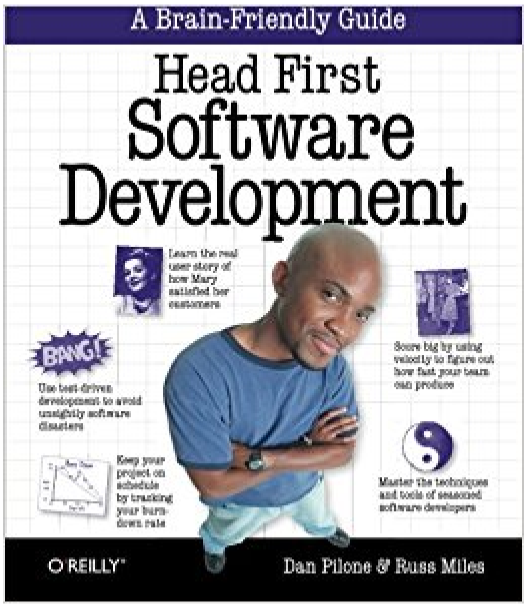
Head First Software Development: A Learner's Companion To Software Development


A Brain-Friendly Guide
Head First
Software
Development
Learn the real user story of how Mary satisfied her customers
BANG!
Use test-driven development to avoid unsightly software disasters
Keep your project on schedule by tracking your burn-down rate
Score big by using velocity to figure out how fast your team can produce
Master the techniques and tools of seasoned software developers
O'REILLY
Dan Pilone & Russ Miles

Synopsis

Even the best developers have seen well-intentioned software projects fail -- often because the customer kept changing requirements, and end users didn’t know how to use the software you developed. Instead of surrendering to these common problems, let Head First Software Development guide you through the best practices of software development. Before you know it, those failed projects will be a thing of the past.With its unique visually rich format, this book pulls together the hard lessons learned by expert software developers over the years. You’ll gain essential information about each step of the software development lifecycle -- requirements, design, coding, testing, implementing, and maintenance -- and understand why and how different development processes work.This book is for you if you are:Tired of your customers assuming you’re psychic. You’ll learn not only how to get good requirements, but how to make sure you’re always building the software that customers want (even when they’re not sure themselves)Wondering when the other 15 programmers you need to get your project done on time are going to show up. You’ll learn how some very simple scheduling and prioritizing will revolutionize your success rate in developing software.Confused about being rational, agile, or a tester. You’ll learn not only about the various development methodologies out there, but how to choose a solution that’s right for your project.Confused because the way you ran your last project worked so well, but failed miserably this time around. You’ll learn how to tackle each project individually, combine lessons you’ve learned on previous projects with cutting-edge development techniques, and end up with great software on every project.Head First Software Development is here to help you learn in a way that your brain likes... and you’ll have a blast along the way. Why pick up hundreds of boring books on the philosophy of this approach or the formal techniques required for that one? Stick with Head First Software Development, and your projects will succeed like never before. Go on, get started... you’ll learn and have fun. We promise.

Book Information

Series: Head First

Paperback: 498 pages

Publisher: O’Reilly Media; 1 edition (January 11, 2008)

Language: English

ISBN-10: 0596527357

ISBN-13: 978-0596527358

Product Dimensions: 8 x 1 x 9.2 inches

Shipping Weight: 2.4 pounds (View shipping rates and policies)

Average Customer Review: 4.1 out of 5 stars 42 customer reviews

Best Sellers Rank: #157,677 in Books (See Top 100 in Books) #28 inÃ Â Books > Business & Money > Management & Leadership > Project Management > PMP Exam #179 inÃ Â Books > Textbooks > Computer Science > Software Design & Engineering #395 inÃ Â Books > Computers & Technology > Programming > Software Design, Testing & Engineering > Software Development

Customer Reviews

About &#039;Head First&#039; Books We think of a Head First Reader as a Learner Learning isn&#039;t something that just happens to you. It&#039;s something you do. You can&#039;t learn without pumping some neurons. Learning means building more mental pathways, bridging connections between new and pre-existing knowledge, recognizing patterns, and turning facts and information into knowledge (and ultimately, wisdom). Based on the latest research in cognitive science, neurobiology, and educational psychology, Head First books get your brain into learning mode. Here&#039;s how we help you do that: We tell stories using casual language, instead of lecturing. We don&#039;t take ourselves too seriously. Which would you pay more attention to: a stimulating dinner party companion, or a lecture? We make it visual. Images are far more memorable than words alone, and make learning much more effective. They also make things more fun. We use attention-grabbing tactics. Learning a new, tough, technical topic doesn&#039;t have to be boring. The graphics are often surprising, oversized, humorous, sarcastic, or edgy. The page layout is dynamic: no two pages are the same, and each one has a mix of text and images. Metacognition: thinking about thinking If you really want to learn, and you want to learn more quickly and more deeply, pay attention to how you pay attention. Think about how you think. The trick is to get your brain to see the new material you&#039;re learning as Really Important. Crucial to your well-being. Otherwise, you&#039;re in for a constant battle, with your brain doing its best to keep the new content from sticking.

Here&#039;s what we do: We use pictures, because your brain is tuned for visuals, not text. As far as your brain&#039;s concerned, a picture really is worth a thousand words. And when text and pictures work together, we embedded the text in the pictures because your brain works more effectively when the text is within the thing the text refers to, as opposed to in a caption or buried in the text somewhere. We use redundancy, saying the same thing in different ways and with different media types, and multiple senses, to increase the chance that the content gets coded into more than one area of your brain.   We use concepts and pictures in unexpected ways because your brain is tuned for novelty, and we use pictures and ideas with at least some emotional content, because your brain is more likely to remember when you feel something.   We use a personalized, conversational style, because your brain is tuned to pay more attention when it believes you&#039;re in a conversation than if it thinks you&#039;re passively listening to a presentation. We include many activities, because your brain is tuned to learn and remember more when you do things than when you read about things. And we make the exercises challenging-yet-do-able, because that&#039;s what most people prefer.   We use multiple learning styles, because you might prefer step-by-step procedures, while someone else wants to understand the big picture first, and someone else just wants to see an example. But regardless of your own learning preference, everyone benefits from seeing the same content represented in multiple ways.   We include content for both sides of your brain, because the more of your brain you engage, the more likely you are to learn and remember, and the longer you can stay focused. Since working one side of the brain often means giving the other side a chance to rest, you can be more productive at learning for a longer period of time.   We include challenges by asking questions that don&#039;t always have a straight answer, because your brain is tuned to learn and remember when it has to work at something. Finally, we use people in our stories, examples, and pictures, because, well, you&#039;re a person. Your brain pays more attention to people than to things.

Dan Pilone is a Senior Software Architect with Blueprint Technologies, Inc. He has designed and implemented systems for Hughes, ARINC, UPS, and the Naval Research Laboratory. He also teaches project management, software design, and software engineering at The Catholic University in Washington D.C. Dan has written several books on software development, including UML 2.0 in a Nutshell (0-596-00795-7) and UML 2.0 Pocket Reference (0-596-10208-9), both O’Reilly.Russell Miles is a senior consultant for SpringSource in the UK where he works with various companies to help them take full advantage of the Spring Framework. To ensure that he has as little spare time as possible, Russ contributes to various open source projects while working on books for O’Reilly.

Awesome book. As a freelance software developer, I was previously using the book "Systems Analysis and Design 5th Edition by Dennis, Wixom, Roth" as a reference for the SDLC, but I find this book to be so much better. I just picked it up two days ago and read the first half in one sitting, though I did skim through parts of it. HFSD is very practical, where as SAD5th packs massive details into the book, but lacks the real life usability. Reading this book, I feel like I can take their base process and apply it to my particular needs, and start using it immediately. I especially like the part about using a Scrum board, creating user stories(features), introduction to Version Control and test suites, as well as the multiple iterations. Systems Analysis and Design 5th is great if you want to find a ton of charts and diagramming methods, many of which are needed. This book is great if you are in particular a software developer who wants a more simple development life cycle.

One of the best books I've read about software development. I love the head first series as they made everything simplified and visual.Highly recommended book.

In order to keep this book compact, the authors made choices about what to explain. This makes the book leaner and friendlier, but don't expect the encyclopedic nature of more traditional software engineering books. I like this book because it sets a relatively modest goal, and achieves that goal splendidly.

The core concept of this book it to tell you how important it is to interact with your customer during the developing process. If the customers are happy, (sometimes they just give you stupid stupid ideas though) , then everyone is happy. DON'T EVEN TRY HARD to make your customer follow your thoughts because they paid you and you make whatever they want. :3

Great book, up to date material!

One of the best books for studying software development. Anything from HeadFirst is excellent!

Pretty good book. I liked the diagrams.

I have come around to the usefulness of the book. I first got the book as a kindle download and I could not read the illustrations because they were too small. However, I am now working through the physical book and I find it very useful. One has to work through all the playful exercises - the authors convey the information through fictional situations where one has to go through all the steps of software development in response to the needs presented by the situation. In this way the authors present the method to be used and then incorporate it into the story of this fictional situation. Included in this approach are use of a a lot of conceptual illustrations along with exercises to be completed on 'index cards' and crossword puzzles, among other things. Being able to read the illustrations (I couldn’t read them on my machine in kindle) is pretty essential to the book. Plus having the physical book allows me to do the exercises which helped me incorporate and memorize the method employed.

Download to continue reading...

Head First Software Development: A Learner’s Companion to Software Development Head First PMP: A Learner’s Companion to Passing the Project Management Professional Exam Head First Physics: A learner’s companion to mechanics and practical physics (AP Physics B - Advanced Placement) Head First Rails: A Learner’s Companion to Ruby on Rails Software Engineering: The Current Practice (Chapman & Hall/CRC Innovations in Software Engineering and Software Development Series) Oxford Learner’s German Dictionary (Oxford Learner’s Dictionary) Head First Data Analysis: A learner’s guide to big numbers, statistics, and good decisions Head First Programming: A learner’s guide to programming using the Python language Head First HTML and CSS: A Learner’s Guide to Creating Standards-Based Web Pages Agile Software Development with Scrum (Series in Agile Software Development) Agile Project Management: Agile Revolution, Beyond Software Limits: A Practical Guide to Implementing Agile Outside Software Development (Agile Business Leadership, Book 4) Envisioning the Faculty for the Twenty-First Century: Moving to a Mission-Oriented and Learner-Centered Model (The American Campus) Companion Planting: Companion Gardening - A Practical Guide For Beginners To Learn Everything About Companion Planting (Organic Gardening, Container Gardening, Vegetable Gardening) Agile Product Management: Product Owner: 27 Tips To Manage Your Product And Work With Scrum Teams (scrum, scrum master, agile development, agile software development) Agile Project Management: QuickStart Guide - The Simplified Beginners Guide To Agile Project Management (Agile Project Management, Agile Software Development, Agile Development, Scrum) The Software Requirements Memory Jogger: A Pocket Guide to Help Software And Business Teams Develop And Manage Requirements (Memory Jogger) Don’t Buy Software For Your Small Business Until You Read This Book: A guide to choosing the right software for your SME & achieving a rapid return on your investment Software Agreements Line by Line, 2nd ed.: A Detailed Look at Software Agreements and How to Draft Them to Meet Your Needs IEC 62304 Ed. 1.0 b:2006, Medical device software - Software life cycle processes Headscarves, Head Wraps & More: How to Look Fabulous in 60 Seconds with Easy Head Wrap Tying Techniques