O'REILLY®

Übersetzung der 2. US-Auflage

Laravel

Die umfassende Einführung

Das Framework für moderne PHP-Entwicklung

Matt Stauffer

Übersetzung von Jens Olaf Koch

ÜBERSETZUNG DER 2. AMERIKANISCHEN AUFLAGE

Laravel
Die umfassende Einführung

Das Framework für moderne PHP-Entwicklung

Matt Stauffer

Deutsche Übersetzung von
Jens Olaf Koch

O'REILLY®

Matt Stauffer

Lektorat: Ariane Hesse
Übersetzung: Jens Olaf Koch
Fachlicher Review: Mitarbeiter von mindtwo, *www.mindtwo.de*
Korrektorat: Claudia Lötschert, *www.richtiger-text.de*
Satz: III-satz, *www.drei-satz.de*
Herstellung: Stefanie Weidner
Umschlaggestaltung: Michael Oreal, *www.oreal.de*
Druck und Bindung: mediaprint solutions GmbH, 33100 Paderborn

Bibliografische Information der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über *http://dnb.d-nb.de* abrufbar.

ISBN:
Print 978-3-96009-129-5
PDF 978-3-96010-374-5
ePub 978-3-96010-372-1
mobi 978-3-96010-373-8

1. Auflage 2020

Wieblinger Weg 17
69123 Heidelberg

Authorized German translation of the English edition of titled *Laravel: Up & Running*, 2E
ISBN 9781492041214 © 2019 Matt Stauffer.

Dieses Buch erscheint in Kooperation mit O'Reilly Media, Inc. unter dem Imprint »O'REILLY«. O'REILLY ist ein Markenzeichen und eine eingetragene Marke von O'Reilly Media, Inc. und wird mit Einwilligung des Eigentümers verwendet.

Hinweis:
Dieses Buch wurde auf PEFC-zertifiziertem Papier aus nachhaltiger Waldwirtschaft gedruckt. Der Umwelt zuliebe verzichten wir zusätzlich auf die Einschweißfolie.

Schreiben Sie uns:
Falls Sie Anregungen, Wünsche und Kommentare haben, lassen Sie es uns wissen: kommentar@oreilly.de.

5 4 3 2 1 0

Dieses Buch ist meiner Familie gewidmet.
Mia, meiner kleinen Prinzessin, diesem Bündel voller Freude und Energie.
Malachi, meinem kleinen Prinzen, Abenteurer und Empathen.
Tereva, meiner Inspiration, Ermutigerin, Upgraderin, Motivatorin, Rippe.

Inhalt

Vorwort

Der Weg, der mich zu Laravel geführt hat, ist ganz typisch: Ich hatte jahrelang in PHP programmiert, war aber bereits dabei, mich davon zu verabschieden und mich den Möglichkeiten von Rails und anderen modernen Web-Frameworks zu widmen. Insbesondere Rails besaß eine lebendige Community und bot eine perfekte Kombination aus meinungsstarken Vorgaben und Flexibilität sowie das Potenzial von Ruby Gems, vorgefertige Packages mit Programmen und Bibliotheken zu nutzen.

Aber etwas hielt mich noch davon ab, PHP endgültig hinter mir zu lassen, und ich war froh darüber, als ich auf Laravel stieß. Laravel bot alles, was mich an Rails anzog, aber es war mehr als bloß ein Rails-Klon. Es war ein innovatives Framework mit unglaublich guter Dokumentation, einer einladenden Community und deutlichen Einflüssen verschiedener Sprachen und Frameworks.

Seitdem habe ich in Blogs, Podcasts und Vorträgen auf Konferenzen geteilt, was ich auf meiner Reise mit Laravel gelernt habe; ich habe Dutzende von Laravel-Apps für Arbeits- und Nebenprojekte geschrieben; und ich habe Tausende von Laravel-Entwicklern online und persönlich getroffen. Es gibt viele Tools in meinem Entwicklungs-Werkzeugkasten, aber ich bin ehrlich gesagt am glücklichsten, wenn ich mich vor eine Kommandozeile setzen und `laravel new` *`projektName`* eingeben kann.

Worum es in diesem Buch geht

Dies ist nicht das erste Buch über Laravel, und es wird nicht das letzte sein. Es soll kein Buch sein, das jede Zeile Code oder jedes mögliche Implementierungsmuster abdeckt. Und es soll kein Buch sein, das automatisch veraltet ist, sobald eine neue Version von Laravel veröffentlicht wird. Stattdessen geht es in erster Linie darum, Entwicklern einen wirklichen Überblick und konkrete Beispiele zu geben, damit sie wissen, was sie benötigen, um in allen Laravel-Releases mit jedem einzelnen Feature und Subsystem arbeiten zu können. Anstatt einfach die Dokumentation nachzuerzählen, möchte ich Ihnen helfen, die grundlegenden Konzepte zu verstehen, auf denen Laravel aufbaut.

Laravel ist ein leistungsfähiges und flexibles PHP-Framework. Es gibt eine florierende Community und ein umfassendes Ökosystem an Werkzeugen, was dazu beiträgt, dass Laravels Attraktivität und Reichweite ständig wachsen. Dieses Buch richtet sich an Entwickler, die bereits wissen, wie man Websites und Anwendungen erstellt, und die jetzt lernen wollen, wie man genau das erfolgreich auch mit Laravel macht.

Die Laravel-Dokumentation ist umfassend und ausgezeichnet. Wenn Sie feststellen, dass ich ein bestimmtes Thema Ihrem Geschmack zufolge nicht gründlich genug abdecke, empfehle ich Ihnen, die Onlinedokumentation (*https://laravel.com/docs*) zu besuchen und tiefer in dieses spezielle Thema einzutauchen.

In diesem Buch erwartet Sie eine angenehme Balance zwischen einer Einführung auf hohem Niveau und konkreter Anwendung, und am Ende werden Sie sich hoffentlich dabei wohlfühlen, wenn Sie eine komplette Anwendung in Laravel von Grund auf neu schreiben. Und wenn ich meinen Job gut gemacht habe, werden Sie auch heiß darauf sein.

Für wen dieses Buch gedacht ist

Dieses Buch setzt Kenntnisse grundlegender objektorientierter Programmierpraktiken, PHP (oder zumindest der allgemeinen Syntax der Sprachen der C-Familie), grundlegender Konzepte des Entwurfsmusters »Model-View-Controller« (MVC, auf Deutsch »Modell-Präsentation-Steuerung«) und der Arbeit mit Template Engines voraus. Wenn Sie zuvor noch nie selbst eine Website entworfen haben, wird es vielleicht etwas zu anspruchsvoll sein. Aber solange Sie über Programmiererfahrung verfügen, müssen Sie vor der Lektüre dieses Buchs kein Vorwissen zu Laravel haben – wir decken alles ab, was Sie wissen müssen, angefangen beim einfachsten »Hallo, Welt!«.

Laravel kann mit jedem Betriebssystem eingesetzt werden, aber es wird einige Kommandozeilen-Befehle im Buch geben, die am einfachsten unter Linux/macOS ausgeführt werden können. Windows-Benutzer werden es mit diesen Befehlen und mit moderner PHP-Entwicklung möglicherweise etwas schwerer haben, aber wenn Sie Homestead (eine virtuelle Linux-Maschine) zum Laufen bringen, können Sie alle Befehle von dort aus ausführen.

Wie dieses Buch aufgebaut ist

Dieses Buch ist in gewisser Weise chronologisch gegliedert: Wenn Sie Ihre erste Webanwendung mit Laravel erstellen, decken die ersten Kapitel die grundlegenden Komponenten ab, die Sie zum Einstieg benötigen, während die späteren Kapitel weniger grundlegende bzw. etwas ausgefallenere Funktionen behandeln.

Jeder Abschnitt des Buchs kann für sich allein gelesen werden, aber für alle, die dieses PHP-Framework zum ersten Mal einsetzen, habe ich versucht, die Kapitel so zu strukturieren, dass es sinnvoll ist, alles in der vorgegebenen Reihenfolge zu lesen.

Wenn es passt, schließt ein Kapitel mit zwei speziellen Abschnitten: »Testen« und »TL;DR.« Falls Sie es nicht kennen: »TL;DR« steht für »too long; didn't read«, also »zu lang, nicht gelesen«. Diese letzten Abschnitte zeigen, wie Sie Tests für die im Kapitel behandelten Funktionen schreiben können, und fassen sehr kompakt zusammen, um was es im betreffenden Kapitel geht.

Dieses Buch ist für Laravel 6 geschrieben, deckt aber in der Regel Funktionen und Syntaxänderungen von Laravel 5.1 an ab – auch um die permanente Weiterentwicklung des Frameworks lebendig darzustellen.

Über die zweite Ausgabe

Die erste Ausgabe von *Laravel: Up & Running* erschien im November 2016 und umfasste die Laravel-Versionen 5.1 bis 5.3. Diese zweite Ausgabe deckt die Versionen 5.4 bis 6.6 sowie Laravel Dusk und Horizon ab, zudem ist ein 18. Kapitel über Community-Ressourcen und andere Laravel-Pakete hinzugekommen, die nicht zum Kern des Frameworks gehören und in den ersten 17 Kapiteln nicht behandelt wurden.

Vorbemerkung zur deutschen Ausgabe

Laravel gibt es nur in einer englischsprachigen Version, und das gilt auch für nahezu alle zusätzlichen Module, Programmpakete und Dienstleistungsangebote des gesamten Laravel-Ökosystems.

Wir haben deshalb bei der Übersetzung dieses Buchs versucht, eine angenehme Balance zwischen englischen Originalbegriffen und deutschen Fachausdrücken zu erreichen. Zumal es für viele der in Programmierung und Anwendungsentwicklung benutzten Begrifflichkeiten gar keine deutsche Entsprechung gibt. Deshalb benutzen wir mal die englischen, mal die deutschen Begriffe, mit einer Tendenz zum Englischen. Da Sie sich als angehender Laravel-Entwickler überwiegend im englischsprachigen Umfeld bewegen werden, wäre es eher kontraproduktiv, würden wir zu viele Begriffe oder Teile von Programmlistings eindeutschen.

Dazu kommt, dass Laravel von sich aus einige Funktionen mitbringt, die selbstständig Pluralformen (zum Beispiel von Modell-Namen) erstellen. Dabei wird standardmäßig ein »s« benutzt und angehängt. Obwohl man dieses Verhalten individuell übersteuern kann, liest sich der Code insgesamt geschmeidiger, und man hat weniger Arbeit, wenn man Laravel seinen »Willen« lässt.

Der Schöpfer von Laravel, Taylor Otwell, gibt sich extrem viel Mühe, den Code so lesbar und so nah an natürlicher Sprache zu halten, wie es nur geht. Und es liest

sich natürlich besser, wenn eine Datenbank-Migration *create_customers_table* heißt und nicht *create_kundens_table* (ohne Übersteuerung des Standardverhaltens) oder *create_kunden_table* (mit angepasster Pluralform).

Konventionen, die in diesem Buch verwendet werden

Die folgenden typografischen Konventionen werden in diesem Buch verwendet:

Kursiv
: Zeigt neue Begriffe, URLs, E-Mail-Adressen, Dateinamen und Dateierweiterungen an.

`Nichtproportional`
: Wird für Programm-Listings verwendet, aber auch innerhalb von Absätzen, um dort auf Programmelemente wie Variablen- oder Funktionsnamen, Datenbanken, Datentypen, Umgebungsvariablen, Anweisungen und Schlüsselwörter zu verweisen.

`Nichtproportional fett`
: Zeigt Befehle oder anderen Text an, der vom Benutzer wortgetreu eingegeben werden muss.

`Nichtproportional kursiv`
: Zeigt Programmcode an, der durch Benutzereingaben oder durch kontextabhängige Werte ersetzt werden soll.

{Kursiv in Klammern}
: Zeigt Dateinamen oder Dateipfade an, die durch Benutzereingaben oder durch kontextabhängige Werte ersetzt werden sollen.

Dieses Element weist auf einen Tipp oder Vorschlag hin.

Dieses Element kennzeichnet einen allgemeinen Hinweis.

Dieses Element weist auf eine Warnung hin.

O'Reilly Online-Lernen

Seit fast 40 Jahren bietet *O'Reilly Media* Technologie- und Business-Training, Wissen und Einsichten, um Unternehmen zum Erfolg zu verhelfen.

Unser einzigartiges Netzwerk von Experten und Innovatoren teilt sein Wissen und seine Expertise in Büchern, Artikeln, Konferenzen und auf unserer Online-Lernplattform. Die Online-Lernplattform von O'Reilly bietet Ihnen On-Demand-Zugriff auf Live-Schulungen, detaillierte Lernpfade, interaktive Codierumgebungen und eine umfangreiche Sammlung von Texten und Videos von O'Reilly und über 200 anderen Verlagen. Für weitere Informationen besuchen Sie bitte *https://www.oreilly.com*.

Englischsprachige Website zu diesem Buch

Es gibt eine englischsprachige Website zu diesem Buch, auf der wir Errata, Beispiele und alle weiteren Informationen aufführen. Sie können diese Seite unter *https://bit.ly/laravel-up-and-running-2e* aufrufen.

Danksagung für die erste Ausgabe

Dieses Buch wäre ohne die verständnisvolle Unterstützung meiner wunderbaren Frau Tereva und meines Sohnes Malachi (»Papa schreibt, Buddy!«) nicht möglich gewesen. Und obwohl sie es selbst nicht wirklich bewusst miterlebt hat, war meine Tochter Mia fast über die gesamte Zeit der Entstehung des Buchs dabei, sodass dieses Buch der ganzen Familie gewidmet ist. Es gab viele lange Abendstunden und Arbeitsbesuche bei Starbucks am Wochenende, sodass ich oft nicht bei meiner Familie war, und ich könnte nicht dankbarer sein für ihre Unterstützung und vor allem für ihr Da-Sein, das mein Leben einfach bereichert.

Darüber hinaus hat mich die gesamte Tighten-Familie während des Schreibens des Buchs unterstützt und ermutigt, und mehrere meiner Kollegen haben Codebeispiele bearbeitet (Keith Damiani, »Editor Extraordinaire«) und mir bei schwierigen Projekten geholfen (Adam Wathan, König der Collection-Pipeline). Dan Sheetz, mein Partner bei Tighten, war so freundlich, mir viele Stunden nachzusehen, die ich mit der Arbeit an diesem Buch verbracht habe, und war dabei immer eine Stütze und Ermutigung; und Dave Hicking, unser Betriebsleiter, half mir, meine Zeiteinteilung und Verantwortlichkeiten rund um meine Schreibzeiten zu organisieren.

Taylor Otwell verdient Dank und Anerkennung als Schöpfer von Laravel – und damit auch für die Schaffung vieler Jobs und dafür, dass so viele Entwickler ihr Leben noch mehr lieben. Er verdient Anerkennung dafür, wie sehr er sich auf die Zufriedenheit der Entwickler konzentriert und wie hart er gearbeitet hat, um mit

Empathie für uns Entwickler eine positive und ermutigende Community aufzubauen. Aber ich möchte ihm auch dafür danken, dass er ein liebenswürdiger, unterstützender und anregender Freund ist. Taylor, du bist ein Held.

Vielen Dank an Jeffrey Way, der einer der besten Lehrer im Internet ist. Er hat mich mit Laravel bekannt gemacht und bringt Laravel heute noch jeden Tag vielen Menschen näher. Er ist auch, wenig überraschend, ein fantastischer Mensch, den ich gerne Freund nenne.

Vielen Dank an Jess D'Amico, Shawn McCool, Ian Landsman und Taylor, die mich schon früh als Konferenzsprecher geschätzt haben und mir eine Plattform bieten, von der aus ich wirken kann. Vielen Dank an Dayle Rees, der in den frühen Tagen von Laravel so vielen Menschen dabei geholfen hat, es zu erlernen.

Vielen Dank an alle Menschen, die ihre Zeit und Mühe dem Schreiben von Blog-Posts über Laravel gewidmet haben, besonders in den Kindertagen von Laravel: Eric Barnes, Chris Fidao, Matt Machuga, Jason Lewis, Ryan Tablada, Dries Vints, Maks Surguy und viele mehr.

Und vielen Dank an die gesamte Community von Freunden auf Twitter, IRC und Slack, die über die Jahre mit mir kommuniziert haben. Ich wünschte, ich könnte jeden Namen nennen, aber ich würde einige vergessen und mich dann dafür schämen. Ihr seid alle brillant, und ich fühle mich geehrt, regelmäßig mit euch in Kontakt zu stehen.

Vielen Dank an meine O'Reilly-Lektorin, Ally MacDonald, und alle meine technischen Redakteure: Keith Damiani, Michael Dyrynda, Adam Fairholm und Myles Hyson.

Und natürlich danke an den Rest meiner Familie und meine Freunde, die mich direkt oder indirekt unterstützt haben – meine Eltern und Geschwister, die Gainesville-Community, Geschäftspartner und Autoren, andere Konferenzredner und die einzigartige DCB. Ich muss aufhören, weiter zu schreiben, bevor ich noch meinen Starbucks-Baristas danke.

Danksagung für die zweite Ausgabe

Die zweite Ausgabe ist der ersten sehr ähnlich, sodass alle vorherigen Danksagungen noch gültig sind. Aber ich habe diesmal Hilfe von ein paar weiteren Personen bekommen. Meine technischen Korrekturleser waren Tate Peñaranda, Andy Swick, Mohamed Said und Samantha Geitz, und meine neue O'Reilly-Lektorin war Alicia Young, die mich im letzten Jahr über viele Veränderungen in meinem Leben und innerhalb der Laravel-Community hinweg bei der Arbeit gehalten hat. Matt Hacker vom Atlas-Team beantwortete alle meine dummen AsciiDoc-Formatierungsfragen, auch zur überraschend schwierigen Formatierung der `__()`-Methode.

Und ohne die Hilfe meines Assistenten Wilbur Powery hätte ich es nicht geschafft, diese zweite Ausgabe zu schreiben. Wilbur war bereit, die Changelogs,

Pull-Requests und Ankündigungen der letzten Jahre zu durchsuchen und jedes Feature der aktuellen Struktur des Buchs zuzuordnen, und testete tatsächlich jedes einzelne Codebeispiel, damit ich meine begrenzte Zeit und Energie auf das Schreiben der neuen und aktualisierten Abschnitte konzentrieren konnte.

Außerdem hat meine Tochter Mia jetzt den Weg aus dem Bauch ihrer Mutter gefunden. Also lassen Sie mich einfach noch ihre Freude und Energie und Liebe und Niedlichkeit und ihren Abenteuergeist der Liste meiner Inspirationsquellen hinzufügen.

KAPITEL 1

Warum Laravel?

In den frühen Tagen des dynamischen World Wide Web sah das Schreiben einer Anwendung ganz anders aus als heute. Die Entwickler waren damals nicht nur dafür verantwortlich, den Code für die jeweils spezielle Anwendungslogik zu schreiben, sondern auch für all jene Komponenten, die immer wieder für Websites benötigt werden: für die Authentifizierung von Benutzern, die Validierung von Eingaben, Datenbankzugriffe, die Erstellung von Vorlagen und vieles mehr.

Heutzutage gibt es Dutzende von Frameworks für die Anwendungsentwicklung und Tausende von Komponenten und Bibliotheken, die allen Programmierern leicht zugänglich sind. Es ist gängige Rede unter Entwicklern, dass, kaum habe man ein Framework gelernt, es bereits drei neuere (und angeblich bessere) Frameworks gebe, die es gerne ersetzen möchten.

»Weil er halt existiert«, mag eine sinnvolle Rechtfertigung dafür sein, einen Berg zu besteigen. Aber es gibt bessere Gründe als »Weil es halt existiert«, um sich für ein bestimmtes Framework zu entscheiden – oder überhaupt eines zu verwenden. Stellen wir uns also die berechtigte Frage: Wozu sind Frameworks eigentlich gut? Genauer gesagt: Wozu ist Laravel gut?

Warum ein Framework verwenden?

Es ist leicht nachzuvollziehen, warum es hilfreich ist, die einzelnen Komponenten und Pakete zu verwenden, die es für PHP-Entwickler gibt. Bei Packages ist jemand anderes für die Entwicklung und Wartung eines isolierten Stücks Programmcode verantwortlich, mit dem eine bestimmte Aufgabe gelöst wird, und theoretisch sollte diese Person ein tieferes Verständnis für diese einzelne Komponente besitzen, als Sie sich selbst in angemessener Zeit aneignen können.

Frameworks wie Laravel – und Symfony, Lumen und Slim – versammeln eine Reihe Komponenten von Drittanbietern und bündeln diese mit Framework-eigenem »Klebstoff« wie Konfigurationsdateien, Service Providern, vordefinierten Verzeichnisstrukturen und Anwendungs-Bootstraps. Ganz allgemein besteht der Vorteil, ein Framework zu verwenden, darin, dass jemand für Sie nicht nur bereits über einzelne

Komponenten entschieden hat, sondern auch darüber, wie diese Komponenten zusammenarbeiten sollen.

»Ich baue es einfach selbst«

Nehmen wir an, Sie beginnen mit einer neuen Webapplikation, ohne die Vorteile eines Frameworks zu nutzen. Womit fangen Sie an? Ziemlich sicher müssen HTTP-Requests geroutet werden, sodass Sie sich alle verfügbaren HTTP-Request- und Response-Bibliotheken anschauen und danach eine auswählen müssen. Dann brauchen Sie einen Router. Und wahrscheinlich benötigen Sie auch irgendeine Art Konfigurationsdatei für die Routen. Welche Syntax soll dabei verwendet werden? An welcher Stelle soll die Konfigurationsdatei abgelegt werden? Und was ist mit Controllern? Wo sollen die liegen und wie sollen sie geladen werden? Wahrscheinlich benötigen Sie einen Dependency Injection Container, um die Controller und ihre Abhängigkeiten aufzulösen. Aber welchen?

Wenn Sie sich die Zeit nehmen, all diese Fragen zu beantworten und Ihre Anwendung erfolgreich zu erstellen, was bedeutet das dann für einen möglichen späteren Entwickler? Und was machen Sie, wenn Sie beispielsweise vier dieser Anwendungen mit solch einem Framework Marke Eigenbau programmiert haben – oder gar ein ganzes Dutzend – und Sie sich jeweils daran erinnern müssen, wo sich die Controller befinden oder wie genau die Routing-Syntax lautet?

Konsistenz und Flexibilität

Frameworks lösen dieses Problem, indem sie eine sorgfältig durchdachte Antwort auf die Frage »Welche Komponente sollen wir verwenden?« geben und zudem sicherstellen, dass die jeweils gewählten Komponenten auch gut zusammenarbeiten. Darüber hinaus bieten Frameworks feste Konventionen, die die Menge an Code reduzieren, die ein Entwickler, der neu zu einem Projekt stößt, verstehen muss: Sobald Sie begriffen haben, wie Routing in *einem* Laravel-Projekt funktioniert, wissen Sie sofort, wie es in *allen* Laravel-Projekten funktioniert.

Wenn jemand vorschreibt, dass sein eigenes Framework für jedes neue Projekt benutzt werden muss, dann geht es letztlich darum, zu *kontrollieren*, was in die Grundlage seiner Applikation einfließt und was nicht. Das bedeutet, dass Ihnen die besten Frameworks nicht nur ein solides Fundament bieten, sondern auch die Freiheit geben, praktisch alles Ihren eigenen Wünschen entsprechend anzupassen. Und genau das ist es, was Laravel so besonders macht und was ich Ihnen im weiteren Verlauf dieses Buchs zeigen möchte.

Eine kurze Geschichte der Web- und PHP-Frameworks

Will man die Frage »Warum Laravel?« beantworten, muss man seine Geschichte kennen – und seine Vorläufer. Bevor Laravel populär wurde, gab es bereits eine