

Materialien zum Buch

Auf der Webseite zu diesem Buch stehen folgende Materialien für Sie zum Download bereit:

- **alle Beispielprogramme**

Gehen Sie auf www.rheinwerk-verlag.de/4848. Klicken Sie auf den Reiter **MATERIALIEN ZUM BUCH**. Sie sehen die herunterladbaren Dateien samt einer Kurzbeschreibung des Dateiinhalts. Klicken Sie auf den Button **HERUNTERLADEN**, um den Download zu starten. Je nach Größe der Datei (und Ihrer Internetverbindung) kann es einige Zeit dauern, bis der Download abgeschlossen ist.

Geleitwort des Fachgutachters

Für Webentwickler ist die Entscheidung, welche Bibliotheken und Frameworks sie verwenden sollten, dieser Tage nicht leicht. Nahezu täglich kommt ein neues Tool heraus, das »es besser macht als andere«. Da fällt es mitunter schwer, die Übersicht zu behalten und die richtige Auswahl zu treffen, denn oft ist das »gehypte« Tool von heute schon morgen wieder »out«. Bei komponentenorientierten Webframeworks haben sich in den letzten Jahren zum Glück nur einige wenige Frameworks durchgesetzt, sodass hier die Auswahl vermeintlich leichter fällt, wenn auch nicht ganz einfach: Soll man nun auf React setzen oder auf Angular, oder doch lieber auf das neuere Vue.js, von dem jeder gerade spricht?

Als technischer Leiter werde ich häufig mit Fragen dieser Art konfrontiert. Auf welches der Frameworks soll man langfristig setzen? Bei welchem Framework lohnt es sich, sich intensiv damit auseinanderzusetzen? Klar: Über die Vorteile und Nachteile der genannten Frameworks lässt sich streiten, und es gibt nicht die eine allgemeingültige Antwort. Letztlich ist die Entscheidung für oder gegen ein Framework auch abhängig vom jeweiligen Projekt, von der persönlichen Erfahrung und oft auch einfach vom persönlichen Geschmack.

Das von Facebook entwickelte Framework React stellt derzeit insofern meine persönliche Präferenz dar, als es nicht nur außerordentlich performant ist, sondern auch bezüglich der dahinterstehenden Konzepte sehr durchdacht wirkt und sich – gegenüber anderen großen Frameworks – in einem äußerst attraktiven Kosmos bewegt. Sei es der alternative Package Manager Yarn, der ebenfalls von Facebook entwickelt wird und von dem sich auch der offizielle Node.js Package Manager NPM schon bezüglich des ein oder anderen Features inspirieren hat lassen. Oder das Testing-Tool Jest (mit dem sich natürlich nicht nur React-Anwendungen testen lassen), das sich dank Snapshot-Testing besonders für das Testen von React-Anwendungen eignet. Auch GraphQL, die flexiblere Alternative zu REST, stammt von Facebook und fügt sich nahtlos in React-Anwendungen ein.

Keine Frage: Auch bei React gibt es eine gewisse Einstiegshürde. Konzepte wie Redux, JSX und Hooks müssen verstanden werden, bevor Sie sie effektiv einsetzen können. Und genau hierbei hilft Ihnen das vorliegende Buch von Sebastian Springer. Hier lernen Sie, wie Sie React-Applikationen am besten aufsetzen, organisieren und planen, wie Sie Komponenten strukturieren und welche Best Practices Sie dabei beachten müssen. Besonders hervorzuheben ist, dass Sebastian es schafft, die Themen praxisnah anhand eines durchgängigen Beispiels zu erklären. Mit diesem durchgehenden Beispiel hat er sogar mich überzeugt, der ich eigentlich kein Fan von solchen Komplettbeispielen bin. Insofern: Chapeau, Sebastian! Und Ihnen, liebe Leserinnen und Leser, viel Spaß mit dem Buch und beim Lernen von React.

Philip Ackermann

Chief Technology Officer bei Cedalo AG

Vorwort

React steht für mich für eine schnelle und leichtgewichtige, aber trotzdem professionelle Webentwicklung. Die Bibliothek lässt Ihnen bei der Entwicklung viele Freiheiten, was den Aufbau und die Gestaltung einer Applikation angeht. Das ist Fluch und Segen zugleich. Gerade für Einsteiger wird es an dieser Stelle schwierig: Wo soll ich anfangen? Wie strukturiere ich meine Applikation? Wie löse ich konkrete Problemstellungen? Welche Bibliotheken und Hilfsmittel benötige ich für die Entwicklung meiner Applikation? Das sind nur einige Fragen, die man sich zu Beginn auf dem Weg mit React stellt, und genau an dieser Stelle setzt dieses Buch an. Zusammen implementieren wir eine vollständige Applikation, die zahlreiche Problemstellungen aus dem Praxisalltag abdeckt. Dabei lernen Sie nicht nur React selbst, sondern Teile des Ökosystems um die Bibliothek herum kennen.

Egal ob Sie erst in das Thema React einsteigen oder schon Erfahrung damit haben, ich möchte Sie einladen, dieses Buch als Gelegenheit zur aktiven Arbeit mit React zu nutzen. Versuchen Sie die Applikation selbst zu bauen, setzen Sie die verschiedenen Anforderungen um, und lassen Sie sich von den Codebeispielen und Lösungsansätzen für eigene Lösungen inspirieren. Es gibt kaum eine bessere Strategie, sich tiefer in ein Thema einzuarbeiten, als die Technologie oder das Werkzeug selbst zu verwenden, auch einmal einen Fehler zu machen und daraus zu lernen.

Für die Arbeit mit diesem Buch sollten Sie über ein solides Grundwissen in HTML, CSS und JavaScript verfügen. Falls Sie sich an der einen oder anderen Stelle unsicher sind oder sich fragen, was ein bestimmtes Sprachelement genau macht, lege ich Ihnen das Mozilla Developer Network unter <https://developer.mozilla.org/de/> ans Herz. Hierbei handelt es sich um eine umfangreiche aktuelle Referenz für alle Webtechnologien. Arbeiten Sie lieber mit Büchern, kann ich Ihnen an dieser Stelle das JavaScript-Handbuch von Philip Ackermann empfehlen. Ansonsten empfehle ich Ihnen, neugierig zu sein und Dinge auszuprobieren. Fragen Sie sich: Was passiert, wenn ich an dieser Stelle dieses oder jenes tue? Probieren Sie es aus, öffnen Sie die Entwicklerwerkzeuge Ihres Browsers, und sehen Sie sich die Auswirkungen Ihres Experiments an. Der Vorteil der frontendseitigen Webentwicklung ist, dass Sie außer dem Frontend in Ihrer Applikation nichts weiter kaputt machen können, und auch das können Sie durch den Einsatz eines Versionskontrollsystems wie Git auf ein Minimum reduzieren, da Sie immer wieder auf einen funktionierenden Stand zurückwechseln können. Neben diesen Experimenten sollten Sie auch versuchen, die Beispielapplikation eigenständig weiterzuentwickeln oder eine eigene Applikation zu bauen.

Dieses Buch ist sowohl für den Einstieg in React als auch als Nachschlagewerk für den täglichen Gebrauch gedacht. Sie können die Beispiele entweder selbst nachvollziehen, indem Sie den Quellcode selbst schreiben, oder sie laden sich den Code herunter und passen ihn nach Ihren Wünschen an. Für mich ist eigentlich nur wichtig, dass Sie selbst mit React arbeiten, die Bibliothek und ihre Möglichkeiten kennenlernen und viel Spaß dabei haben.

Eine der häufigsten Fragen im Zusammenhang mit JavaScript-Bibliotheken und -Frameworks ist, welche/welches das beste ist. Natürlich ist React eine gute Wahl, wenn es um die Implementierung eines Web-Frontends geht. Allerdings sind andere Lösungen wie Angular oder Vue deshalb nicht schlechter. Versuchen Sie, sich an dieser Stelle selbst ein Bild zu machen,

geben Sie den verschiedenen Ansätzen eine Chance. Für mich hat sich React in der Praxis in mehreren kleinen, aber auch großen Projekten bewährt.

Aufbau des Buchs

Das Buch besteht aus zwei Teilen, die sich in insgesamt 18 Kapitel unterteilen. Der erste Teil des Buchs beschäftigt sich mit React selbst, der zweite Teil beleuchtet das Ökosystem der Bibliothek mit verschiedenen Problemstellungen aus dem Projektalltag.

Der erste Teil beginnt mit einer Einleitung in React, die Ihnen das Grundwissen und die wichtigsten Begriffe und Konzepte erläutert ([Kapitel 1](#)), und einer Erklärung, wie Sie React installieren und verwenden können ([Kapitel 2](#)). Anschließend lernen Sie in [Kapitel 3](#) und [Kapitel 4](#) die verschiedenen Arten von React-Komponenten kennen. Sie erfahren, wie Sie den Komponentenbaum Ihrer Applikation aufbauen und wie die Daten in diesem Baum fließen. Weiterführende Konzepte wie Higher-Order Components und Render Props ergänzen dieses Grundlagenwissen. [Kapitel 5](#) widmet sich der Hooks-API, einer neueren Erweiterung von React, die die Art und Weise, wie eine Applikation aufgebaut wird, entscheidend beeinflusst. Mit TypeScript lernen Sie in [Kapitel 6](#) ein Werkzeug kennen, das Sie bei der Umsetzung unterstützt und Ihnen zusätzliche Sicherheit im Entwicklungsprozess bietet.

React unterstützt verschiedene Ansätze, wenn es um das Styling von Komponenten geht. [Kapitel 7](#) stellt Ihnen einige dieser Ansätze vor und zeigt Ihnen, wie Sie sie in Ihre Applikation integrieren können. Ein weiteres Hilfsmittel im Zusammenhang mit der Qualitätssicherung einer Applikation behandelt [Kapitel 8](#), in dem es um das Formulieren von Unittests geht. [Kapitel 9](#) schließt den ersten Teil des Buchs mit einem Blick auf Formulare ab. Mithilfe von Formularen geben Sie Ihren Benutzern die Möglichkeit, aktiv mit der Applikation zu interagieren und Daten zu produzieren.

Der zweite Teil des Buchs beginnt mit einem Kapitel über die Integration externer Komponentenbibliotheken ([Kapitel 10](#)). Am Beispiel von Material-UI sehen Sie, wie Sie existierende Komponenten in Ihre Applikation integrieren können. In [Kapitel 11](#) sehen Sie, wie Sie mithilfe des React-Routers innerhalb Ihrer Single-Page-Applikation navigieren und damit unterschiedliche Teilbäume Ihrer Applikation rendern können. [Kapitel 12](#) und [Kapitel 13](#) beschäftigen sich mit dem zentralen State-Management in einer Applikation mit der Bibliothek Redux und mit der Behandlung von Seiteneffekten mit verschiedenen asynchronen Middleware-Implementierungen wie beispielsweise Redux-Thunk.

In [Kapitel 14](#) lernen Sie, wie Sie in Ihrer React-Applikation eine GraphQL-Schnittstelle ansprechen können. Diese Abfragesprache stellt eine flexible Alternative zu den herkömmlichen RESTful-Schnittstellen dar, die üblicherweise in der Webentwicklung zum Einsatz kommen. [Kapitel 15](#) bindet mit `react-intl` eine weitere externe Bibliothek in die Applikation ein, um Internationalisierung zu unterstützen. Hier erfahren Sie neben der reinen Übersetzung von Zeichenketten auch mehr über den Umgang mit Zahlen und Datumswerten.

Die letzten drei Kapitel beschäftigen sich mit unterschiedlichen Umgebungen, die von einer React-Applikation berührt werden. Den Beginn macht [Kapitel 16](#) mit einer Einführung in das serverseitige Rendering, um die Performance einer Applikation zu verbessern. Anschließend erweitern Sie Ihre Applikation um einige Features, sodass sie zu einer Progressive Web App wird, die sich auf dem System eines Benutzers installieren und auf möglichst vielen Systemen

ausführen lässt ([Kapitel 17](#)). Die Idee von React auf unterschiedlichen Systemen wird im letzten Kapitel, [Kapitel 18](#), mit React Native noch einen Schritt weitergeführt, indem Sie erfahren, wie Sie mit dieser Bibliothek native Apps mit React umsetzen können.

Download der Codebeispiele

Sämtliche Codebeispiele dieses Buches sind auf der Website des Verlages unter www.rheinwerk-verlag.de/4848 zum Download verfügbar.

Sollten Sie Probleme bei der Umsetzung haben, oder sollte ich trotz sorgfältiger Kontrolle einen Fehler übersehen haben, können Sie mich gerne unter react@sebastian-springer.com kontaktieren.

Danksagung

Ich möchte mich bei allen Personen bedanken, die mich beim Schreiben dieses Buchs unterstützt haben. Allen voran bei Philip, der, wie schon bei meinem Node.js-Buch, die Begutachtung übernommen und viele Anmerkungen und Tipps beigesteuert hat.

Außerdem danke ich Petra Biedermann für den sprachlichen Feinschliff.

Auch dem ganzen Team vom Rheinwerk Verlag möchte ich danken und hier vor allem Anne Scheibe und Stephan Mattescheck für die Betreuung.

Schließlich möchte ich auch noch meiner Frau Alexandra herzlichen Dank für ihre Geduld und Unterstützung sagen.

Sebastian Springer

Aßling