

1	Potenziale der Low-Code-Programmierung für Industriebetriebe	1
	Sven Hinrichsen und Benjamin Adrian	
1.1	Paradigmenwechsel der Produktionssystemgestaltung	2
1.2	Low-Code-Development als Ansatz zur Beherrschung von Komplexität	6
1.3	Einordnung und Bedeutung der Low-Code-Programmierung	10
1.4	Merkmale, Anwendungsgebiete und Potenziale der Low-Code-Programmierung	11
	Literatur	14
2	Merkmale und Entwicklungslinien der Low-Code-Programmierung	17
	Stefan Sauer, Nils Weidmann und Jonas Kirchhoff	
2.1	Überblick und Einordnung	18
2.2	Programmiersprachen der 4. Generation	18
2.3	Generative Programmierung	20
2.4	Modellgetriebene Softwareentwicklung	21
2.5	Domain-Driven Design & domänen spezifische Sprachen	24
2.6	Rapid Application Development	25
2.7	End-User Development	26
2.8	No Code vs. Low Code vs. High Code	28
	Literatur	29
3	Auswahl und Einführung einer Low-Code-Plattform	31
	Alexander Nikolenko, Kai Leon Becker, Uwe Wohlhage, Benjamin Adrian und Sven Hinrichsen	
3.1	Einleitung	32
3.2	Vorgehensmodell zur Auswahl und Einführung einer Low-Code-Plattform	33

3.3	Phase 1: Analyse der Ausgangssituation und Klären des Projektrahmens	34
3.4	Phase 2: Beschreiben von Anwendungsfällen und Ermitteln der Anforderungen	36
3.5	Phase 3: Vorauswahl der Low-Code-Plattform	39
3.6	Phase 4: Finale Auswahl der Low-Code-Plattform	43
3.7	Phase 5: Einführen der Low-Code-Plattform	45
3.8	Phase 6: Einsetzen der Low-Code-Plattform und Evaluieren des Projektes	48
	Literatur	49
4	Architektur von Low-Code-Plattformen	53
	Klaus Schröder	
4.1	Kontextsicht auf Low-Code-Plattformen	54
4.2	Die Kontextsicht als Methode der Software-Architektur	54
4.3	Kontextsicht auf Low-Code-Plattformen	55
4.4	Programmiermodell von Low-Code-Plattformen	56
4.5	Exkurs: Evil Wizards	59
4.6	Verfeinerung der Bausteinsicht	60
	Literatur	61
5	Strukturierte Entwicklung von Softwareanwendungen mit Low-Code	63
	Jonas Kirchhoff, Nils Weidmann, Stefan Sauer und Gregor Engels	
5.1	Effiziente und Effektive Softwareentwicklung	64
5.2	Einführung in Softwareentwicklungsmethoden	66
5.3	Situative Faktoren der Low-Code-Entwicklung	66
5.4	Dokumentationsformat	68
5.5	Phase 1: Anforderungserhebung	70
5.6	Phase 2: Modellierung der Domäne	73
5.7	Phase 3: Integration externer Systeme	75
5.8	Phase 4: Entwicklung der Geschäftslogik	78
5.9	Phase 5: Umsetzung der Benutzeroberfläche	79
5.10	Phase 6: Qualitätssicherung	82
5.11	Fazit und Ausblick	84
	Literatur	86
6	Erfolgreiche Auswahl und Einführung einer Low-Code-Plattform bei ISRINGHAUSEN – eine Fallstudie	87
	Alexander Nikolenko, Benjamin Adrian, Sven Hinrichsen, Michael Rohrig und Nils Weidmann	
6.1	Phase 1: Analyse der Ausgangssituation und Klären des Projektrahmens	88

6.2	Phase 2: Beschreiben von Anwendungsfällen und Ermitteln von Anforderungen	89
6.3	Phase 3: Vorauswahl der Low-Code-Plattform	91
6.4	Phase 4: Finale Auswahl der Low-Code-Plattform	92
6.5	Phase 5: Einführung der Low-Code-Plattform	93
6.6	Phase 6: Einsetzen der Low-Code-Plattform und Evaluieren des Projektes	96
6.7	Fazit und Ausblick	96
	Literatur	97
7	Low-Code-Development als integraler Bestandteil der Digitalisierungsstrategie von DENIOS	99
	Udo Roth und Jan Regtmeier	
7.1	Zum Unternehmen Denios SE	100
7.2	Einführung und Motivation	101
7.3	Fallstudie „Buchungssystem für Ladesäulen“	102
7.4	Fazit und Ausblick	107
	Literatur	110
8	Software für Speicherprogrammierbare Steuerungen entwickeln: Low-Code oder Modellgetrieben?	111
	Nils Weidmann, Johannes Heil und Micha Wegener	
8.1	Einleitung	112
8.2	Use Case und Zielsetzung	113
8.3	Aufbau und Funktionsweise der Tool-Chain	116
8.4	Metamodellierung	118
8.5	Evaluation	120
8.6	Fazit und Ausblick	122
	Literatur	123