

Stichwortverzeichnis

A

Abläufe
 unnötige 52
 verbessern 51
Abrufbereitschaft 242
Abstimmung von Anreizen 316
Adaptionskurve 69
Agile 30
 Manifest 30, 96
 und DevOps 95
 Ursprünge 96
Agiles Projektmanagement 95
Agile Story Points 178
Airbnb 101
Aktivitäten,
 überflüssige 51
Alarmsignale 183
Alpha-Version 106
Amazon Elastic Container Service (ECS) 290
Amazon Web Services (AWS) 297, 300, 302
Analysen einplanen 112
Änderungen
 inkrementelle 32
 schrittweise
 durchführen 81
Anpassungsfähigkeit 62
Ansible 240
Anti-Patterns 133
Antwortcodes 283
Apache License 268
Apollo 11 88
Application Packaging 157
Application Performance Management (APM) 79
Application Programming Interface (APIs) 281
 Endpunkte 283
 Paginierung 285
 Parameter
 hinzufügen 283
 Statuscodes 283
 Versionierung 285

Arbeitspuffer 60
Architektur 110
Architekturteams 112
Aufgaben
 nach links schieben 92
 weitergeben 111
Ausfallsicherheit 247
 Rechenzentren 229
Auslieferung
 beschleunigen 314
Ausnahmen 128
Austausch 35
Automation Engineer 204
Automatisierung 33,
 155, 236, 314
 Cloud 295
 realistische 237
Automatisierungsteam 200
Autonomie 219–221
Azure Kubernetes Service (AKS) 289

B

Bedrohungen 118
Belohnungen 220
Benutzererfahrung 104, 119
Benutzerfreundlichkeit 119
Bereitschaftsdienst 198,
 242, 296, 317
 Anforderungen 243
 Burnout vorbeugen 242
Bereitschaftsschichten 244
Bereitstellung 90
 kontinuierliche 32, 313
Berufsbezeichnung 203
Besprechungskosten 77
Beständiger Wandel 309
Beständigkeit 246
Best Practices 137
Beta-Release 106
Beurteilungen 48
Bleeding Edge 134
Blockbuster 62
Blue-Green-Deployments 160
Boilerplate 139

Brownfield 129
Budget 98–99
 realistisches 248
Bugs 141
Burnout 242

C

CALMS-Modell 33
Canary Deployment 162
Cargo Culting 133
CFR 181
Change Failure Rate 181
Checklisten 246
Chef (Anwendung) 241
CI/CD 32, 152, 313
 implementieren 153
 Vorteile 153
CircleCI 239
Clan 44, 201
Clean Code 132
Closed-Source-Software 265
Cloud 295, 310
 Anbieter wählen 301
 Arten 298
 Auslieferung 297
 Deployments 297
 Fehlermöglichkeiten
 verringern 298
 Kosteneffizienz 297
 Skalierbarkeit 298
Cloud as a Service 299
Cloud-basierte
 Anwendungen 117
Cloud Engineer 204
Cloud-Hosting 277
Cloud Transformation Team 201
Code
 debuggen 129
 freigeben 151
 immutabler 130
 lesbarer 131
 protokollieren 130
 Reviews 139
 sauberer 135

- überprüfen 139
- unveränderbarer 130
- wartungsfreundlicher 128
- Codeanalyse 114
- Codebasis
 - einheitliche 114
 - veraltete 112
- Code-Coverage 148
- Code-Review im
 - Vorstellungsgespräch 209
- Communications Chief 249
- Compilersprachen 270
- Container 277, 285–286
 - absichern 292
 - konfigurieren 290
 - Lebenszyklus 291
 - Nachteile 291
 - Schwachstellen 293
 - überwachen 291
- Continuous Delivery 152, 154
- Continuous Deployment 154
- Continuous
 - Integration 152–153
- Contraction 77
- Coppypasta 134
- Cost per Incident (CPI) 254
- Currying 272
- Customer Events 166

D

- Daily-Stand-up-Meeting 216
- Daten
 - analysieren 195
 - formatieren 285
- Datenerfassung
 - automatisieren 80
- Datensammlung 166
- Deadline 173
- Debugging 128–129
- Delivery 87
- Deltas 157
- Deployment 151
 - automatisiertes 297
 - Blue-Green 161
 - Canary 162
 - Dauer 78
 - demokratisieren 159
 - Frequenz 78, 84, 180
 - managen 240
 - nach links verschieben 93
 - Rolling 163

- Deployment-Pipeline automa-
tisieren 314
- Deploys
 - fehlgeschlagene 79
 - Größe 78
- Design 90
- Design by Committee 133
- Detraktoren 194
- DEV 144, 151
- Development Lead Time 83
- DevOps
 - als kultureller Wandel 39
 - als Philosophie 29
 - Definition 29
 - Menschen im Mit-
telpunkt 30
 - quantifizieren 77
 - und Agile 30, 95
 - Werte 40

- DevOps Engineer 203
- DevOps-Team 199
- Dienstaussfall 235
- Dienstverlässlichkeit 168
- Diversität 45, 126
- Docker Swarm 289
- Dogfooding 163
 - Feedback 191
 - Vor- und Nachteile 192
- Dokumentation 121, 139
- Dominoeffekt 227
- Dringlichkeit vs.
Wichtigkeit 249
- Driver 148

E

- Early Adopters 192
- Echtzeit-Code-Interview 209
- Einzelleistungen 49
- Eisenhower'sche Entschei-
dungsmatrix 172
- E-Mail 183
- Empirische Messung 253
- Endpunkte,
pluralisierte 283
- Engpass 58
 - beseitigen 180
- Entlohnung 220
- Entscheidungsmatrix 172
- Entwickler
 - motivieren 218, 220
 - vs. Programmierer 136

- Entwicklerfeedback 111
- Entwicklungsdurchlaufzeit 83
- Entwicklungsleistung ver-
bessern 180
- Entwicklungspipeline 87
- Entwicklungsteams ska-
lieren 213
- Entwicklungszeit
einschätzen 111
- Ergonomie 236
 - kognitive 236
 - organisatorische 236
- Erprobung 90
- Ersthelfer 249
- Ethnische Herkunft 207
- Expansion 77
- Experimente anregen 232
- Extreme Programming 96

F

- Facebook 101, 171, 217
- Fail Deadly 228
- Fail Fast 171, 227, 320
- Fallstricke 315
- Fallstudien
 - interne 80
 - qualitative 81
 - quantitative 83
- Fallstudien-Meeting 191
 - Vor- und Nachteile 191
- Feature-Flags 165
- Features dokumentieren 138
- Feature-Toggle 165
- Feedback 42
 - analysieren 188
 - Dogfooding 191
 - empfangen 187
 - kommunizieren 188
 - kontinuierliches 193
 - Rhythmus 194
 - sammeln 190
 - Zyklus 193
- Feedback-Schleife 187
- Fehler
 - abmildern 158
 - akzeptieren 230
 - wiederkehrende 78
- Fehlerausbreitung
 - einschränken 228
- Fehlerauslöser 228
- Fehlerbehandlung 128

Fehlerkosten 53
 Fehlermeldung 128, 285
 Fehlerquelle 228
 Fehlerquote 79
 Fehlerursachen-Analyse 255
 Flatarchie 44
 Flexibilität 120
 Flow-Gefühl 232
 Flow-Zustand 220
 Fortschritt messen 75, 253
 Freizeitaktivitäten 50
 Führungskräfte,
 Unterstützung durch 65
 Führungsstil 38
 Full Stack 122
 Funktionale
 Programmierung 132, 272
 Funktionale Teams 198
 Funktionen
 höherer Ordnung 272
 unverzichtbare 103
 unwichtige 103
 Funktionsübergreifendes
 Team 200

G

Ganzheitliche Sichtweise 41
 Garbage Collection 273
 Geheimnisse absichern 293
 Geschäftsziele,
 gemeinsame 100
 Geschwindigkeit 177
 Gesprächsprotokolle 262
 GitHub 137
 GitLab 137
 GNU General Public Licenses
 (GPL) 268
 Go CD 239
 God Objects 133
 Goldilocks-Aufgaben 233
 Google 232
 Google Cloud Platform (GCP)
 297, 300, 303
 GraphQL 267
 Grenzfälle 128
 Großunternehmen 216
 Grundursache 236

H

Hackathon 50, 232
 Happy-Path-Tests 138

Herausforderungen,
 soziotechnische, 213
 Hierarchie 44
 Hochstaplersyndrom 127, 208
 Holokratie 44
 HTTP-Statuscodes 283
 Humanfaktor 236
 Hybrid Cloud 299
 Hype Cycle 71

I

Ideenwettbewerbe 50
 Image 285
 Immutabler Code 130
 Incident Commander 249
 Incident-Response-Ver-
 fahren 249
 Indifferente Kunden 194
 Influencer 206
 Informations-
 beschaffungsphase 99
 Infrastructure as a Service
 (IaaS) 299–300
 Infrastructure Engineer 204
 Infrastruktur
 Kosten 99
 managen 240
 Integration 126
 kontinuierliche 32
 Integrationstests 148
 Interessenskonflikte 35
 Interpretier 270
 Interpretersprachen 270
 Investition in Mitarbeiter 218
 Iteration, schnelle 171

J

Jenkins 239
 Jobbezeichnungen 203

K

Kaizen 230, 313, 320
 Kanban-Board 189
 Kandidatenpool 206
 Kandidatensuche 206
 Kennzahlen 80, 318
 Key Performance Indicators
 (KPIs) 75, 176
 Notfallmanagement 238
 Kodak 67
 Kognitive Ergonomie 236

Kommentare 115
 Kommunikation 38,
 42, 125, 317
 Kommunikationskanäle
 berechnen 182, 214
 Kommunikationsknoten-
 punkte 214
 Komponenten isolieren 122
 Komponententest 147
 Konfigurationsmanage-
 ment-Tool 157
 Konkurrenz 102
 Kontinuierliche
 Bereitstellung 154, 313
 Kontinuierliche
 Integration 153
 Kontinuierliches
 Deployment 154
 Kontinuierliches Lernen 174
 Kontinuierliche Tests 149
 Kontinuierliche
 Verbesserung 313
 Krisen 173
 Kubernetes 289
 Kultur 33, 311, 315
 Kunden identifizieren 102
 Kundenanfragen, Anzahl 78
 Kundenbefragung 190
 Kundenbindung 186
 Kundennutzung 77
 Kundenrückmeldungs-
 prozess 186
 Kundenzufriedenheit 78, 185

L

Lagerhaltung 54
 Lambda-Kalkül 132
 Law of the Hammer 134
 Lead Time 78, 181
 Lean 33
 Lean Manufacturing 54, 230
 Lean MVP 100
 Legacy-Codebase 129
 LinkedIn 204
 Linting-Tools 114
 Lizenzierung 267
 MIT 268
 Log-Aggregator 122
 Logging 130
 Frameworks 130
 Logistik 217

M

Macroservices 280
 Manager überzeugen 67
 Manifest für agile Software-
 entwicklung 30, 96
 Märtyrer 211
 Mean Time between Failures
 (MTBF) 254
 Mean Time to Detection
 (MTTD) 79
 Mean Time to Recover
 (MTTR) 181
 Mean Time to Recovery
 (MTTR) 79
 Mean Time to Repair
 (MTTR) 253
 Meetings,
 Durchschnittskosten 82
 Meisterschaft 219, 221
 Menschliches
 Versagen 229, 236
 Automatisierung 237
 Meritokratie 44
 Message-Queuing-
 Lösung 281
 Messdaten-Management 166
 Messungen 34, 77, 318
 Micromanaging 318
 Microservices 277–278
 in Containern
 deployen 286
 Umstieg 280
 Microsoft Azure 297, 302
 Minimum Viable Product
 (MVP) 100, 319
 Misserfolge 41, 320
 Missionare 65
 Mission Statement 46
 Mitarbeiter
 befähigen 186
 befragen 38
 Mitarbeiterzufrieden-
 heit 77, 318
 messen 82
 MIT-Lizenz 268
 Mittelständisches
 Unternehmen 215
 Monitoring-Tools 168
 Monolithen 278
 Motivation 218, 318
 ergründen 219

intrinsische/
 extrinsische 219
 messen 223
 Moving Left 92
 MRR 77
 MRR Churn 77
 Muda 54
 Multithreading 271
 Mund-zu-Mund-Pro-
 paganda 206
 Mura 54
 Muri 54
 Mushroom Management 135
 Muskelgedächtnis 246
 MVP-Team 201

N

Nachbereitung 258
 Vorfall 258
 Nach links schieben 92
 Nebenläufigkeit 271
 Netflix 247
 Net Promoter Score
 (NPS) 194
 NoOps 160, 237
 Cloud-Dienste 296
 Normierungsgremien 266
 Notfallmanagement 245

O

Objektorientierte
 Programmierung
 (OOP) 131
 Offene Architektur 266
 Offene Standards 266
 Open Computing 266
 OpenShift 289
 Open Source 238
 Container-Schwach-
 stellen 294
 Definition 266–267
 Integration 265
 Lizenzierung 267
 Vor- und Nachteile 268
 OpsLite 296
 Optimierung,
 vorzeitige 134
 Orchestrierer 288
 Organisatorische
 Ergonomie 236
 Overengineering 134
 Notfallmanagement 241

P

Paginierung 285
 Pair Programming 209
 Panamakanal 58
 Parallelisierte
 Testumgebung 146
 Parallelisierung 271
 Parsing-Tool 168
 Patch 156
 Pathos 64
 Pay-as-you-go-
 Service 300, 310
 Perfektion 59, 234
 Performance verbessern 180
 Performance-Tests 149
 Persona 106
 ausarbeiten 107
 Geschlecht 108
 Schlüsselqualitäten 108
 Personalbeschaffung 202, 205
 Personalengpässe 59
 Personalentscheidungen 310
 Personalvermittler 205
 Persönlichkeitstypen 63
 Administrator 63
 Berater 64
 Strategie 63
 Visionär 63
 Phasen eines Un-
 ternehmens 214
 Pipeline 87
 Pipeline-Framework
 entwickeln 88
 Planung 90
 Platform as a Service
 (PaaS) 299–300
 Pods 290
 Post-Incident-Re-
 views 126, 255
 Post-Mortems 255
 Prämien 49
 Preflight 246
 Private Cloud 299
 Produkte verbessern 186
 Produktion simulieren 93
 Produktionsausfälle 92
 Produktionsumgebung 146
 Produktteam
 funktionsübergrei-
 fendes 200
 herkömmliches 201

- Programmiersprache
 - neue 270
 - Wahl 132, 273
 - Programmierung 90
 - funktionale 130, 132
 - objektorientierte 131
 - Projekt, Einschränkungen 98
 - Projektmanagement,
 - agiles 95
 - Projektfumfang 100
 - Promotoren 194
 - Promotorenüberhang 194
 - Proof of Concept (POC) 319
 - Protokollierung 130
 - Prozessoptimierung 52
 - Public Cloud 298
 - Puffer 60
 - Puppet 241
- Q**
- Qualitätssicherung (QS) 145
 - Quartalsziele 176
- R**
- Rahmenbedingungen 98
 - Rapid Iteration 171
 - Reaktion auf Vorfälle 248
 - Regressionstests 148
 - Reibung verringern 183
 - Reine Funktionen 272
 - Rekursion 273
 - Release 151
 - Rolling 163
 - Release Engineer 203–204
 - Release-Pipeline 143
 - Reparaturzeit 253
 - Repository, gemeinsames 137
 - REST 266
 - RESTful-Ansatz 281
 - Risikobereitschaft 233
 - Roadmap 174
 - Rollbacks 158
 - Rolling Release 163
 - Rom 216
 - Root Cause Analysis 255
 - Rufbereitschaft 242
- S**
- Sad-Path-Tests 138
 - Sandbox-Umgebung 144
 - Scale-out 117
 - Scale-up 117
 - Scheitern
 - Freiheit zum 231
 - gutes 230, 312
 - schnelles 227
 - sicheres 228
 - Schlank 33
 - Schnelles Scheitern
 - 171, 320
 - Schuldzuweisung 259
 - Scope Creep 100, 105
 - Scrum 96
 - Secrets 293
 - Selbstanwendung 191
 - Selbstreflexion 176
 - Semantische
 - Versionierung 156
 - Serverless-Betrieb 301
 - Service Level Agreement (SLA) 166–167, 302
 - Service Level Indicator (SLI) 168
 - Service Level Objective (SLO) 168
 - Shadowing-Tag 199
 - Shifting Left 92
 - Sicherheit 118
 - Sicherheitsmaßnahmen
 - Container 292
 - Sicherheitsprinzipien 118
 - Sicherheitsprüfungen 149
 - Silo 30, 36, 40–41, 110, 152
 - Single Point of Failure (SPoF) 228, 247
 - Single Responsibility 135
 - Sinnhaftigkeit 219, 222
 - Site Reliability Engineer 203
 - Site Reliability Engineering (SRE) 167, 314
 - Sizing 177
 - Skalierbarkeit 116
 - Skalierung
 - Entwicklungsteam 213
 - horizontale 117
 - vertikale 117
 - Slack 248
 - SOA-Architektur 278
 - Social Media 206
 - Soft Skills 31, 126
 - Software
 - dokumentieren 114
 - kontinuierlich verbessern 113
 - proprietäre 268
 - wiederverwendbare 113
 - Softwarearchitektur 110
 - Software as a Service (SaaS) 299, 301
 - Software Development Life Cycle (SDLC) *siehe* Softwareentwicklungslebenszyklus
 - Softwareentwicklungslebenszyklus
 - 87
 - optimieren 180
 - Phasen 90
 - Softwaretests 141
 - Soziotechnische
 - Herausforderungen 213
 - Spaghetti-Code 134
 - Sparta 215
 - Speicherverwaltung 273
 - Spezialisten 218
 - SQL-Datenbank 120
 - StackOverflow 238
 - Staging 93, 143
 - Staging-Umgebung 146
 - Standardverfahren 247
 - Standpunkte, unterschiedliche 126
 - Start-up 215
 - State of DevOps Report 66
 - Stellenbezeichnung 203
 - Stillschweigen 317
 - Störung, technische 235
 - Story Points 178
 - Stubs 148
 - Swarm 290
 - Sys Admin 204
 - System, verteiltes 277
- T**
- Take-Home-Test 208
 - Team 197
 - Cloud Transformation 201
 - funktionales 198
 - gerade Mitgliederzahl 182
 - kleines 181, 217
 - MVP 201
 - Teamleistung 48
 - Teamwork 40
 - Tech Chief 249
 - Technische Schulden 175
 - Telemetrie 166
 - Terminvorgaben 98

Testabdeckung,
 automatisierte 79
 Test-Driven Development
 (TDD) 137
 Testen 141
 Tester, spezialisierte 142
 Test-Framework 137
 Testphase, erste 144
 Testsuite,
 automatisierte 142
 Testumgebung 145
 parallelisierte 146
 Tools
 beliebte 238
 kommerzielle 266
 Toyota 55
 Toyota Way 55, 73, 230
 Trägheitsaspekt 62
 Transportverluste 53
 Triage 250

U

Überproduktion 53
 Überzeugungskraft 31
 Umfrage 38–39, 82
 beim Kunden 190
 Vor- und Nachteile 190
 Umgebung, lokale 144
 Unabhängigkeit 217
 Underperformer 211
 Unsicherheit 62
 Unternehmenskultur
 30, 37, 315
 beurteilen 38
 Clan 44
 Hierarchie 44
 Holokratie 44
 Kategorien 39
 Meritokratie 44

Unternehmensleitbild 46
 Unveränderbarer Code 130
 Unvollkommenheit 181
 Usability 119
 User Experience (UX) 104
 User Stories 100, 126, 223

V

Velocity 177
 Vendor Lock-in 134
 Veränderungen
 Angst vor 61
 implementieren 189
 Widerstände 63
 zu schnelle 319
 Verbesserung,
 kontinuierliche 313
 Verfügbarkeit 35, 79, 166, 229
 Verschwendung 52
 Flaschenhals 56
 vermeiden 56
 Versionierung 156
 semantische 156
 Verteiltes System 277
 Vertrauen 174, 319
 Videoanruf 248
 Virtuelle Maschinen
 (VMs) 286
 Vision Statement 46–47
 Visuelle Tests 148
 Vorfall 235
 Aufbereitung 257
 einschätzen 250
 erkennen 257
 Eskalation 248
 Fragen stellen 260
 Kommunikationsmittel 248
 nachbereiten 258
 Phasen 257

Reaktion 248, 257
 Reflexion 257
 Schuldzuweisungen 259
 Triage 250
 Wiederherstellung 257
 zeitlicher Ablauf 259

Vorstellungs-
gespräch 202, 310

W

Wachstumsmen-
 talität 230, 320
 Wall of Confusion 110
 Wandel als
 Konstante 309
 Warnmeldungen 183
 Wartezeiten 52
 Wartungsfreundlicher Code
 116, 128
 Wasserfallmethode 88
 Werkzeugauswahl 319
 Wettbewerbsfähigkeit 311
 Whiteboard-Interview 207
 modifizieren 207
 Wichtigkeit vs.
 Dringlichkeit 249
 Widerstand 72
 Window Size 163
 Wissenstransfer 115

Z

Zeitdruck 98
 Zeitnutzung
 protokollieren 182
 Ziele 47
 Zuhören 135
 Zuverlässigkeit 120
 Zwei-Pizza-Team 181