

Auf einen Blick

1	Die ersten Schritte mit React	23
2	Die ersten Schritte im Entwicklungsprozess	41
3	Die Grundlagen von React	67
4	Ein Blick hinter die Kulissen – weiterführende Themen	113
5	Die Hooks-API von React	161
6	Typsicherheit in React-Applikationen mit TypeScript	193
7	Styling von React-Komponenten	221
8	Absichern einer React-Applikation durch Tests	247
9	Formulare in React	279
10	Komponentenbibliotheken in einer React-Applikation	313
11	Navigation innerhalb einer Applikation – der Router	349
12	Zentrales State-Management mit Redux	369
13	Umgang mit Asynchronität und Seiteneffekten in Redux	405
14	Serverkommunikation mit GraphQL und dem Apollo-Client	457
15	Internationalisierung	487
16	Universal React Apps mit Server-Side Rendering	511
17	Progressive Web Apps	531
18	Native Apps mit React Native	557

Inhalt

Materialien zum Buch	16
Geleitwort des Fachgutachters	17
Vorwort	19

1 Die ersten Schritte mit React 23

1.1 Was ist React?	23
1.1.1 Single-Page-Applikationen	24
1.1.2 Die Geschichte von React	25
1.2 Warum React?	28
1.2.1 Releasezyklus	28
1.3 Die wichtigsten Begriffe und Konzepte der React-Welt	29
1.3.1 Komponenten und Elemente	29
1.3.2 Datenfluss	31
1.3.3 Renderer	32
1.3.4 Reconciler	32
1.4 Ein Blick in das React-Universum	35
1.4.1 State-Management	35
1.4.2 Router	35
1.4.3 Material UI	36
1.4.4 Jest	36
1.5 Thinking in React	36
1.5.1 Die Oberfläche in eine Komponentenhierarchie zerlegen	37
1.5.2 Eine statische Version in React implementieren	37
1.5.3 Den minimalen UI State bestimmen	37
1.5.4 Den Speicherort des States bestimmen	37
1.5.5 Den inversen Datenfluss modellieren	38
1.6 Die Beispielapplikation	38
1.7 Zusammenfassung	39

2	Die ersten Schritte im Entwicklungsprozess	41
2.1	Playgrounds für React	41
2.1.1	CodePen – ein Playground für die Webentwicklung	42
2.2	Lokale Entwicklung	44
2.2.1	React in eine HTML-Seite einbinden	45
2.3	Der Einstieg in die Entwicklung mit React	48
2.3.1	Anforderungen	48
2.3.2	Installation von Create React App	50
2.3.3	React Scripts	57
2.3.4	Serverkommunikation im Entwicklungsbetrieb	60
2.3.5	Verschlüsselte Kommunikation während der Entwicklung	61
2.4	Die Struktur der Applikation	61
2.5	Fehlersuche in einer React-Applikation	63
2.5.1	Arbeiten mit den React Developer Tools	65
2.6	Applikation bauen	65
2.7	Zusammenfassung	66
3	Die Grundlagen von React	67
3.1	Vorbereitung	67
3.1.1	Die Applikation aufräumen	68
3.2	Funktionskomponenten	70
3.2.1	Eine Komponente pro Datei	72
3.3	JSX – Strukturen in React definieren	77
3.3.1	Ausdrücke in JSX	79
3.3.2	Iterationen – Schleifen in Komponenten	81
3.3.3	Bedingungen in JSX	83
3.4	Props – Informationsfluss in einer Applikation	87
3.4.1	Props der »Card«-Komponente	87
3.4.2	Typsicherheit mit PropTypes	89
3.5	Klassenkomponenten	92
3.6	Lokaler State	94
3.7	Event-Binding – Reaktion auf Benutzerinteraktionen	98
3.7.1	Gewählte Eigenschaft anzeigen	100

3.7.2	Karte des Gegenspielers aufdecken	105
3.7.3	Karten vergleichen	107
3.8	Zusammenfassung	110
4	Ein Blick hinter die Kulissen – weiterführende Themen	113
<hr/>		
4.1	Komponenten-Lifecycle	113
4.1.1	Constructor	115
4.1.2	»getDerivedStateFromProps«	116
4.1.3	»render«	117
4.1.4	»componentDidMount«	118
4.1.5	»shouldComponentUpdate«	119
4.1.6	»getSnapshotBeforeUpdate«	120
4.1.7	»componentDidUpdate«	121
4.1.8	»componentWillUnmount«	122
4.1.9	Unsafe Hooks	124
4.2	Serverkommunikation	125
4.2.1	Serverimplementierung	125
4.2.2	Serverkommunikation mit der fetch-API	127
4.2.3	Serverkommunikation mit Axios	129
4.3	Container Components	130
4.3.1	Auslagern von Logik in eine Container Component	131
4.3.2	Einbindung der Container Component	134
4.3.3	Implementierung der Presentational Component	135
4.4	Higher-Order Components	136
4.4.1	Eine einfache Higher-Order Component	137
4.4.2	Einbindung einer Higher-Order Component in die Beispielapplikation	139
4.4.3	Implementierung der inneren Komponente	141
4.4.4	Einbindung der Higher-Order Component	142
4.5	Render Props	143
4.5.1	Alternative Namen für Render Props	145
4.5.2	Integration der Render Props in die Applikation	146
4.6	Error Boundaries	147
4.6.1	Loggen von Fehlern mit »componentDidCatch«	147
4.6.2	Alternative Darstellung im Fehlerfall mit »getDerivedStateFromError«	148

4.7	Kontext	151
4.7.1	Die Context-API	151
4.7.2	Einsatz der Context-API in der Beispielapplikation	154
4.8	Fragments	158
4.9	Zusammenfassung	159

5 Die Hooks-API von React 161

5.1	Ein erster Überblick	162
5.1.1	Die drei Basis-Hooks	162
5.1.2	Weitere Bestandteile der Hooks-API	163
5.2	Die Basis-Hooks im Einsatz	164
5.2.1	Lokaler State in Funktionskomponenten mit dem State-Hook	164
5.2.2	Komponenten-Lifecycle mit dem Effect-Hook	170
5.2.3	Zugriff auf den Kontext mit dem Context-Hook	177
5.3	Custom Hooks	178
5.3.1	Ein Beispiel für einen Custom Hook	179
5.3.2	Praktische Anwendung der Custom Hooks	180
5.4	Rules of Hooks – was Sie beachten sollten	185
5.4.1	Regel #1: Hooks nur auf oberster Ebene ausführen	186
5.4.2	Regel #2: Hooks dürfen nur in Funktionskomponenten oder Custom Hooks verwendet werden	186
5.5	Umstieg auf Hooks	187
5.6	Performance	188
5.6.1	Der Callback-Hook	188
5.6.2	Pure Components	190
5.7	Zusammenfassung	191

6 Typsicherheit in React-Applikationen mit TypeScript 193

6.1	Was bringt ein Typsystem?	193
6.2	Die verschiedenen Typsysteme	194

6.3	Typsicherheit in einer React-Applikation mit Flow	195
6.3.1	Einbindung in eine React-Applikation	195
6.3.2	Die wichtigsten Features von Flow	198
6.3.3	Flow in React-Komponenten	198
6.4	Einsatz von TypeScript in einer React-Applikation	200
6.4.1	Einbindung in eine React-Applikation	200
6.4.2	Konfiguration von TypeScript	203
6.4.3	Die wichtigsten Features von TypeScript	204
6.4.4	Typdefinitionen – Informationen über Drittanbieter-Software	204
6.5	TypeScript und React	205
6.5.1	TypeScript zu einer bestehenden Applikation hinzufügen	205
6.5.2	Constructor Shortcut – Parameter Properties	206
6.5.3	Klassenkomponenten mit TypeScript	208
6.5.4	Kontext in TypeScript	210
6.5.5	Funktionskomponenten in TypeScript	211
6.5.6	Die Hooks-API in TypeScript	214
6.6	Zusammenfassung	220

7 Styling von React-Komponenten 221

7.1	CSS Imports	221
7.1.1	Vor- und Nachteile von CSS Imports	222
7.1.2	Umgang mit Klassennamen	223
7.1.3	Verbesserte Behandlung von Klassennamen mit der »classnames«-Bibliothek	226
7.1.4	Verwendung von Sass als CSS-Präprozessor	227
7.2	Inline-Styling	230
7.3	CSS in JS	232
7.3.1	Die Funktionsweise von Radium	235
7.4	CSS-Module	236
7.5	Styled Components	239
7.5.1	Installation und erste Styles	239
7.5.2	Bedingte Styles und Pseudoselektoren	241
7.5.3	Weitere Features von Styled Components	243
7.6	Zusammenfassung	244

8 Absichern einer React-Applikation durch Tests 247

8.1 Die ersten Schritte mit Jest	248
8.1.1 Installation und Ausführung	249
8.1.2 Organisation der Tests	250
8.1.3 Jest – die Grundlagen	250
8.1.4 Aufbau eines Tests – Triple A	252
8.1.5 Die Matcher von Jest	254
8.1.6 Gruppierung von Tests – Testsuites	255
8.1.7 Setup- und Teardown-Routinen	256
8.1.8 Tests überspringen und exklusiv ausführen	257
8.1.9 Umgang mit Exceptions	259
8.1.10 Testen von asynchronen Operationen	261
8.2 Testen von Hilfsfunktionen	264
8.3 Snapshot-Testing	265
8.3.1 Snapshot-Tests für die Komponenten der Beispielapplikation	266
8.4 Komponenten testen	269
8.4.1 Test der »Card«-Komponente mit dem Test-Renderer	269
8.4.2 Interaktion mit einer Komponente testen	270
8.4.3 Klassenkomponenten testen	271
8.5 Umgang mit Serverabhängigkeiten	272
8.6 Bibliotheken für komfortableres Testen	274
8.6.1 Die React-Testing-Library	274
8.7 Zusammenfassung	278

9 Formulare in React 279

9.1 Uncontrolled Components	279
9.1.1 Umgang mit Referenzen in React	279
9.2 Controlled Components	292
9.2.1 Synthetic Events	297
9.3 Upload von Dateien	298
9.4 Formularvalidierung mit Formik	301
9.4.1 Erzeugung eines Validierungsschemas	302

9.4.2	Styling der Formularelemente	304
9.4.3	Aufbau eines Formulars mit Formik	305
9.5	Zusammenfassung	310

10 Komponentenbibliotheken in einer React-Applikation 313

10.1	Installation und Integration von Material-UI	313
10.2	Listendarstellung mit der »Table«-Komponente	314
10.2.1	Filtern der Liste der Tabelle	317
10.2.2	Tabelle sortieren	319
10.3	Grids und Breakpoints	323
10.4	Icons	325
10.5	Datensätze löschen	328
10.5.1	Löschoperation vorbereiten	328
10.5.2	Implementierung des Bestätigungsdialogs	329
10.5.3	Datensätze löschen	331
10.6	Neue Datensätze erzeugen	334
10.6.1	Erzeugen von Datensätzen vorbereiten	334
10.6.2	Umbau der »Form«-Komponente	336
10.6.3	Integration des Formularioals	340
10.7	Datensätze editieren	343
10.8	Zusammenfassung	348

11 Navigation innerhalb einer Applikation – der Router 349

11.1	Installation und Einbindung	350
11.1.1	Die Routerkomponenten	350
11.2	Navigation in der Applikation	351
11.2.1	Nur eine Route aktivieren	352
11.2.2	Navigationsleiste in der Applikation	353
11.2.3	Integration der Navigationsleiste	356
11.3	»Not found«	359

11.4	Auth Redirect	361
11.5	Dynamische Routen	363
11.5.1	Subrouten definieren	363
11.5.2	Navigation zu den Subrouten	365
11.6	Zusammenfassung	367

12 Zentrales State-Management mit Redux 369

12.1	Die Flux-Architektur	369
12.1.1	Der zentrale Datenspeicher – der Store	370
12.1.2	Die Anzeige der Daten mit den Views	370
12.1.3	Actions – die Beschreibung von Änderungen	371
12.1.4	Der Dispatcher – die Schnittstelle zwischen Actions und dem Store	372
12.2	Installation von Redux	373
12.3	Den zentralen Store konfigurieren	373
12.3.1	Debugging mit den Redux Dev Tools	375
12.4	Umgang mit Änderungen am Store mit Reducern	378
12.4.1	Der »Admin«-Reducer	378
12.4.2	Einbindung des »Admin«-Reducers	380
12.5	Verknüpfung von Komponenten und Store	381
12.5.1	Eine erste Container Component	382
12.5.2	Selektoren	383
12.5.3	Selektoren mit Reselect umsetzen	387
12.6	Beschreibung von Änderungen mit Actions	391
12.6.1	Löschen von Datensätzen	391
12.6.2	Typsichere Actions	394
12.7	Ausblick Redux-React-Hook	400
12.8	Zusammenfassung	403

13 Umgang mit Asynchronität und Seiteneffekten in Redux 405

13.1	Middleware in Redux	405
13.1.1	Eine eigene Middleware implementieren	406

13.2	Redux mit Redux Thunk	407
13.2.1	Installation von Redux Thunk	407
13.2.2	Daten vom Server lesen	408
13.2.3	Umgang mit Fehlern	413
13.2.4	Löschen von Datensätzen	418
13.2.5	Anlegen und Modifizieren von Datensätzen	421
13.3	Async/Await und Generators – Redux Saga	424
13.3.1	Installation und Einbindung von Redux Saga	425
13.3.2	Daten vom Server laden	426
13.3.3	Bestehende Daten löschen	428
13.3.4	Datensätze erstellen und modifizieren mit Redux Saga	430
13.4	State-Management mit RxJS – Redux Observable	433
13.4.1	Installation und Einbindung von Redux Observable	434
13.4.2	Lesender Zugriff auf den Server mit Redux Observable	436
13.4.3	Löschen mit Redux Observable	438
13.4.4	Datensätze anlegen und editieren mit Redux Observable	439
13.5	JWT zur Authentifizierung	441
13.5.1	Erweiterung des »login«-Moduls	442
13.5.2	Aus Redux heraus navigieren	448
13.6	Zusammenfassung	455

14 Serverkommunikation mit GraphQL und dem Apollo-Client 457

14.1	Einführung in GraphQL	457
14.1.1	Die Charakteristik von GraphQL	457
14.1.2	Die Nachteile von GraphQL	458
14.1.3	Die Prinzipien von GraphQL	459
14.2	Apollo, ein GraphQL-Client für React	463
14.2.1	Installation und Einbindung in die Applikation	464
14.2.2	Lesender Zugriff auf den GraphQL-Server	465
14.2.3	Zustände einer Anfrage	468
14.2.4	Typunterstützung im Apollo-Client	469
14.2.5	Löschen von Datensätzen	472
14.3	Die Apollo Client Devtools	475

14.4	Lokales State-Management mit Apollo	476
14.4.1	Den lokalen State initialisieren	476
14.4.2	Den lokalen State benutzen	478
14.5	Authentifizierung	483
14.6	Zusammenfassung	484

15 Internationalisierung 487

15.1	Einsatz von React Intl	487
15.1.1	Die Sprache des Browsers verwenden	490
15.1.2	Erweiterung der Navigation um Sprachumschaltung	491
15.2	Verwendung von Platzhaltern	496
15.3	Programmatische Übersetzungen	497
15.4	Formatierung von Zahlen	499
15.5	Singular und Plural	502
15.6	React Intl und Redux	505
15.7	Zusammenfassung	509

16 Universal React Apps mit Server-Side Rendering 511

16.1	Wie funktioniert Server-Side Rendering?	512
16.2	Umsetzung von Server-Side Rendering	513
16.2.1	Initialisierung und Konfiguration der Applikation	514
16.2.2	Die Komponenten der Applikation	516
16.2.3	Die Clientseite der SSR-Applikation	518
16.2.4	Der Serverteil einer SSR-Applikation	519
16.2.5	Starten der Applikation	524
16.2.6	Resultate des Server-Side Renderings	525
16.3	Ausblick: Server-Side Rendering bei Applikationen mit Authentifizierung	526
16.4	Ausblick: Server-Side Rendering und Redux	528
16.5	Zusammenfassung	529

17 Progressive Web Apps 531

17.1 Merkmale einer Progressive Web App	531
17.2 Installierbarkeit	532
17.2.1 Die sichere Auslieferung einer Applikation	533
17.2.2 Das Web App Manifest	536
17.2.3 Service Worker in der React-Applikation	538
17.2.4 Installation der Applikation	538
17.2.5 Den Benutzer fragen	540
17.3 Offlinefähigkeit	543
17.3.1 Integration von Workbox	544
17.3.2 Umgang mit dynamischen Daten	549
17.4 Werkzeuge für die Entwicklung	553
17.5 Zusammenfassung	554

18 Native Apps mit React Native 557

18.1 Der Aufbau von React Native	557
18.2 Installation von React Native	558
18.2.1 Die Projektstruktur	558
18.2.2 Die Applikation starten	559
18.3 Anzeige einer Übersichtsliste	562
18.3.1 Statische Listenansicht	562
18.3.2 Styling in React Native	565
18.3.3 Suchfeld für die »List«-Komponente	570
18.3.4 Serverkommunikation	572
18.4 Debugging in der simulierten React-Native-Umgebung	572
18.5 Bearbeiten von Datensätzen	575
18.5.1 Implementierung der »Form«-Komponente	577
18.6 Publizieren	581
18.6.1 Build der App	582
18.6.2 Upload der gebauten App	583
18.7 Zusammenfassung	583