

Payroll Engine - Design einer skalierbaren Lohnsoftware



Getrieben durch Bürokratisierung, Digitalisierung und stetige Technologiewechsel ist die Gestaltung einer modernen Software eine komplexe Herausforderung. Basierend auf Erfahrungswerten der Praxis ist bei der Lohnsoftware insbesondere die Skalierbarkeit der Geschäftsfälle entscheidend.

Um die Komplexität zu reduzieren, werden Softwaresysteme üblicherweise in verschiedene Bereiche (Schichten oder Dienste) unterteilt. Diese sind in der Regel das Frontend/Client mit der Benutzeroberfläche (UI) sowie das Backend/Server, welches aus der Applikation und der Datenpersistenz (Datenbank) besteht. Aus Sicht der Wertschöpfung ist die Applikation der Softwarekern, weil sie die Geschäftsprozesse implementiert. Die Benutzeroberfläche und Persistenz sollten optimalerweise aus dem Businessmodell generiert werden.

Wird die Applikation monolithisch gebaut, muss jede Anpassung von hochqualifizierten Softwareingenieuren realisiert werden, welche nicht immer die relevanten Businessanforderungen kennen.

Beim Design der *Payroll Engine* orientierten wir uns an das Konzept von *Domain-Specific Application Frameworks*¹⁾ (geschäftorientierte Applikations-Rahmenwerke), welches die Trennung der Applikation in ein Framework (Rahmenwerk) und der Domain/Business Applikation vorsieht. Das Framework beinhaltet die Basisfunktionen und wird von Softwareingenieuren entwickelt, welches nur bei grundlegenden Änderungen tangiert ist. Die Business Applikation wird von Fachspezialisten entwickelt, welche dafür fachgerechte Hilfsmittel erhalten. Vom Fachspezialisten wird ein gewisses technisches Verständnis vorausgesetzt.

Diese Trennung reduziert den administrativen, finanziellen und kommunikativen Aufwand zwischen dem Business- und Framework-Entwickler. Zentraler Designaspekt ist die korrekte Wissensaufteilung, was die agile Entwicklung von Geschäftsanwendungen ermöglicht und den Knowhow-Transfer zwischen den Parteien, zum Beispiel bei einem Personalwechsel, auf das Notwendige reduziert.

In der *Payroll Engine* repräsentiert die REST API das Framework und die Regulierung ein Bestandteil der Business-Applikation. Die Regulierung beinhaltet die Geschäftsfälle, die Lohnberechnung sowie die Auswertungen/Reports.

Eine weitere Herausforderung in der Gestaltung einer Lohnsoftware ist der Umstand, dass die Businessfunktionalität aus verschiedenen Quellen stammt wie z.B. Swissdec, LGAV, Versicherung usw.. Die *Payroll Engine* integriert jede Businessquelle als eigene Regulierung und führt diese in einem Schichtenmodell zu einer Business-Applikation zusammen. Analog der Bildebene in einem Bildbearbeitungsprogramm, ergeben alle Regulierungsschichten das Gesamtbild der Business-Applikation. Anpassungen für den Lohnmandanten werden in der obersten Regulierungsschicht (Bildebene) integriert.

Mit dem *Payroll Engine* Ökosystem werden Regulierungen zwischen den Lohndienstleistern ausgetauscht. Damit wird kontinuierlich Lohnwissen aufgebaut und geteilt, was für die Skalierbarkeit der Lohnsoftware langfristig ein enormes Potential bedeutet.

1) Mohamed E. Fayad & Ralph E. Johnson – Domain-Specific Application Frameworks, ISBN: [978-0-471-33280-0](https://www.wiley.com/en-us/Domain-Specific+Application+Frameworks-p9780471332800), Wiley