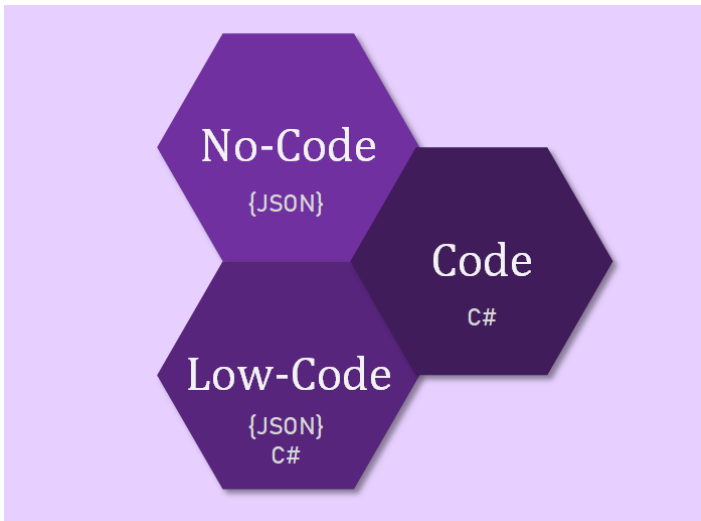


No-Code/Low-Code Entwicklungsplattform für Lohndienstleister

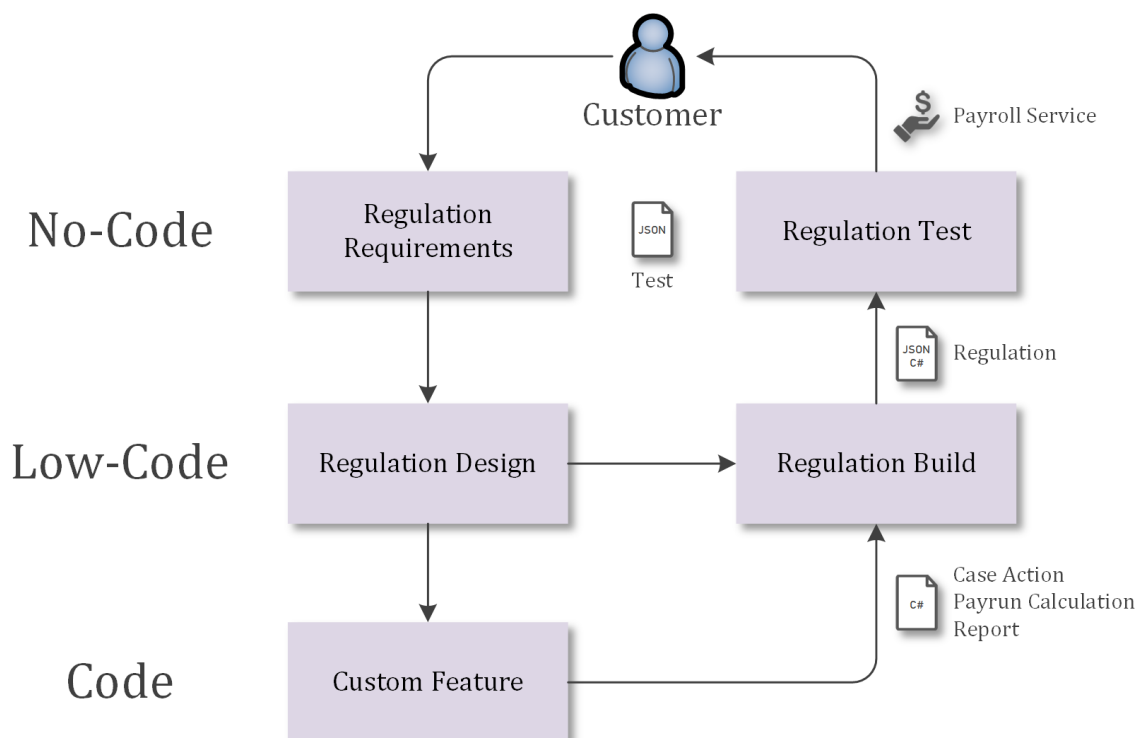


Kundenorientierte Entwicklung von Lohnanwendungen mit und ohne Programmierung

No-Code und Low-Code Plattformen versprechen eine markante Vereinfachung bei der Entwicklung von Business-Software und verzeichnen einen starken [Wachstumstrend](#). Sie bieten mit Hilfe von UI-Tools oder Konfigurationen die Möglichkeit, ohne Programmierkenntnisse rasch individuelle Kundenlösungen zu entwickeln.

Auch wenn sich dieses Versprechen gut anhört, stoßen reine No-Code/Low-Code Lösungen, in der Regel, bei der Gestaltung von komplexen Lösungen schnell an ihre Grenzen. Sehr spannend wird es jedoch, wenn zudem eine Programmierschnittstelle zur Verfügung steht.

Mit der *Payroll Engine* können Regulierungen (für ein Land, eine Branche, eine Pensionskasse, firmenspezifische Zusätze, usw.) mittels JSON-Konfiguration und einer Scripting-Programmierschnittstelle komplett autonom entwickelt werden.



Der testgetriebene Ansatz gewährleistet eine dokumentierte Abgrenzung zwischen den Beteiligten und bietet die Sicherheit, dass die Lösung den Kundenanforderungen entspricht. Die Testdaten dienen auch als Grundlage bei zukünftigen Erweiterungen.

Im *Payroll Engine* Entwicklungsprozess sind die verschiedenen Implementierungsansätze auf die jeweilig qualifizierten Entwickler aufgeteilt:

	Anwender	Aufgaben	Technische Skills	Ergebnis
No-Code	Regulation Owner	Anforderungen erfassen und Regulation testen	JSON	Case-Test Payrun-Test Report-Test
Low-Code	Regulation Builder	Regulation gestalten und entwickeln	JSON, C# Beginner	Regulation
Code	Regulation Developer	Features entwickeln	JSON, C# Intermediate	Case-Action Payrun Formel Report

Da der *Regulation Builder* nur minimale C# Kenntnisse (Methodenaufruf) benötigt, ist ein technisch versierter *Regulation Owner* in der Lage, die Anwendung selbstständig zu entwickeln. Der Einsatz des *Regulation Developer* wird dann erforderlich, wenn die Funktion nicht im *Payroll Engine* Regulation Ökosystem vorhanden ist.

Die Kombination von No-Code/Low-Code mit individueller Programmierung ermöglicht es Payroll Providern auch die komplexesten Kundenanforderungen abzudecken. Dies mit einer nie dagewesenen Entwicklungsgeschwindigkeit.

Die Begeisterung, die bei Kunden durch diese Individualisierung ausgelöst wird, motiviert Payroll Spezialisten schnell für das Erlernen von JSON-Konfiguration und Basiswissen in der Codierung. Das Know-how und die Erfahrung im Bereich Lohn ist nach wie vor eine zwingende Voraussetzung für den *Regulation Builder*. Das Berufsbild der heutigen Payroll Spezialisten wird durch die *Payroll Engine* somit nochmals stark aufgewertet.