

Vibe Coding mit Bolt

Web-Apps mit KI planen, entwickeln
und betreiben

DAS INHALTS- VERZEICHNIS

» Hier geht's
direkt
zum Buch

Vorwort	11
Einleitung	15
1 Einführung in Bolt.new und Prompt-first-Development	23
Was ist Bolt?	23
Die Vorteile von Prompt-first-Development	24
Wer profitiert von Bolt?	25
Prompt-first-Development: ein einfaches Beispiel	26
KI als zuarbeitender Assistent	29
In React denken	30
Komponenten: die Bausteine von Benutzeroberflächen	30
Props: Daten an Komponenten übergeben	31
State: dynamische Daten und Interaktionen verwalten	32
Wie React in das Arbeiten mit Bolt passt	34
Zusammenfassung	35
2 Erste Schritte mit Bolt	37
Registrierung und Start von Bolt	37
Ihr erster Prompt	40
Den Workspace erkunden und anpassen	43
Metaprompts verstehen: die KI-Richtlinien Ihres Projekts	44
Beispiel: Metaprompting	46
Zusammenfassung	46
3 Seien Sie präzise, seien Sie konkret: Prompt Engineering und iterative Entwicklung	49
Erwartungsmanagement: Grenzen und Stärken von KI	49

Tipps für effektives Prompting	51
Beginnen Sie allgemein, verfeinern Sie sukzessive	53
Schritt für Schritt, Aufgabe für Aufgabe	54
Vibe Coding und kreative Exploration	56
Wenn die Dinge aus dem Ruder laufen	57
Beispielhafte Prompt-Szenarien	59
Best Practices für einen Prompt-first-Workflow	62
Zusammenfassung	64
4 Fixes, Checkpoints und Rollbacks	65
Troubleshooting	65
Checkpoints und Rollbacks nutzen	67
Iteration und Versionierung als Grundlage für schnelles Experimentieren . .	71
Wann man eingreifen und wann man neu anfangen sollte	73
Wann man eingreifen sollte	73
Wann man von vorne beginnen sollte	73
Behalten Sie den Polarstern im Blick	74
Zusammenfassung	74
5 Entwicklung einer Movie-Explorer-App, Teil 1: Browsing und Details	75
Einrichten des API-Keys	76
Einrichten eines neuen Bolt-Projekts	76
Konfiguration des TMDB-API-Keys	78
Abruf und Anzeige beliebter Filme	79
Hinzufügen einer Suchfunktion	82
Implementierung der Filmdetailseite	85
Wenn Bolt React Router verwendet	86
Falls Bolt Next.js verwendet	87
Testen der Detailseite	87
Feinschliff der UI	88
Testen der Anwendung	92
Häufige Probleme und Troubleshooting	93
Tipps und Best Practices	95
Zusammenfassung	97

6	Entwicklung einer Movie-Explorer-App, Teil 2: Favoriten und Authentifizierung	99
	Supabase-Setup	100
	Ein Supabase-Konto erstellen	100
	Supabase mit Bolt verbinden	100
	Supabase-Projekt mit einer Favoriten-Tabelle einrichten	101
	Authentifizierung mit Supabase integrieren	103
	Bolt-Projekt mit Supabase verbinden	104
	Bolt per Prompt um eine Authentifizierungs-UI erweitern	104
	Favoritenfunktion implementieren	109
	Favorit-Button hinzufügen	109
	Seite »Meine Favoriten« hinzufügen	112
	Häufige Probleme und Troubleshooting	115
	Tipps und Best Practices	117
	Zusammenfassung	119
7	Bereitstellung Ihrer App mit Netlify	121
	Deployment-Schritte	122
	Netlify in Bolt verbinden und konfigurieren	122
	Überwachung des Deployment-Prozesses	123
	Testen der Live-Site	124
	Netlify-Deployment anpassen (optional)	125
	Bei weiteren Änderungen neu deployen	125
	Häufige Probleme und Troubleshooting	126
	Tipps und Best Practices	127
	Zusammenfassung	128
8	Entwicklung eines Supabase-gestützten Workout Tracker	131
	Erste Schritte	131
	Setup	132
	Planung des Datenbankschemas	132
	Tabellen in Supabase einrichten	133
	Row-Level Security (RLS) implementieren	133
	Entwicklung des Workout Tracker	134
	Das Model-View-Controller-(MVC-)Muster	140
	Fortgeschrittene Funktionen (optional)	141

Häufige Probleme und Troubleshooting	143
Best Practices	144
Zusammenfassung	146
9 Entwicklung einer Einkaufslisten-App mit Bolts Figma-Integration...	147
Die Design-to-Code-Pipeline	147
Einrichten Ihrer Designumgebung	148
Projektanforderungen	148
Erstellen des Designs für die Einkaufsliste	149
Vorbereitung des Exports	151
Importieren in Bolt	152
Vom statischen Design zur dynamischen Anwendung	155
Interaktivität hinzufügen	156
Implementierung der Checkbox-Funktionalität	156
Einträge entfernen	157
Persistenz hinzufügen	158
Responsivität verbessern	158
User Experience verbessern	159
Häufige Probleme und Troubleshooting	159
Design Patterns für bessere Importe	160
Mögliche Erweiterungen der App	161
Zusammenfassung	161
10 StackBlitz für Bolt-Anwender	163
Einführung in StackBlitz	164
Exportieren Ihres Bolt-Projekts zu StackBlitz	166
Entwicklung im StackBlitz Cloud Editor	168
Rundgang durch die StackBlitz-Oberfläche	169
Sofortiges Feedback bei der Bearbeitung von Code	171
Debugging und Testen in StackBlitz	173
Unterschiede im Workflow zwischen Bolt und StackBlitz	174
Von KI-Prompts zum direkten Coding	174
Anpassen und verfeinern	175
Projektstruktur und Transparenz	176
KI-Unterstützung in StackBlitz	177

Speichern und Versionierung	177
Verwendung des integrierten Terminals	179
GitHub-Integration für Versionsverwaltung und Zusammenarbeit	182
Zusammenfassung	185
11 Mit Bolt und Expo mobile Apps entwickeln.....	187
Setup	188
Initialer Prompt und Standortdienste	188
Integration einer Wetter-API	190
Gerätefunktionen verwenden: Kamera und Benachrichtigungen	194
Zugriff auf die Kamera	194
Benachrichtigungen planen	196
Styling der React-Native-App	198
Testen der App	200
Live-Tests während der Entwicklung	201
Automatisiertes Testen	202
Testen von Production-Builds	202
Adaptierung der Einkaufslisten-App für mobile Plattformen	203
Zusammenfassung	205
12 Fortgeschrittene Bolt-Tipps.....	207
Debugging	207
Die Live-Entwicklungsumgebung von Bolt verstehen	208
Debugging und Troubleshooting in Bolt	209
Best Practices beim Debuggen von Bolt-Apps	210
Debugging-Beispiel	212
Weniger bekannte Bolt-Features	214
Gezieltes Targeting und Sperren von Dateien	214
Dateien ignorieren: fortgeschrittenes Kontextmanagement	215
Import und Integration externer Tools	216
Chrome-Erweiterungen: Supercharge und Snippets	217
Praxisbeispiele für versteckte Funktionen	218
Entwicklung von (Backend-gestützten) Full-Stack-Apps in Bolt	219
Best Practices für die Nutzung der Backend-Funktionalität	221
Beispiel: Ausführen einer Backend-Aufgabe in Bolt	224

Deployment und Export Ihres Backends.....	224
Abschließende Gedanken zum Backend in Bolt.....	225
Zusammenfassung.....	226
Index.....	229