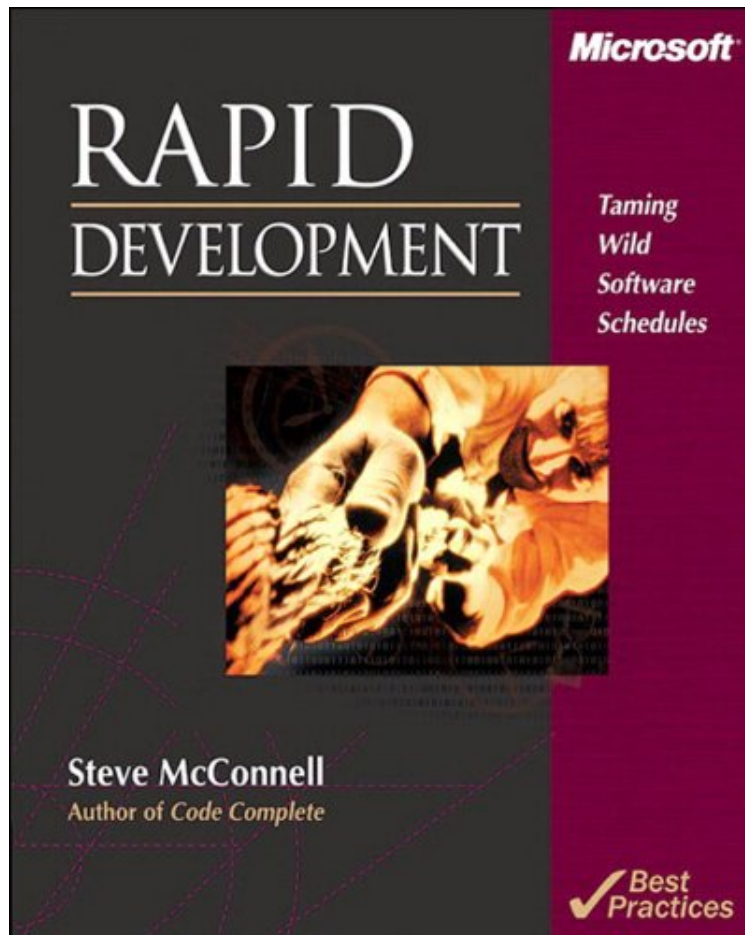


(Free download) Rapid Development (Developer Best Practices)

Rapid Development (Developer Best Practices)

Von Steve McConnell

ebooks | Download PDF | *ePub | DOC | audiobook



 Download

 Read Online

Produktinformation -Verkaufsrank: #314001 in eBooksVerffentlicht am: 1996-07-02Erscheinungsdatum: 1996-07-02File Name: B00JDMPOB6Anzahl der Produkte: 1 | File size: 64.Mb

Von Steve McConnell : Rapid Development (Developer Best Practices) before purchasing it in order to gage whether or not it would be worth my time, and all praised Rapid Development (Developer Best Practices):

KundenrezensionenHilfreichste Kundenrezensionen0 von 0 Kunden fanden die folgende Rezension hilfreich. Rapid Team Development with ColdFusionVon Michael LightI read Michael Dinowitz's review of Steve McConnell's Code Complete in last week's issue of Fusion Authority, and wanted to tell you about another great book by him called "Rapid Development". This one is good for team ColdFusion projects on a short deadline (and what ColdFusion project is not rapid development!). The sub-title "Taming Wild Software Schedules" sums it up.The book covers both technical ideas (such as estimation, risk management and mini milestones) and the psychology of both programmers and teams. The latter may be particularly useful on the high burnout 12-hour-a-day work cycle of many ColdFusion programming teams I have seen. Keeping everyone motivated and working together as a team is often more important than technical programming tricks on a large project.Project Management CookbookThe second half of the book, "Best Practices," is a cookbook of 36 techniques for project leaders designed to stop overruns and control those out-of-

control, large ColdFusion projects. I have been developing for over 20 years, and nearly all of McConnell's tips ring true. The book is not about ColdFusion per se; it concerns any Rapid Development language. I liked the fact that he rated each of these cookbook methods by the following criteria: Potential reduction from nominal schedule Improvement in progress visibility Effect on schedule risk Chance of first time success Chance of long-term success Here are some of the 36 classic mistakes that McConnell describes in detail: People Related Mistakes Heroics Adding people to a late project Politics placed over substance (etc.) Process Related Mistakes Abandonment of planning under pressure Planning to catch up later "Code-like-hell" programming (etc.) Technology Related Mistakes Silver-Bullet syndrome Overestimating Savings From New Tools or Methods Switching Tools in the Middle of a Project (etc.) If you are like me and have been involved in a few "projects from hell" then I am sure you recognise some (or all) of the above! There are also fun sidebars from the trenches of real development teams, both successes and disasters. Twenty real-life case studies round it out. This is a long book (650 pp) but I have found myself dipping into it for new insights in my daily work as a ColdFusion project manager. Review by: Michael Smith, TeraTech2 von 2 Kunden fanden die folgende Rezension hilfreich. Best practices for developers and managers Von Peter Crowther I've been kicking around the industry for 25 years, as a developer and technical manager, and this book expresses many concepts that I have half-formed over that time but never really been able to express. McConnell also backs them up with relevant data. If you're a developer, read this for the sections on estimation, negotiation and overtime. Then you'll want to read the rest of it. If you're a technical manager, just buy and read the whole thing. It's all *very* relevant, because you're in the middle of the whole process. Buy the book because you'll want to keep it nearby afterwards to point your line managers and developers at the hard data when they try to get round you! If you're a line manager, read this for the sections on teamwork, how to motivate developers, and overtime. Then read the rest anyway; it's relevant to you, too. McConnell has an easy style, and the book is an entertaining read. It's also split into small sections; I read it over three weeks on the train to and from work. 1 von 1 Kunden fanden die folgende Rezension hilfreich. RAPID DEVELOPMENT: The project manager's Bible Von Ein Kunde I work for an Internet software development company, and I have made this book required reading for every project manager and technical lead in our company. McConnell's combination of conceptual knowledge, supported by hard facts, is a rare thing in software development "how-tos." The best-practices section at the back of the book is an invaluable reference. His "bad" case studies depressed me sometimes -- mainly because they were too close to my own projects -- but the "good" ones have become the scripts for my presentations to clients. He has a way of capturing the essence of the atmosphere in a development shop, so the case studies feel as if they took place in your own office. I would highly recommend this book to anyone whose job is managing the development process, whether that be in a technical lead or a project management position. Maybe if more people read this book and follow its guidelines, we could all stop working weekends.

Kurzbeschreibung Corporate and commercial software-development teams all want solutions for one important problem: how to get their high-pressure development schedules under control. In RAPID DEVELOPMENT, author Steve McConnell addresses that concern head-on with overall strategies, specific best practices, and valuable tips that help shrink and control development schedules and keep projects moving. Inside, you'll find: A rapid-development strategy that can be applied to any project and the best practices to make that strategy work Candid discussions of great and not-so-great rapid-development practices estimation, prototyping, forced overtime, motivation, teamwork, rapid-development languages, risk management, and many others A list of classic mistakes to avoid for rapid-development projects, including creeping requirements, shortchanged quality, and silver-bullet syndrome Case studies that vividly illustrate what can go wrong, what can go right, and how to tell which direction your project is going RAPID DEVELOPMENT is the real-world guide to more efficient applications development. de I can hear some of you exclaiming, "How can you possibly recommend a book about software scheduling published by Microsoft Press and written by a consultant to Microsoft?!" Well, put aside any preconceived biases. This is a tremendous book on effective scheduling software development, and it drinks deeply from the wisdom of all the classics in the field such as Brook's Mythical Man Month -- and is likely well-informed by McConnell's experiences, good and bad, in Redmond. The nine page section entitled "Classic Mistakes Enumerated" is alone worth the price of admission and should be required reading for all developers, leads, and managers. Here are some types of the 36 classic mistakes that McConnell describes in detail: People Related Mistakes Heroics Adding people to a late project Politics placed over substance (etc.) Process Related Mistakes Abandonment of planning under pressure Planning to catch up later "Code-like-hell" programming (etc.) Technology Related Mistakes Silver-Bullet syndrome Overestimating savings from new tools or methods Switching tools in the middle of a project (etc.) I suspect that if you've ever been involved in software development, you winced after reading each of these nine points. And you will learn a great deal from the remaining 640 pages about concrete solutions. My only substantive gripe: cheesy Powerpoint graphics. Nonetheless, this book is Very Highly Recommended. Kurzbeschreibung Corporate and commercial software-development teams all want solutions for one important problem: how to get their high-pressure development schedules under control. In RAPID

DEVELOPMENT, author Steve McConnell addresses that concern head-on with overall strategies, specific best practices, and valuable tips that help shrink and control development schedules and keep projects moving. Inside, you'll find: A rapid-development strategy that can be applied to any project and the best practices to make that strategy work Candid discussions of great and not-so-great rapid-development practices estimation, prototyping, forced overtime, motivation, teamwork, rapid-development languages, risk management, and many others A list of classic mistakes to avoid for rapid-development projects, including creeping requirements, shortchanged quality, and silver-bullet syndrome Case studies that vividly illustrate what can go wrong, what can go right, and how to tell which direction your project is going RAPID DEVELOPMENT is the real-world guide to more efficient applications development.