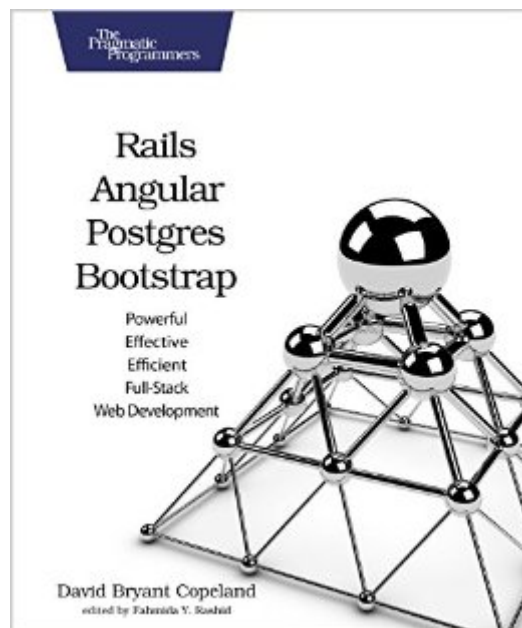## The book was found

Rails, Angular, Postgres, And Bootstrap: Powerful, Effective, And Efficient Full-Stack Web Development

Synopsis

As a Rails developer, you care about user experience and performance, but you also want simple and maintainable code. Achieve all that by embracing the full stack of web development, from styling with Bootstrap, building an interactive user interface with AngularJS, to storing data quickly and reliably in PostgreSQL. Take a holistic view of full-stack development to create usable, high-performing applications, and learn to use these technologies effectively in a Ruby on Rails environment. Rails is a great tool for building web applications, but it's not the best at everything. Embrace the features built into your database. Learn how to use front-end frameworks. Seize the power of the application stack through AngularJS, Bootstrap, and PostgreSQL. When used together, these powerful and easy-to-use tools will open you to a new world of possibilities. In each chapter of this book, you'll learn how these technologies can work together inside a Rails app to deliver well-performing applications that offer great user experiences.Create a usable and attractive login form using Bootstrap's styles, while ensuring the database table backing it is secure, using Postgres' check constraints. You'll see how creating an advanced Postgres index for a case-insensitive search can speed up your back-end--allowing you to create a highly dynamic user experience using AngularJS. You'll create complex interfaces using Bootstrap's grid, together with Angular's form support, backed by a materialized view for caching within Postgres. You'll get your front-end working with the Asset Pipeline, use Postgres' features from migrations, and write unit tests for all of it. All of this within Rails.You'll gain the confidence to work at every level of the application stack, bringing the right solution to every problem.What You Need:You'll need Postgres 9.4, Rails 4.2, and Ruby 2.2. You'll learn how to install Postgres on your computer or use a free version of it in the cloud. You should have some experience with basic Rails concepts and a cursory understanding of JavaScript, CSS, and SQL, but by no means need to be an expert.

Book Information

Paperback: 240 pages

Publisher: Pragmatic Bookshelf; 1 edition (February 5, 2016)

Language: English

ISBN-10: 1680501267

ISBN-13: 978-1680501261

Product Dimensions: 7.5 x 0.6 x 9.2 inches

Shipping Weight: 12.6 ounces (View shipping rates and policies)

Average Customer Review: 4.5 out of 5 stars See all reviews (20 customer reviews)

Best Sellers Rank: #39,771 in Books (See Top 100 in Books)   #7 inÂ Books > Computers & Technology > Programming > Languages & Tools > Ruby   #32 inÂ Books > Computers & Technology > Programming > Web Programming > JavaScript   #172 inÂ Books > Textbooks > Computer Science > Programming Languages

Customer Reviews

Full-Stack Web Development Tips by Dave Copeland   Embrace Your Database   Most SQL databases are packed with handy features, but Postgres has more than most. Accurately model your domain with sophisticated data types like JSON and Arrays, speed up your queries with complex indexes and materialized views, and keep your data consistent with powerful check constraints.   Use a CSS Framework   Hand-written CSS can be complex and unmanageable fast. A CSS framework can allow you to style you UI more quickly and efficiently by using pre-build classes. An all-inclusive framework like Bootstrap goes even farther by giving you the ability to create detailed components that look and feel greatÃ¢Â Â"quickly, easily, and with no CSS.   Invest in a JavaScript Framework   When your user interface calls for more dynamic, interactive elements, you don't want to start from scratch. Even with JQuery, your front-end code can be messy and untestable. Using a framework like Angular, you can confidently test your code, while easily providing complex features and interactions without making a mess.   Ten Features of Your Stack You're Not Using   Grid-based design with Bootstrap. When you think of a UI as a grid, creating it becomes a lot simpler.   Promises instead of Callbacks in Angular. Write your front-end code imperatively and clearly, without callback soup.   MVC on the client using Angular. Sometimes UI code is complex, and you want it distinct from your templates. Keep your front-end code organized.

View larger   Solve the Problem in the Right Place   With the ability to be effective at all levels of the stack, you can see the entire problem at once, and solve it where it needs it. It could be adding an index to your query takes your view from sluggish to snappy. Or, you may need to use AJAX to grab results one at a time. When you're comfortable up and down the stack, you'll solve it the right way.   Test your front-end code with Teaspoon. By separating your front-end logic from your views, you can unit test it, and be confident it will work.   Materialized Views in Postgres. Automatically cache de-normalized data right in your database. Get further without standing up new caching infrastructure.   Indexes on derived values in Postgres. Create a database index on normalized or modified values. Make your indexes match your queries without

any hacks.   Store, query, and index JSON with Postgres. Don't sacrifice performance if you have to deal with loosely defined data.   Native arrays in Postgres. Sometimes you just need a list of strings without making a lot of new tables.   Headless integration tests with PhantomJS. Run full-blown integration tests quickly from the command line, without launching an actual browser. Bring it all together with Rails. Despite being an opinionated framework, Rails has all the hooks you need to enhance your stack.

I am the architect and implementer of nderground (nderground dot net) a social network built with the Grails framework. The implementation languages of Grails are Groovy and Java. In building nderground I used Postgres (the managed service RDS/Postgres). I have used Bootstrap to build the nderground web pages.When I saw the book Rails, Angular, Postgres and Bootstrap I thought that it would be a chance to learn about how the Ruby on Rails framework can apply some of the components I had used with the Grails framework. I was especially interested in learning about how AngularJS could be used. One of the challenges in building modern web applications is the complexity of the client side code and I was hoping that learn whether AngularJS could be used to manage this complexity.The first thing that the potential reader should know about David Copeland's book is that this is not a book for beginners. If you have never worked with a web application framework like Ruby on Rails, Django or Grails, I think that you're going to have a hard time with this book. The author implicitly assumes that you understand the Model-View-Controller structure that these frameworks use. Similarly, if you don't know anything about JavaScript or Relational Databases and SQL parts of this book may be difficult to understand.David Copeland's book reminded me of a cookbook that I was once given by the famous French Chef Escoffier. As with this book, Escoffier was writing for experienced practitioners. His recipes are outlines of how to build French cuisine, assuming that the reader has already mastered the necessary technique. I have never been a cook on that level, so I ended up giving the book away (Julia Child is more my speed).Rails, Angular, Postgres and Bootstrap will help the reader understand how to build powerful web applications, within the Rails framework. The power of modern web frameworks is that they allow the programmer to leverage the framework to create an application more quickly. David Copeland outlines how each of the components can be used and how they fit together in a web application architecture. He provides code snippets in Ruby, HTML/Bootstrap that show how the elements fit together.The book starts out where most web applications do, with the construction of a login and authentication infrastructure. This uses the Devise Ruby component. As David Copeland notes, building your own authentication can be complicated (I did this for nderground). Leveraging

Devise is a way to do this much more rapidly. Devise provides high quality encryption that resists cracking.A feature of the book that I really liked was the focus on the open source Postgres database. Some developers mindlessly use MySQL because it is the current fad for web databases (and because it is easy to interface MySQL with JavaScript). As Copeland points out, Postgres is a much more powerful, industrial strength database.The book assumes a fair amount of familiarity with Ruby on Rails. For example, David Copeland refers to Rails' Active Record object-relational-mapping framework and seems to assume that the reader is familiar it. Active Record has some excellent documentation on the Active Record Wiki, but I think that it would have been useful to provide an Active Record overview.The book jumps into the analysis and optimization of SQL query performance. However, for web many applications, if the designer chooses the database indices properly, Postgres performance will be very good. Only in the case where the database must handle huge amounts of data (e.g., hundreds of gigabytes or a terabyte or so) or there are complex queries is query analysis necessary. The book does not mention the open source Postgres tool PGAdmin, which can be used for query analysis. The examples in the book use the Postgres command line.In applying Rails, AngularJS and Bootstrap the book necessarily mixes these frameworks in some of the example code. However, at times I had to read the examples carefully to understand what was Bootstrap and what was Angular and how they interacted.If you have some solid Ruby on Rails experience and are well versed in relational databases, especially Postgres, this book will allow you to build more powerful web applications, with less effort. However, even for those who are experienced, careful reading will be necessary, since the book covers a huge amount of material.

Overall this is a decent book. The author is covering a very wide range of topics and he does a good job of providing enough information without getting bogged down. He also does a great job of building to an interface and slowly adding complexity to the examples.I got the paper and Kindle version of the book and found the paper version to be much better (isn't that always the case though?).

For the most part, I'm a .NET developer who does by far more middle and back-end code than anything else. The one thing that is keeping me from declaring myself a full-stack developer is that I've never done much with web front end coding. That means I've never done anything with Angular or Bootstrap. Also, as someone who lives primarily in the .NET world, I've never messed around with Ruby, Rails, or Postgres. All we need to add now is the desire to learn all of these things and

I'm the ideal audience for this book.I appreciate that there is a short section in the beginning to help you set up your computer properly. I was reading another book recently about a different programming language and I had to wind up scouring all over the web to find the components I needed just to play with it.Each section of the book starts off by showing where in the development stack we will be focusing our attention. As we progress through the book, we gradually build an application the author calls "Shine". The author does a good job of walking you through the development process for this project.If you are looking to get your hands dirty in this set of technologies, I think this is a great way to learn it. A note, however: if you are trying to learn just one piece of it (say, Bootstrap) and then tie that into a different stack, this is not the way to go about it. Sure, you can pick stuff up, but the idea of this book is to give you an idea of how these programming tools work together. If you want to learn more about any one tool, you should get a book dedicated to that topic. Similarly, it should also be noted that the stuff you learn about each tool is very basic. I took this book over to my colleague in the next room over who does primarily web front end stuff all the time and asked him what he thought and after spending several minutes reading through the sections he was familiar with, he pronounced it much too much of a beginner book for him to get anything out of it (at least for those components). Now, in my opinion, that's good in this context--you are learning about a whole bunch of things all at once; you don't want to get into all the "power-user" stuff on your first go-around. Still, it underscores my point--the intended audience is someone who is trying to learn this specific stack and has little to no exposure to most (if not all) of it.If you know at least a little bit about programming and you want to learn this stack, I think this is a great book. If you already know most of this stack OR if you only want to learn about 1 or 2 components of the stack (and incorporate them into something else), you are not the intended audience. Thus, this book has a pretty narrow target audience, but it does a great job for those people.

Rails, Angular, Postgres, and Bootstrap: Powerful, Effective, and Efficient Full-Stack Web Development Ruby on Rails Tutorial: Learn Web Development with Rails (3rd Edition) (Addison-Wesley Professional Ruby) Ruby on Rails Tutorial: Learn Web Development with Rails (4th Edition) (Addison-Wesley Professional Ruby Series) Data Visualization Toolkit: Using JavaScript, Rails, and Postgres to Present Data and Geospatial Information (Addison-Wesley Professional Ruby Series) Learning Rails 5: Rails from the Outside In Ruby on Rails 3 Tutorial: Learn Rails by Example (Addison-Wesley Professional Ruby) Head First Rails: A Learner's Companion to Ruby on Rails The McKinsey Engagement: A Powerful Toolkit For More Efficient and

Effective Team Problem Solving Web Development with Node and Express: Leveraging the JavaScript Stack Angular 2 Development with TypeScript ASP.NET MVC 5 with Bootstrap and Knockout.js: Building Dynamic, Responsive Web Applications Beginning Rails 4 (Expert's Voice in Web Development) Agile Web Development with Rails 4 (Facets of Ruby) High Availability for the LAMP Stack: Eliminate Single Points of Failure and Increase Uptime for Your Linux, Apache, MySQL, and PHP Based Web Applications Pro Web 2.0 Application Development with GWT (Expert's Voice in Web Development) Getting MEAN with Mongo, Express, Angular, and Node High Performance ISAPI/NSAPI Web Programming: Your Complete Guide to Creating Fast, Powerful Web Server Programs Developing ActiveX Web Controls: The Hands-On Guide to Creating Powerful Controls on the Web The Bootstrap VA: The Go-Getter's Guide to Becoming a Virtual Assistant, Getting and Keeping Clients, and More! The DIY Wedding: How to Bootstrap Your Way to the Perfect Wedding

[Dmca](#)