben, werden Sie uns hoffentlich zustimmen, dass No-Code-Entwicklung kein Mythos ist, sondern tatsächlich funktioniert.

In **Kapitel 4**, »Für Admins: SAP Build Apps einrichten«, wird es dann technisch. Damit Sie SAP Build Apps in der täglichen Arbeit verwenden können, muss das Entwicklerwerkzeug auch der IT-Landschaft Ihres Unternehmens bereitgestellt werden. Wir zeigen daher, wie Administrator\*innen SAP Build Apps in der SAP BTP einrichten. Dazu gehören auch die Themen Sicherheit und Berechtigungen, denen wir ebenfalls einen Abschnitt widmen.

**Teil II** In **Teil II** des Buchs, »SAP-Anwendungen mit SAP Build Apps entwickeln«, beschäftigen wir uns mit der Entwicklung von Anwendungen mit SAP Build Apps. Da nur wenige Apps ohne Integration zu Fremdsystemen wie beispielsweise SAP S/4HANA oder SAP SuccessFactors auskommen, zeigen wir Ihnen in **Kapitel 5**, »Wie kommen die Daten in die App?«, wie die Daten in Ihre Apps gelangen. Dabei lernen Sie auch Technologien wie REST und OData kennen und finden heraus, worum es sich bei diesen Services handelt und wie Sie diese einsetzen. Die Daten allein sind in einer App jedoch nicht ausreichend. Sie müssen in Ihren Apps üblicherweise auch eine Busi-

ness-Logik abbilden. Darauf gehen wir in **Kapitel 6**, »Was soll Ihre App können?«, ein. Dort sehen Sie, wie Seiten erstellt und Inhalte darin dargestellt werden. Innerhalb der Apps werden Sie mit Variablen in verschiedenen Ausprägungen arbeiten und auf Events wie das Klicken auf einen Button reagieren.

In **Kapitel 7**, »Wie soll Ihre App aussehen?«, geht es um das Aussehen Ihrer Apps, also das Erstellen von Benutzeroberflächen. Dabei gehen wir auf die wichtigsten Komponenten ein und zeigen Ihnen, wie Sie damit Formulare für die Erfassung und Anzeige von Daten erstellen können. Neben Formularen ist auch die Ausgabe in Form von Listen eine oft verwendete Funktionalität, die wir Ihnen vorstellen möchten. Abschließend zeigen wir Ihnen, wie Sie eigene Layouts erstellen können.

Nachdem Sie nun erste Erfahrungen mit der Entwicklung von Apps gemacht haben, hoffen wir, dass wir Ihr Interesse und die Neugierde in Ihnen wecken konnten. Deshalb steigen wir in **Kapitel 8**, »Tiefer in die App-Entwicklung einsteigen«, tiefer in die App-Entwicklung ein. Sie lernen, wie Sie eigenen JavaScript-Code integrieren können, falls die zur Verfügung gestellten Funktionen Ihre Anforderungen nicht abdecken. Danach zeigen wir Ihnen, wie Sie Ihre Apps an Ihr Corporate Design anpassen können, indem Sie sogenannte Themes verwenden. Abschließend zeigen wir in diesem Kapitel, wie Sie sogenannte Custom Controls erstellen können. Damit haben Sie die Möglichkeit, eigene wiederverwendbare Komponenten zu erstellen.

Nachdem Sie Ihre App erstellt haben, möchten Sie diese hoffentlich auch der weiten Welt oder Ihren Kolleg\*innen zur Verfügung stellen. Darauf gehen wir in **Kapitel 9**, »App veröffentlichen«, ein. Dort zeigen wir Ihnen, welche Möglichkeiten es gibt, um eine App zu veröffentlichen. Besonders interessant ist die Bereitstellung der Apps in Form von Web-Apps auf der SAP BTP. Aber auch die Erstellung und Bereitstellung von mobilen Apps ist ein gefragtes Feature.

Nachdem Sie nun alle Aspekte der App-Entwicklung kennengelernt haben, zeigen wir Ihnen in **Teil III** des Buchs, »Entwicklungsbeispiele und Einsatzbereiche«, konkrete Entwicklungsbeispiele und Einsatzbereiche von Apps. Sie werden anhand von praktischen Beispielen erfahren, wie Sie bestimmte Anforderungen schnell und einfach mit SAP Build Apps umsetzen können.

In **Kapitel 10**, »Entwicklung einer App für Aufgabenlisten«, werden Sie eine App für die Verwaltung von Aufgabenlisten erstellen und testen. Dabei werden die Daten lokal auf dem jeweiligen Endgerät gespeichert und nicht mit anderen Benutzer\*innen oder Endgeräten geteilt.

Teil III

In **Kapitel 11**, »Entwicklung einer GPS-Tracking-App«, werden Sie eine GPS-Tracking-App entwickeln. Sie werden die Position des Endgeräts in Form von GPS-Koordinaten bestimmen. Das ist eine Standardfunktion, die Ihnen von SAP Build Apps bereitgestellt wird. Da die Koordinaten in Form von numerischen Werten üblicherweise für Menschen nur schwer zu interpretieren sind, erweitern wir dieses Beispiel, indem wir die Koordinaten in eine lesbare postalische Adresse umwandeln. Dazu bedienen wir uns einer REST-API.

In **Kapitel 12**, »Entwicklung einer Kundenverwaltungs-App«, werden Sie eine Kundenverwaltungs-App entwickeln. Das hört sich zunächst nicht besonders spannend an, da es sich um eine Darstellung von Daten in einer Liste beziehungsweise die Erfassung von Daten in einem Formular handelt. Was dieses Beispiel jedoch besonders macht, ist die Speicherung der Daten. Die erfolgt in Google Firebase. Dadurch haben Sie die Möglichkeit, die Daten mit anderen Benutzer\*innen und Endgeräten zu teilen und zentral in der Cloud zu speichern.

In **Kapitel 13**, »Entwicklung einer App zur Anzeige von Produktinformationen«, wird es besonders spannend. Sie werden eine App zur Anzeige von Produktinformationen erstellen. Dabei wird mit der Kamera des mobilen Endgeräts ein Barcode gescannt. Danach wird versucht, die Informationen zum jeweiligen Produkt über eine REST-API abzurufen und darzustellen.

Abschließend werden Sie in **Kapitel 14**, »Entwicklung eines Formulars zur Datenausgabe und -pflege«, ein Formular zur Datenausgabe und -pflege entwickeln. Die Besonderheit an diesem Beispiel ist die Verwendung einer OData-API. Damit werden Sie über den De-facto-Standard von SAP einen Service integrieren, in dem die Daten zentral gespeichert werden.

## Informationskästen

Um Sie auf wichtige Informationen hinzuweisen und Ihnen so die Arbeit mit diesem Buch zu erleichtern, verwenden wir im Text die folgenden Symbole:



In Kästen, die mit diesem Symbol gekennzeichnet sind, finden Sie Informationen zu weiterführenden Themen oder wichtigen Inhalten, die Sie sich merken sollten.



Mit diesem Symbol sind Tipps und Hinweise aus der Berufspraxis markiert, die praktische Empfehlungen geben, die Ihnen die Arbeit erleichtern können.

Dieses Symbol weist Sie auf Besonderheiten hin, die Sie beachten sollten. Es warnt Sie außerdem vor häufig gemachten Fehlern oder Problemen, die auftreten können.



Wir danken Ihnen für das Interesse an unserem Buch und wünschen Ihnen viel Vergnügen bei der Lektüre.