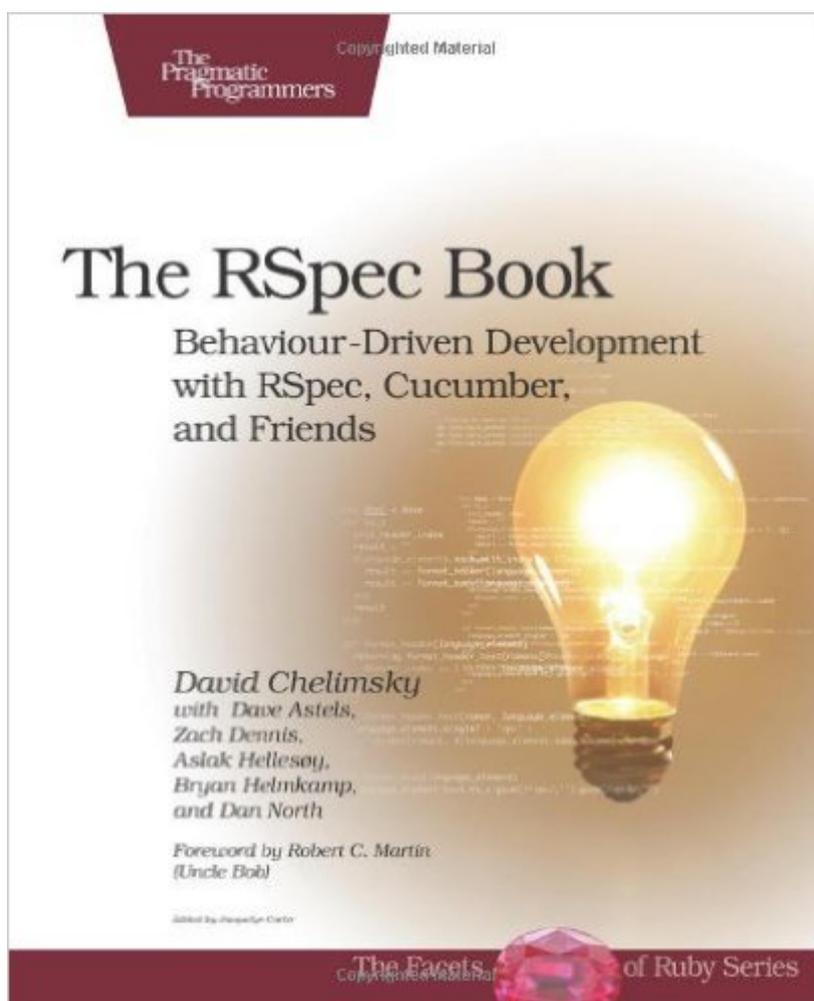# The RSpec Book: Behaviour Driven Development With RSpec, Cucumber, And Friends (Facets Of Ruby)

Synopsis

Behaviour-Driven Development (BDD) gives you the best of Test Driven Development, Domain Driven Design, and Acceptance Test Driven Planning techniques, so you can create better software with self-documenting, executable tests that bring users and developers together with a common language.Get the most out of BDD in Ruby with The RSpec Book, written by the lead developer of RSpec, David Chelimsky.You'll get started right away with RSpec 2 and Cucumber by developing a simple game, using Cucumber to express high-level requirements in language your customer understands, and RSpec to express more granular requirements that focus on the behavior of individual objects in the system. You'll learn how to use test doubles (mocks and stubs) to control the environment and focus the RSpec examples on one object at a time, and how to customize RSpec to "speak" in the language of your domain.You'll develop Rails 3 applications and use companion tools such as Webrat and Selenium to express requirements for web applications both in memory and in the browser. And you'll learn to specify Rails views, controllers, and models, each in complete isolation from the other.Whether you're developing applications, frameworks, or the libraries that power them, The RSpec Book will help you write better code, better tests, and deliver better software to happier users.

Book Information

Series: Facets of Ruby

Paperback: 448 pages

Publisher: Pragmatic Bookshelf; 1 edition (December 25, 2010)

Language: English

ISBN-10: 1934356379

ISBN-13: 978-1934356371

Product Dimensions: 7.5 x 0.9 x 9.2 inches

Shipping Weight: 1.4 pounds (View shipping rates and policies)

Average Customer Review: 4.0 out of 5 stars See all reviews (25 customer reviews)

Best Sellers Rank: #137,056 in Books (See Top 100 in Books)   #33 in Books > Computers & Technology > Programming > Languages & Tools > Ruby   #57 in Books > Computers & Technology > Programming > Software Design, Testing & Engineering > Testing   #110 in Books > Business & Money > Management & Leadership > Quality Control & Management > Quality Control

Customer Reviews

Behavior Driven Development is an important new methodology, and the authors of The RSpec

Book provide a solid introduction to the theory and practice of BDD. However, I feel this book would have been just as good as a series of blog posts. This is not meant to denigrate the book, into which I know much effort was invested. However, the examples in the book fail to tackle the real-world challenges that you will face when implementing BDD.Einstein said "Everything should be kept as simple as possible, but not simpler." A common fault of software courses and books is an avoidance of real-world complexity. For example, in The RSpec Book, the last 3 or 4 chapters are on BDD with Ruby on Rails. The sample application that is developed is ridiculously simple. Also, no cucumber specs are developed for it, so we are basically writing code for its own sake, rather than executing on the BDD mantra of "writing software that matters."Both BDD and Ruby on Rails are meant to offer solutions for large, complex software projects and the ins and outs of their proper usage can only be learned by application to software that goes well beyond toy functionality. In a large Rails projects, with dozens of models with complex associations interacting with multiple gems, managing RSpec examples and Cucumber scenarios is a project in and of itself. The introductory example application "CodeBreaker" is better because it shows the full BDD development cycle with both cucumber and RSPec. Perhaps the authors should have built on that same example in the Ruby on Rails chapters.In the end, if you want to learn BDD, you definitely should buy this book. The authors would do well, however, to bring in more of their real world experience in future editions.One final note for those interested in advice on real-word BDD best practices, google "You're cuking it wrong" by Jonas Nicklas.

I hate to be the first one to give a less than great review of this book, especially since several of the authors are chiefly responsible for these great testing tools to be in existence. But I'd probably be even harder on the book than I am if I knew of a good alternative, which I don't. So best I can tell, this is still the best book to learn BDD using RSpec and Cucumber.There seems to be a fair amount of errata that didn't get fixed prior to going to print. In fact, the book on whole seems like it could have used more in the editing process. I question the organization of the book; however I do get a sense of what the authors were trying to accomplish.I am sympathetic to the challenges of writing a book for technologies that are very rapidly changing; that said, at a conference in June 2010, the author had already switched to using Capybara instead of Webrat, so I was shocked that the book went to print in December 2010 without mention of Capybara, which from what I can tell, seems to be the new de facto standard for browser simulation.No doubt BDD while easy to understand at an abstract level, seems to be an art hard to explain concretely. Surely examples are the best way to learn, and fortunately this book does use plenty of examples. I love that they devote 100 pages

specifically to BDD in Rails (although I'm sure developers using other languages and frameworks don't). I'd say this edition of the book is a good 0.8 release, and I look to the inevitable 1.0 (aka 2nd Edition).

I am a long time Web Developer, ramping back up on RoR. I was looking for a book that would help me develop a good solid testing environment on this new Rails 3 project that I am creating.As I read and tried many of the little tiny examples in the book, and eventually decided that I do not want to do Cucumber (I do not need to spend the extra time to generate code to translate requirements from English, RSpec is clear enough for me). Unfortunately (from my perspective), much of the book rambles on about Cucumber and integrating it with RSpec.As I went through the book and I found a section of code that interested me, it too frequently told me that I would hear more details later on, which I found quite frustrating. I was ready for the down-low, and never seemed to find it, until I eventually jumped to Chapters 23, 24 and 25. Chapters 23, 24 and 25 are the chapters that walk you through the process of developing Test/Behavior driven View, Controllers and Models. This is what I needed to get my project going.This book is worth it, even if you only look at the RSpec chapters.Oh, by the way, when you are looking into the tools you want to use for integration testing, I recommend looking into Capybara, which is not talked about in the book.I hope this helps.Dave Taylor (tayloredwebsites.com)

The title is misleading. This book is not about RSpec in particular, rather this book is about Behavior Driven Development (BDD). It just so happens that the state of the art in BDD happens to be around RSpec, Cucumber, and friends. If you're looking to learn about BDD, then this is probably one of the best references today - don't worry about the Ruby dependencies, many use Cucumber and RSpec to drive testing of their JavaScript code, Java, etc.I found the organization of the book to be a little disorienting: right off the bat we jump into a BDD workflow, and only later in the book do we get introduced to the actual frameworks. As someone who has already spent enough time with the tools prior to picking up this book, I was able to follow along, but I wonder if someone less familiar would not find this organization confusing. In reality, I think this book is a reflection of the very learning process that the authors have gone through themselves while writing the frameworks and the book itself. It feels like this book is really a two in one: philosophy of and for BDD, and a manual for existing tools - and hence the confusing organization.Having said all that, if you are interested in learning about BDD, or improving your existing BDD workflow, then this is definitely the book you are looking for. Along the way you'll also learn about the internals of RSpec, Cucumber, Rails +

BDD, and a variety of other tools.

The RSpec Book: Behaviour Driven Development with RSpec, Cucumber, and Friends (Facets of Ruby) Ruby: Learn Ruby in 24 Hours or Less - A Beginner's Guide To Learning Ruby Programming Now (Ruby, Ruby Programming, Ruby Course) The Cucumber for Java Book: Behaviour-Driven Development for Testers and Developers The Cucumber Book: Behaviour-Driven Development for Testers and Developers (Pragmatic Programmers) Metaprogramming Ruby 2: Program Like the Ruby Pros (Facets of Ruby) Programming Ruby 1.9 & 2.0: The Pragmatic Programmers' Guide (The Facets of Ruby) Agile Web Development with Rails 4 (Facets of Ruby) 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) Effective Ruby: 48 Specific Ways to Write Better Ruby (Effective Software Development Series) Ruby Red (Ruby Red Trilogy Book 1) What Customers Want: Using Outcome-Driven Innovation to Create Breakthrough Products and Services: Using Outcome-Driven Innovation to Create Breakthrough Products and Services RSpec Essentials Ruby's Tea for Two (Max and Ruby) Ruby's Cupcakes (Max and Ruby) Ruby's Rainbow (Max and Ruby) Max & Ruby's Storybook Treasury (Max and Ruby) Ruby's Falling Leaves (Max and Ruby) Ruby Performance Optimization: Why Ruby is Slow, and How to Fix It Ruby and the Booker Boys #1: Brand New School, Brave New Ruby