

# Inhalt

<b>Einleitung</b> .....	<b>7</b>
<b>1 Grundlegende Konzepte</b> .....	<b>11</b>
Was ist Versionskontrolle, und warum brauche ich sie? .....	11
Die Grundlagen von Git. ....	13
<b>2 Git installieren und konfigurieren</b> .....	<b>23</b>
Installation .....	23
Git konfigurieren .....	24
<b>3 Arbeiten mit Git</b> .....	<b>27</b>
Hilfe finden .....	27
Das Git-Repository .....	28
Dateien ignorieren und von der Versionierung ausschließen .....	35
Commits erstellen .....	37
Änderungen synchronisieren .....	45
Mit Branches arbeiten .....	50
Änderungen zusammenführen .....	54
Änderungen temporär lokal speichern .....	65
Änderungen nachvollziehen und betrachten .....	67
Änderungen rückgängig machen .....	72
<b>4 Git-Onlinedienste</b> .....	<b>81</b>
Die Relevanz von GitHub .....	82
Ein Repository auf GitHub anlegen .....	82
Pull-Requests .....	86
Issues: Projektmanagement und Bug-Reports .....	94

<b>5</b>	<b>Typische Git-Workflows</b> .....	<b>99</b>
	Trunk-based: ausschließlich im Haupt-Branch arbeiten .....	99
	Git Feature Branch Workflow .....	102
	Gitflow: der Workflow für versionierte Software .....	103
	Durch den Fork-basierten Workflow zu Open-Source-Projekten beitragen .....	105
<b>6</b>	<b>Good Practices</b> .....	<b>109</b>
	Gute Commits .....	109
	Geschichte nur im Notfall neu schreiben .....	114
	Den Überblick über Branches behalten .....	116
	Regelmäßig aufräumen .....	117
	Entwicklungsstränge häufig zusammenführen und Konflikten nicht aus dem Weg gehen .....	118
	Langlebige Feature-Banches vermeiden .....	119
<b>7</b>	<b>Häufige Fehler und Probleme</b> .....	<b>123</b>
	Lokaler Stand und Remote-Stand weichen voneinander ab .....	123
	Detached-HEAD-Fehlermeldung .....	124
	Du hast im falschen Branch gearbeitet .....	125
	Git-Diff zeigt keine Änderungen an .....	126
	Keine Tracking-Information für den aktuellen Branch vorhanden .....	126
<b>8</b>	<b>Fortgeschrittenere Kommandos, Tipps und Tricks</b> .....	<b>129</b>
	Im Reflog merkt sich Git fast alles .....	129
	Kürzel für Git-Befehle erstellen .....	131
	Pre- und Post-Commit-Hooks .....	131
<b>9</b>	<b>Git als Baustein moderner und agiler Softwareentwicklung</b> .....	<b>133</b>
	DevOps-Kultur .....	133
	Kontinuierlich integrieren und ausliefern mit Git .....	134
	Infrastructure as Code .....	137
<b>10</b>	<b>Die wichtigsten Kommandos zum Nachschlagen</b> .....	<b>139</b>
<b>11</b>	<b>Glossar</b> .....	<b>143</b>
	<b>Index</b> .....	<b>145</b>