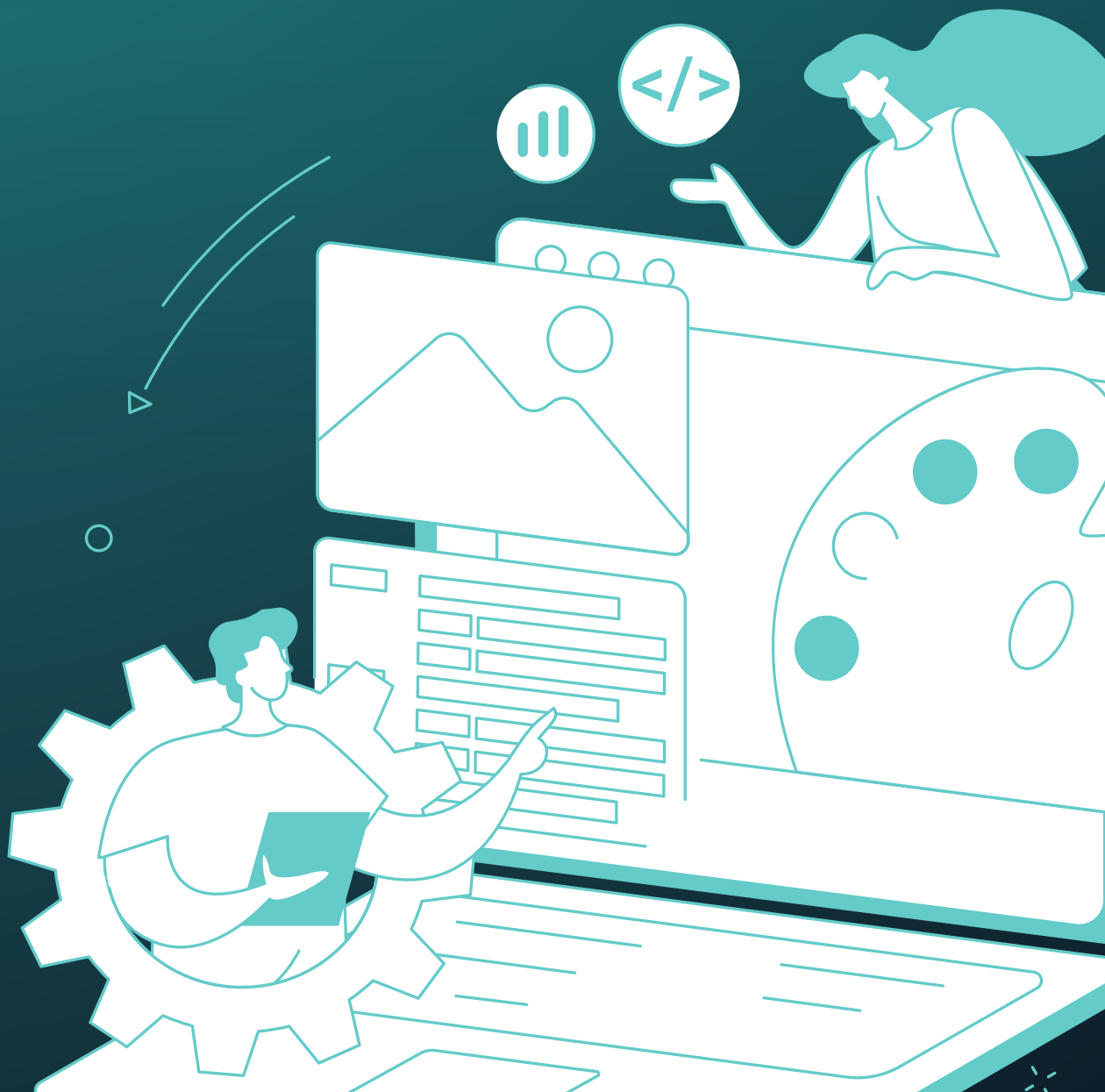


WHITEPAPER

# Agile Entwicklung & Low-/No-Code: Wie Unternehmen maßgeschneiderte Software schneller und effizienter realisieren





# Agile Softwareentwicklung mit Low- / No-Code

Was nach einem Schlagwort klingt, ist in der Realität für viele Unternehmen noch immer eine große Hürde. Starre Softwarelandschaften, lange Entwicklungszyklen und fehlende Ressourcen bremsen Innovationskraft und Veränderungsgeschwindigkeit. Doch genau diese Trägheit entscheidet heute darüber, ob ein Unternehmen als Vorreiter agiert oder nur hinterherläuft. Wer sich nicht permanent anpassen kann, verliert schnell den Anschluss. Low-/No-Code-Technologien in Verbindung mit agiler Entwicklung bieten hier einen praxisnahen Hebel: Sie machen Flexibilität tatsächlich umsetzbar – nicht als Vision, sondern als gelebte Realität im Tagesgeschäft.

Digitale Transformation entscheidet heute über Markterfolg. Low-/No-Code-Technologien und agile Methoden sind dabei ein starkes Duo: Sie ermöglichen individuelle Anwendungen in Rekordzeit – effizient, flexibel und skalierbar. Durch die kluge Verbindung beider Ansätze lassen sich Innovationszyklen verkürzen, Entwicklungskosten reduzieren und die Wettbewerbsfähigkeit langfristig sichern.



# Agile Softwareentwicklung mit Low- / No-Code

## 1 Warum Flexibilität in der Softwareentwicklung heute entscheidend ist.

Die Anforderungen an Unternehmenssoftware wachsen stetig: Prozesse werden individueller, Anpassungen an Marktveränderungen müssen immer schneller erfolgen, und die Time-to-Market für neue Lösungen verkürzt sich drastisch. Klassische Softwareentwicklungsprozesse stoßen dabei häufig an ihre Grenzen – sowohl zeitlich als auch finanziell. Monatelange Entwicklungsphasen und starre Projektpläne passen nicht mehr zu den dynamischen Anforderungen moderner Märkte.

In der heutigen Geschäftswelt ist Anpassungsfähigkeit kein Nice-to-have mehr, sondern überlebenswichtig. Digitalisierung, disruptive Technologien und sich ständig wandelnde Kundenbedürfnisse verlangen von Unternehmen, flexibel und schnell auf Veränderungen zu reagieren. Flexibilität und Agilität werden damit zu zentralen Erfolgsfaktoren für Organisationen, die wettbewerbsfähig bleiben wollen.

Besonders im Field Service Management (FSM) zeigt sich diese Herausforderung deutlich: Standardisierte Softwarelösungen stoßen schnell an ihre Grenzen, wenn es darum geht, spezifische Anforderungen der Service- und Außendienstprozesse abzubilden. Unternehmen sehen sich gezwungen, entweder auf teure Individualentwicklungen zurückzugreifen oder Kompromisse bei der Funktionalität einzugehen. Trotz bekannter Nachteile wie ineffizienten Workarounds, geringer Nutzerakzeptanz und hohem Anpassungsaufwand.



**Flexibilität und Agilität sind keine Kür mehr - sie sind die Grundvoraussetzung, um in einem dynamischen Marktumfeld wettbewerbsfähig zu bleiben.**

Doch es gibt einen Weg, diesen Spagat zu meistern: Die Kombination aus agilen Methoden und Low-/No-Code-Technologien. Dieser Ansatz bildet die Grundlage für eine neue Art der Softwareentwicklung – individuell, schnell, flexibel und ressourcenschonend. Durch die intelligente Verbindung dieser beiden Prinzipien lassen sich passgenaue FSM-Lösungen entwickeln, die sich an die Prozesse anpassen (und nicht umgekehrt). Dieses Whitepaper zeigt, wie agile Softwareentwicklung in Kombination mit Low-/No-Code-Technologien zum strategischen Erfolgsfaktor wird insbesondere dort, wo Standardlösungen nicht mehr ausreichen.





# 2 Was ist agile Softwareentwicklung?

Agilität in der Softwareentwicklung heißt nicht einfach „flexibel“ oder „schnell mal was anpassen“. Es geht um eine grundsätzliche Denkweise:

- Kleine, inkrementelle Entwicklungsschritte statt starrer Großprojekte
- Kontinuierliches Feedback von Nutzern und Stakeholdern
- Cross-funktionale Teams, die Business und IT zusammenbringen
- Priorisierung nach Business-Value, nicht nach Lastenheft

Gerade bei FSM-Lösungen bedeutet das: Die Software entwickelt sich gemeinsam mit den realen Prozessen im Unternehmen – nicht im Elfenbeinturm der IT-Abteilung.

## Die wichtigsten Merkmale agiler Softwareentwicklung:

### Iterative Zyklen (Sprints)

Die Entwicklung erfolgt in kurzen Zeitabschnitten, sogenannten Sprints, die in der Regel zwei bis vier Wochen dauern. Am Ende jedes Sprints steht ein lauffähiges Software-Inkrement, das bereits produktiv eingesetzt oder getestet werden kann.

**„Agilität heißt, nah an den Menschen zu entwickeln, die täglich mit der Software arbeiten.“**

### Kundenzentrierung & Feedback

Regelmäßige Abstimmungen mit den Fachbereichen stellen sicher, dass die Software wirklich den Bedürfnissen der Nutzer entspricht. Anforderungen können in Echtzeit angepasst werden. Das minimiert Fehlentwicklungen und maximiert den Nutzen.

### Flexibilität & Anpassungsfähigkeit

Agile Teams sind darauf ausgelegt, auch während des laufenden Projekts auf neue Anforderungen, Marktveränderungen oder technologische Entwicklungen schnell zu reagieren. Planungszyklen bleiben kurz und beweglich.

### Teamarbeit & Kommunikation

Die agile Methodik fördert eine enge, interdisziplinäre Zusammenarbeit zwischen Entwicklern, Product Ownern und Stakeholdern. Offene Kommunikation und transparente Arbeitsweisen sind zentrale Bausteine für den Projekterfolg.

## Kernprinzipien der agilen Entwicklung



Individuen & Interaktionen stehen über Prozessen & Tools



Funktionierende Software ist wichtiger als umfassende Dokumentation



Reagieren auf Veränderung ist wichtiger als das Befolgen eines starren Plans



Zusammenarbeit mit dem Kunden ist wichtiger als Vertragsverhandlungen



# 3 Was ist Low-/No-Code?

Low-Code-Plattformen ermöglichen die Entwicklung von Anwendungen mit minimalem Programmieraufwand. No-Code-Plattformen gehen noch einen Schritt weiter und erlauben es, Prozesse, Workflows und Benutzeroberflächen komplett ohne Programmierkenntnisse zu erstellen. Durch visuelle Editoren und vorgefertigte Bausteine wird Softwareentwicklung zugänglicher, schneller und fehlerresistenter.

## Schlüsselfunktionen von Low-/No-Code-Plattformen

- 
**Visuelle Modellierung:** Workflows, Datenbanken und Benutzeroberflächen werden per Drag & Drop konfiguriert.
- 
**Wiederverwendbare Komponenten:** Vorgefertigte Bausteine für gängige Funktionen wie Formulare, Datenintegrationen oder Automatisierungen.
- 
**Direkt-Vorschau & Deployment:** Anwendungen können direkt getestet und ohne lange Entwicklungszyklen veröffentlicht werden.
- 
**Integrationen via API & Schnittstellen:** Moderne Plattformen bieten offene Schnittstellen, um sich nahtlos in bestehende Systemlandschaften zu integrieren.
- 
**Versionskontrolle & Deployment-Automation:** Auch komplexe Entwicklungsprozesse können versioniert und automatisiert ausgerollt werden.



## Vorteile von Low-/No-Code Technologien



**Beschleunigte Entwicklungszeiten:** Dank wiederverwendbarer Komponenten und visueller Modellierung entstehen erste lauffähige Versionen innerhalb weniger Tage.



**Einfache Anpassung:** Prozessänderungen, neue Formularfelder oder Workflows können direkt durch den Fachbereich oder in enger Zusammenarbeit mit IT-Partnern umgesetzt werden.



**Kosteneffizienz:** Da viele Entwicklungsschritte standardisiert und visuell abbildbar sind, sinkt der Bedarf an teurer individueller Programmierung.



**Höhere Innovationsrate:** Iteratives Vorgehen ermöglicht es, neue Ideen schnell zu testen und direkt im Tagesgeschäft zu evaluieren – mit minimalem Risiko.



# 4 Warum sich der Wechsel jetzt lohnt - und wie ihr ihn richtig angeht

**Standardsoftware bietet oft eine breite Palette an Funktionen. Spezifische Anforderungen eines Unternehmens können aber meist nicht vollständig oder nur mit Kompromissen abgedeckt werden. Was zwangsläufig zu Inflexibilität, Komplexität und ineffizienten Prozessen führt.**

Bei der agilen Entwicklung mittels Low-Code werden zuerst die für den Kunden wichtigsten Funktionen realisiert, die den größten Geschäftswert bieten. Dies ermöglicht es, wesentliche Teile der neuen FSM-Software frühzeitig produktiv zu nutzen. Durch das iterative Vorgehen können regelmäßig Anpassungen und Verbesserungen vorgenommen werden, was zu einer kontinuierlichen Optimierung führt.

**„FSM-Software muss so flexibel sein wie eure täglichen Herausforderungen.“**

Ein weiterer entscheidender Aspekt der agilen Entwicklung ist die enge und fortlaufende Kommunikation mit dem Kunden. Diese ständige Interaktion sorgt dafür, dass die entwickelten Lösungen genau den Bedürfnissen der Anwender entsprechen, was nicht nur zu einer erhöhten Akzeptanz der Software sondern auch zu mehr Effizienz in den Abläufen führt.

Viele Unternehmen zögern, ihre bestehende FSM-Lösung abzulösen. Der Gedanke „läuft doch irgendwie“ hält sich hartnäckig. Doch was selten bedacht wird: Die versteckten Kosten der Inflexibilität steigen mit jedem Tag.

- Veraltete Prozesse kosten nicht nur Zeit und Nerven – sie verhindern Innovation.

- Fehlende Transparenz erschwert Steuerung und Planung.

- Jede Anpassung wird zum teuren IT-Projekt.

**Dem gegenüber stehen klare, messbare Vorteile moderner FSM-Ansätze:**

- 50–70 % schnellere Entwicklungszyklen durch Low-Code

- Deutliche Kostenreduktion bei Anpassungen und Erweiterungen

- Mehr Unabhängigkeit von Marktveränderungen

- Höhere Nutzerakzeptanz durch aktive Einbindung der Fachbereiche



## Vorteile maßgeschneiderter (Field Service-) Software

### **Benutzerfreundlichkeit**

Intuitive Oberflächen und Workflows, die den tatsächlichen Arbeitsabläufen entsprechen.

### **Skalierbarkeit**

Lösungen, die problemlos mit den Anforderungen des Unternehmens wachsen können.

### **Perfekte Passform**

Software, die genau auf die spezifischen Bedürfnisse und Prozesse des Unternehmens zugeschnitten ist.

### **Höherer ROI**

Passgenau Prozesse ermöglichen eine deutliche Produktivitätssteigerung und Effizienz.



# 5 Der Weg zur agilen FSM-Lösung: Worauf es jetzt ankommt

Einfach eine neue Plattform einführen? Das reicht nicht. Der Wechsel zu einer agilen, Low-/No-Code-basierten FSM-Lösung wird dann zum Erfolg, wenn ihr ihn als ganzheitliche Veränderung versteht. Diese Faktoren sind entscheidend:

**Top-Management-Commitment:** Agilität beginnt im Kopf der Entscheider.

**Projekt mit klaren Business-Zielen:** Erfolge sichtbar machen.

**Enge Zusammenarbeit** von IT, Fachbereich und Lösungsanbieter: Low-Code lebt vom Dialog.

**Partner mit FSM-Expertise:** Technologie allein löst keine Prozesse.

**IT-Security und Integration:** Datenschutz, Rollen- und Berechtigungskonzepte, System-Schnittstellen – von Anfang an mitdenken.

Und das Wichtigste:

**Die beste Zeit, Prozesse neu zu denken, ist jetzt.**

Ihr müsst nicht sofort alles umstellen. Aber ihr müsst anfangen und zwar in kleinen, konsequenten Schritten. Denn jeder Fortschritt Richtung Flexibilität zahlt sich direkt aus.

**„Agilität ist kein Tool, sondern eine Haltung - Low-/No-Code macht sie erlebbar.“**

## Checkliste: Erfolgsfaktoren für agile FSM-Projekte mit Low-Code

### Business-Ziele definieren

Was soll konkret verbessert werden (Reaktionszeiten, Transparenz, Prozesskosten)?

### Iteratives Vorgehen planen

Kurze Entwicklungszyklen mit direktem Feedback der Anwender.

### Cross-funktionales Team aufstellen

Fachbereich & IT gemeinsam an einen Tisch holen.

### Change-Management einplanen

Mitarbeitende und Anwender frühzeitig einbinden und befähigen.

### Prioritäten festlegen

Kleinen, überschaubaren Use Case wählen, der schnell Erfolge zeigt.

### FSM-Expertise sicherstellen

Partner wählen, die FSM-Prozesse wirklich verstehen.

### Plattform auswählen

Usability, Integrationsfähigkeit & Sicherheitsstandards prüfen.

### Technische Integration vorbereiten

Schnittstellen zu bestehenden Systemen (ERP, DMS etc.) definieren.



# 6 Euer nächster Schritt: Field Service Management mit Perspektive

**Agile Softwareentwicklung und Low-Code-Technologie eröffnen neue Wege, individuelle FSM-Lösungen schnell und flexibel umzusetzen, passgenau für eure Prozesse, eure Anforderungen, eure Teams.**

Unternehmen, die auf diese Methoden setzen, profitieren von:

- höchster Flexibilität im Tagesgeschäft,
- kurzen Entwicklungs- und Implementierungszeiten,
- und Lösungen, die mit den Herausforderungen wachsen.

**In einer dynamischen Geschäftswelt ist genau das der Schlüssel, um langfristig wettbewerbsfähig zu bleiben.**

Wir bei **mobile function** begleiten euch auf diesem Weg – mit Erfahrung, technologischem Know-how und einem echten Verständnis für eure Praxis.

Lass uns darüber sprechen, wie eure Field Service Prozesse von agiler Softwareentwicklung profitieren können.



**Unsere Expertin:  
Myriam Siewert**



+49 7721 69700-350



myriam.siewert@mobile-function.com



www.mobile-function.com

**mobile function** bietet Unternehmen für deren individuelle Geschäftsprozesse die eigenes entwickelte, **leistungsstarke Low-Code Plattform ENGINE4**, die eine besonders agile und flexible Entwicklung ermöglicht. Spezifische Kundenanforderungen können jederzeit berücksichtigt werden und maßgeschneiderte Lösungen werden entwickelt.

Die Lösung **ENGINE4.SERVICE** ist dabei speziell auf die Bedürfnisse des **Field Service-Managements** zugeschnitten. Das Zusammenspiel aus Low-Code-Technologie, agiler Methodik und bestehenden Serviceprozessen sorgt dafür, dass die Field Service Management-Lösungen effizient und perfekt an die Geschäftsprozesse der Kunden angepasst sind.

# ENGINE4



**mobile function GmbH**  
Niederwiesenstraße 28  
D-78050 Villingen-Schwenningen

+49 (0) 77 21 / 69 700-300  
info@mobile-function.com

**[www.mobile-function.com](http://www.mobile-function.com)**

ENGINE