
Inhalt

Vorwort	XI
Ist dieses Buch das richtige für mich?	XI
Für wen ist dieses Buch nicht geeignet?	XII
Der Leser oder die Leserin?	XIII
Wie ist dieses Buch zu lesen?	XIII
Konventionen in diesem Buch	XIV
 Danksagung	 XVI
 1 Was ist GitHub, und wofür brauche ich es?	 1
Was bietet GitHub?	2
Einsatzgebiete von GitHub	3
Git, GitHub, GitLab – alles das Gleiche?	4
Mit welchen Kosten muss ich rechnen?	6
 2 GitHub – Wir verschaffen uns einen Überblick	 9
Anwendungsfälle für GitHub (oder: Was will ich da eigentlich?)	12
Informationen finden (interessierte Anwenderin)	13
Dateien finden und herunterladen (Hilfe suchender Programmierer)	17
 3 Die Basis: Das erste eigene GitHub-Projekt	 19
Account anlegen	20
Account schützen	22
Unsichtbar werden – die eigene Mailadresse schützen	23
Das erste eigene Repository anlegen	24
Eine inhaltliche Änderung am Projekt vornehmen	26
Den ersten Ablauf üben – Issue anlegen und bearbeiten	31
Einen Issue anlegen	32
Den Issue bearbeiten und schließen	35

Ein bestehendes Repository löschen	39
Ein bestehendes Projekt hochladen	40
4 Die wichtigsten Grundlagen für eigene GitHub-Projekte	43
Den zweiten Ablauf üben – Branches, Pull-Requests und Merges	44
Branch – unterschiedliche Handlungsstränge aufmachen	44
Änderungen auf einem Branch vornehmen	48
Pull-Request – Änderungen in Branches aufzeigen	53
Merge – Änderungen aus Pull-Requests übernehmen	59
Reviews durchführen	61
Reviewer manuell anfordern	65
Reviews automatisch zuweisen – CODEOWNERS	65
Gutes schützen – Protected Branches	67
Genehmigung vorschreiben – Required Reviews	68
Genehmigung automatisch zurückziehen	72
Genehmigung durch Eigentümer*innen vorschreiben	73
Den Laden sauber halten – Vorlagen, Diskussionen eingrenzen	75
Vorlagen für Issues	75
Vorlagen für Pull-Requests	79
Für Ruhe sorgen (Teil 1) – Locking Conversations	80
Für Ruhe sorgen (Teil 2) – Interaction Limits	83
5 Rechtliches – Open-Source-Lizenzen	85
Warum Lizenzierung wichtig ist	85
Lizenz Marke Eigenbau	86
Welche Lizenzen gibt es?	88
Softwarecode	88
Musik, Bilder und Texte	90
Was wählen andere als Lizenz?	92
Welche Lizenz ist die richtige für mich?	94
Wo finde ich mehr Infos und Unterstützung zu Lizenzen?	95
Unterstützung bei der Wahl der richtigen Lizenz	95
Tools und Informationen zu/über Lizenzen	96
Eine Lizenz zu einem Repository hinzufügen	97
6 Unterstützung für GitHub-Projekte finden	101
Wie bringt man Leute dazu, beim eigenen Projekt mitzumachen?	101
Dein Projekt auffindbar machen	102
Dein Projekt anschaulich beschreiben	104
Dein Projekt bekannt machen	111
Dein Projekt (gegebenenfalls) zugänglich machen (Rechtevergabe)	114

Ein Projekt finden, das du unterstützen möchtest.	117
Wer bin ich und, wenn ja, wie viele?	117
Fremdes Projekt suchen	117
Fremdes Projekt begutachten	124
Fremdes Projekt unterstützen – Fork	126
7 Ein Projekt lokal mit Git verwalten	135
Warum GitHub allein manchmal nicht ausreicht	135
Git, was ist das? – Eine kurze Einführung	136
Versionsverwaltung	136
Dezentral.	138
Exkurs: Umgang mit der Konsole	139
Git installieren und einrichten	141
Exkurs: Die Konsole für Git einrichten am Beispiel Bash (für Fortgeschrittene).	143
Drei Schritte, um einen Branch farbig anzuzeigen	144
Anpassen der .bashrc	145
Tiefer einsteigen	146
Wie Git tickt – Staging	147
Das eigene Projekt mit Git verwalten	150
Das Arbeitsverzeichnis initialisieren	150
Eine neue Datei ins lokale Repository einfügen	151
Eine Datei im lokalen Repository ändern	152
Dateien von der lokalen Versionsverwaltung ausschließen	153
Branching in Git	154
Branches erzeugen	154
Zwischen Branches wechseln	155
Binärdateien mit Git verwalten	156
Installation von Git LFS	157
Git LFS einrichten.	158
Sich weiter schlaumachen über Git	160
Oldschool: Bücher	160
Neumodischer Kram: Internet	160
8 Git und GitHub im Zusammenspiel	163
Szenario 1: Lokales Git-Projekt auf GitHub hochladen	165
Lokal ein Git-Repository mit einer Datei anlegen.	166
Leeres Repository auf GitHub anlegen	166
Das Git- mit dem GitHub-Repository verknüpfen	167
Git-Repository auf GitHub hochladen (pushen)	170
Lokal Änderungen vornehmen und diese auf GitHub hochladen.	172

Szenario 2: Projekt auf GitHub lokal zu Git holen	173
Ein neues GitHub-Repository mit einer Datei erstellen	174
Das GitHub-Repo mittels Git lokal klonen	174
Das GitHub-Repo anpassen und die Änderung in die lokale Git-Arbeitsumgebung holen.	175
Szenario 3: Geforktes Projekt auf GitHub lokal zu Git holen.	179
Wir forken auf GitHub ein Projekt	180
Wir klonen den Fork lokal mittels Git.	180
Wir legen in Git ein zweites Remote-Repository fest.	181
Wir aktualisieren den lokalen Klon aus dem Originalprojekt	182
Szenario 4: Lokale Änderung an Originalprojekt übergeben	183
Wir richten alles so ein, wie in Szenario 3 beschrieben	184
Wir editieren lokal eine Datei und pushen sie zum Fork auf GitHub	184
Wir erstellen einen Pull-Request aus dem Fork an das Originalprojekt.	185
Wir üben uns in Geduld und warten auf das Mergen des Pull-Requests	185
Merge-Konflikte lösen.	186
Wie entstehen Merge-Konflikte?	186
Konflikte auflösen mit GitHub (Webeditor)	187
Konflikte auflösen mit Git (Konsole)	189
Log-in-Erleichterungen bei HTTPS	194
Zugangsdaten auf Zeit zwischenspeichern (meine Empfehlung)	194
Zugangsdaten dauerhaft speichern	195
9 Der GitHub Marketplace – Actions und Apps	197
Was können Actions und Apps?	197
Eine App aus dem Marketplace installieren	198
App installieren	199
App anpassen	202
Eine Action aus dem Marketplace installieren	204
Action installieren	205
Action ausprobieren und feinjustieren.	208
Hinter den Kulissen einer Action	210
Eine eigene Action erstellen (für Fortgeschrittene)	212
Einige grundlegende Begriffe	212
Anatomie einer Action	215
Unseren Anwendungsfall einrichten	216
Unseren Anwendungsfall verstehen.	219
Passwörter geheim halten – GitHub Secrets.	222

10 Pimp my Repo – Weitere GitHub-Features	225
Websites aus GitHub generieren (GitHub Pages)	225
GitHub Pages einrichten	226
GitHub Pages verschönern mit dem Theme Chooser	227
GitHub Pages ausbauen – die Navigation einrichten	229
GitHub Pages – weitere Themes	233
Angriff der Klone – Repo-Templates anlegen	235
Eigene Projektboards – mit Projects den Überblick behalten	236
Grundlegendes zu Projektboards	236
Ein eigenes Projektboard erstellen	239
Projektboard automatisieren	241
11 Nützliches und Kurioses rund um GitHub	245
GitHub auf der Kommandozeile	245
GitHub CLI	245
Hub	247
GitHub-API (für Fortgeschrittene)	247
Sich mit GitHub weiter auseinandersetzen	248
GitHub Learning Lab	248
Weitere Ressourcen zum Recherchieren	250
Editoren und Handy-Apps	250
Klein und schlank – Atom	251
Visual Studio Code	252
Für Website-Gestalter – Brackets	252
GitHub Desktop	253
GitHub auf dem Handy – GitHub Mobile	253
GitHub auf dem Handy – Octodroid	254
Nützliches und kleine Spielereien	255
Übersetzungsmanagementtools – crowdin und Weblate	255
Zeigen, wo man steht – Badges	256
Sag es mit einem Bild – Gitmoji	258
Ideen für eigene Repositories – ohne programmieren	259
A Gängige Git-Befehle zum Nachschlagen	261
B Quellcode	263
C Glossar (oder: Was bedeutet noch mal ...?)	265
Index	271