

Peopleware Productive Projects Teams 3rd

Why People on Your Team Don't Act on Good Ideas, and how to Convince Them They Should

Productive Projects and Teams

your journey to mastery, 20th Anniversary Edition

Peopleware

Death March

Rapid Development

The Intelligence Behind Successful Software Management

Project Management)

Team Writing

Managing Humans

The Software Developer's Life Manual

The Pragmatic Programmer

Productive Projects and Teams (3rd Edition)

Peopleware

The Design of Design

Joel Spolsky's Concise Guide to Finding the Best Technical Talent

A Novel about Project Management

Understanding Patterns of Project Behavior

Five Core Metrics

More Joel on Software

Management, Measurement & Estimation

Further Thoughts on Diverse and Occasionally Related Matters That Will Prove of Interest to Software Developers, Designers, and Managers, and to Those Who, Whether by Good Fortune or Ill Luck, Work with Them in Some Capacity

Dreaming in Code

How Project Leaders Create Cohesive, Happy, High-Performing Teams

Startup Engineering Management, 2nd Edition

Productive Projects and Teams

Better Productivity Through Collaboration

Artful Making

Dark Harbor House

Driving Technical Change

Fourteen Observations of Good Scrum Practice

What Managers Need to Know about how Artists Work

Controlling Software Projects

The Deadline

Managing Risk on Software Projects

Rethinking Productivity in Software Engineering

Productive Projects an Teams

Taming Wild Software Schedules

Facts and Fallacies of Software Engineering

Peopleware Productive Projects Teams 3rd

Downloaded from ecobankpayservices.ecobank.com by guest

DEREK YANG

Why People on Your Team Don't Act on Good Ideas, and how to Convince Them They Should

Apress

A "good" programmer can outproduce five, ten, and sometimes more run-of-the-mill programmers. The secret to success for any software company then is to hire the good programmers. But how to do that? In *Joel on Hiring*, Joel Spolsky draws from his experience both at Microsoft and running his own successful software company based in New York City. He writes humorously, but seriously about his methods for sorting resumes, for finding great candidates, and for interviewing, in person and by phone. Joel's methods are not complex, but they do get to the heart of the matter: how to recognize a great developer when you see one.

Productive Projects and Teams "O'Reilly Media, Inc."

Widely considered one of the best practical guides to programming, Steve McConnell's original *CODE COMPLETE* has been helping developers write better software for more than a decade. Now

this classic book has been fully updated and revised with leading-edge practices—and hundreds of new code samples—illustrating the art and science of software construction. Capturing the body of knowledge available from research, academia, and everyday commercial practice, McConnell synthesizes the most effective techniques and must-know principles into clear, pragmatic guidance. No matter what your experience level, development environment, or project size, this book will inform and stimulate your thinking—and help you build the highest quality code. Discover the timeless techniques and strategies that help you: Design for minimum complexity and maximum creativity Reap the benefits of collaborative development Apply defensive programming techniques to reduce and flush out errors Exploit opportunities to refactor—or evolve—code, and do it safely Use construction practices that are right-weight for your project Debug problems quickly and effectively Resolve critical construction issues early and correctly Build quality into the beginning, middle, and end of your project

your journey to mastery, 20th Anniversary Edition Pearson Education

Fourteen Observations of Good Scrum Practice is based on my years of practical experience applying Scrum in a variety of domains for a number of different teams - from small start-ups to

companies that create products regulated by the FDA. The observations contained in this guide are a collection of the common patterns discovered through trial-and-error that made each of these teams and organizations successful with Scrum.

Peopleware Crown Business

Two of the computer industry's best-selling authors and lecturers return with a new edition of the software management book that started a revolution. With humor and wisdom drawn from years of management and consulting experience, DeMarco and Lister demonstrate that the major issues of software development are human, not technical -- and that managers ignore them at their peril. Now, with a new Preface and eight new chapters, the authors enlarge upon their previous ideas and add fresh insights, examples, and anecdotes. Discover dozens of helpful tips on- putting more quality into a product- loosening up formal methodologies- fighting corporate entropy- making it acceptable to be uninterruptible *Peopleware*, 2nd ed. shows you how to cultivate teams that are healthy and productive. The answers aren't easy -- just incredibly successful.

Death March Stanford University Press

Controlling Software Projects shows managers how to organize software projects so they are

objectively measurable, and prescribes techniques for making early and accurate projections of time and cost to deliver.

Rapid Development FT Press

The Principles of Project Management lays out clear steps that anyone can follow to get projects done right, and delivered on time. This full color book covers: Why Project Management is important The 6 fundamental truths of project management Getting started: Discovering, Initiating, Planning and Resourcing a project Getting the Job Done: Executing and controlling Keeping it Smooth: Communication, collaboration and managing change Following through: Ongoing support and maintenance, measuring operational success Resources: Review of various tools, recommended reading, professional resources for project management Short, and to the point, this book aims to do to provide a solid foundation for anyone who finds themselves responsible for executing projects. From the Back Cover Every project you manage will be unique. Scope, budgets, team dynamics, and timeframes will differ. As a project manager, the most important factor in achieving project success will be your understanding of The Principles Of Project Management. This book will show you that project management isn't rocket science: using the information contained in this book, you'll deliver projects on time and on budget, again and again. With The Principles Of Project Management you'll: Learn how to start every project on the right foot. Master the planning, execution, and control of your projects. Discover the secrets of effective communication and change management. Identify project warning signals and learn to keep your projects on track. Understand the benefits of using the right tools, resources, and people. Learn how to give a superstar project handover. And much, much more

The Intelligence Behind Successful Software Management Apress

The authors show how to "manage" ingenuity--and "manufacture" the next great idea, in other words they tell what managers need to know about how artists and highly creative people work.

Project Management) Pearson Education

No other book combines the practice of project management and leadership into one balanced approach with practical examples--except this book.

Team Writing Pearson Education

This practical handbook on software project success and survival explains how to confront five important issues involved in all software projects--people, politics, process, project management, and tools.

Managing Humans Addison-Wesley Professional

From prolific and influential consultant and author Tom DeMarco comes a project management novel that vividly illustrates the principles--and the outright absurdities--that affect the productivity of a software development team. With his trademark wit set free in the novel format, DeMarco centers the plot around the development of six software products. Mr. Tompkins, a manager downsized from a giant telecommunications company, divides the huge staff of developers at his disposal into eighteen teams--three for each of the software products. The teams are different sizes and use different methods, and they compete against each other and against an impossible deadline. With these teams--and with the help of numerous "fictionalized" consultants who come to his aid--Tompkins tests the project management principles he has gathered over a lifetime. Each chapter closes with journal entries that form the core of the eye-opening approaches to management illustrated in this entertaining novel.

Addison-Wesley

Provides a variety of ideas, techniques, and strategies for effective software development.

The Software Developer's Life Manual Addison-Wesley

Few books in computing have had as profound an influence on software management as Peopleware . The unique insight of this longtime best seller is that the major issues of software development are human, not technical. They're not easy issues; but solve them, and you'll maximize your chances of success. "Peopleware has long been one of my two favorite books on software engineering. Its underlying strength is its base of immense real experience, much of it quantified. Many, many varied projects have been reflected on and distilled; but what we are given is not just lifeless distillate, but vivid examples from which we share the authors' inductions. Their premise is right: most software project problems are sociological, not technological. The insights on team jelling and work environment have changed my thinking and teaching. The third edition adds strength to strength." — Frederick P. Brooks, Jr., Kenan Professor of Computer Science, University of North Carolina at Chapel Hill, Author of The Mythical Man-Month and The Design of Design "Peopleware is the one book that everyone who runs a software team needs to read and

re-read once a year. In the quarter century since the first edition appeared, it has become more important, not less, to think about the social and human issues in software development. This is the only way we're going to make more humane, productive workplaces. Buy it, read it, and keep a stock on hand in the office supply closet." —Joel Spolsky, Co-founder, Stack Overflow "When a book about a field as volatile as software design and use extends to a third edition, you can be sure that the authors write of deep principle, of the fundamental causes for what we readers experience, and not of the surface that everyone recognizes. And to bring people, actual human beings, into the mix! How excellent. How rare. The authors have made this third edition, with its additions, entirely terrific." —Lee Devin and Rob Austin, Co-authors of The Soul of Design and Artful Making For this third edition, the authors have added six new chapters and updated the text throughout, bringing it in line with today's development environments and challenges. For example, the book now discusses pathologies of leadership that hadn't previously been judged to be pathological; an evolving culture of meetings; hybrid teams made up of people from seemingly incompatible generations; and a growing awareness that some of our most common tools are more like anchors than propellers. Anyone who needs to manage a software project or software organization will find invaluable advice throughout the book.

The Pragmatic Programmer Apress

Bring together a wonderfully varied mix of characters in a once-grand Maine island summer cottage, leave them to their own devices over the course of a long, idyllic summer in the late 1940s, and you have all the ingredients for a fine comedy of manners. Author Tom DeMarco starts with a simple little love story, weaves in tantalizing details of the old mansion's not totally respectable history, and adds a hint of gentle satire to create a novel that is touching, memorable, and deliciously entertaining.

Productive Projects and Teams (3rd Edition) Currency

In this comprehensive yet accessible overview for software leaders, the author presents an impactful, action-oriented prescription-covering the practical considerations needed to ensure you reap the full benefits of effective Agile

Peopleware Down East Books

The practice of building software is a "new kid on the block" technology. Though it may not seem this way for those who have been in the field for most of their careers, in the overall scheme of professions, software builders are relative "newbies." In the short history of the software field, a lot of facts have been identified, and a lot of fallacies promulgated. Those facts and fallacies are what this book is about. There's a problem with those facts--and, as you might imagine, those fallacies. Many of these fundamentally important facts are learned by a software engineer, but over the short lifespan of the software field, all too many of them have been forgotten. While reading Facts and Fallacies of Software Engineering , you may experience moments of "Oh, yes, I had forgotten that," alongside some "Is that really true?" thoughts. The author of this book doesn't shy away from controversy. In fact, each of the facts and fallacies is accompanied by a discussion of whatever controversy envelops it. You may find yourself agreeing with a lot of the facts and fallacies, yet emotionally disturbed by a few of them! Whether you agree or disagree, you will learn why the author has been called "the premier curmudgeon of software practice." These facts and fallacies are fundamental to the software building field--forget or neglect them at your peril!

The Design of Design Prentice Hall Professional

Joel, Apress, Blogs, and Blooks ...I was learning the hard way about how to be a publisher and probably spending way too much time looking at web sites and programming than I should have in response to that. Anyway, one day I came across this web site called , which was run by a guy with strong opinions and an unusual, clever writing style, along with a willingness to take on the conventional wisdom. In particular, he was writing this ongoing series about how bad most user interfaces were--mostly because programmers by and large knew, as Joel and I would say, using the same Yiddish-derived NYC vernacular that we both share, "bupkis" about what users really want. And I, like many, was hooked both by the series and the occasional random essay that Joel wrote. And then I had this epiphany: I'm a publisher, I like reading his stuff, why not turn it into a book?... Read the complete Foreword — Gary Cornell, Cofounder, Apress Since the release of the bestselling title Joel on Software in 2004, requests for a sequel have been relentless. So, we went back to the famed JoelonSoftware.com archives and pulled out a new batch of favorites, many of which have been downloaded over one million times. With Joel's newest book, More Joel on Software, you'll get an even better (not to mention updated) feast of Joel's opinions and impressions on software development, software design, running a software business, and so much

more. This is a new selection of essays from the author's web site, <http://www.joelonsoftware.com>. Joel Spolsky started his weblog in March 2000 in order to offer his insights, based on years of experience, on how to improve the world of programming. This weblog has become infamous among the programming world, and is linked to more than 600 other web sites and translated into 30+ languages! Spolsky's extraordinary writing skills, technical knowledge, and caustic wit have made him a programming guru. With the success of Joel on Software, there has been a strong demand for additional gems and advice, and this book is the answer to those requests. Containing a collection of all-new articles from the original, More Joel on Software has even more of an edge than the original, and the tips for running a business or managing people have far broader application than the software industry. We feel it is safe to say that this is the most useful book you will buy this year.

Joel Spolsky's Concise Guide to Finding the Best Technical Talent Pearson Education

"One of the most significant books in my life." —Obie Fernandez, Author, The Rails Way "Twenty years ago, the first edition of The Pragmatic Programmer completely changed the trajectory of my career. This new edition could do the same for yours." —Mike Cohn, Author of Succeeding with Agile, Agile Estimating and Planning, and User Stories Applied ". . . filled with practical advice, both technical and professional, that will serve you and your projects well for years to come." —Andrea Goulet, CEO, Corgibytes, Founder, LegacyCode.Rocks ". . . lightning does strike twice, and this book is proof." —VM (Vicky) Brasseur, Director of Open Source Strategy, Juniper Networks The Pragmatic Programmer is one of those rare tech books you'll read, re-read, and read again over the years. Whether you're new to the field or an experienced practitioner, you'll come away with fresh insights each and every time. Dave Thomas and Andy Hunt wrote the first edition of this influential book in 1999 to help their clients create better software and rediscover the joy of coding. These lessons have helped a generation of programmers examine the very essence of software development, independent of any particular language, framework, or methodology, and the Pragmatic philosophy has spawned hundreds of books, screencasts, and audio books, as well as thousands of careers and success stories. Now, twenty years later, this new edition re-examines what it means to be a modern programmer. Topics range from personal responsibility and career development to architectural techniques for keeping your code flexible and easy to adapt and reuse. Read this book, and you'll learn how to: Fight software rot Learn continuously Avoid the trap of duplicating knowledge Write flexible, dynamic, and adaptable code Harness the power of basic tools Avoid programming by coincidence Learn real requirements Solve the underlying problems of concurrent code Guard against security vulnerabilities Build teams of Pragmatic Programmers Take responsibility for your work and career Test ruthlessly and effectively, including property-based testing Implement the Pragmatic Starter Kit Delight your users Written as a series of self-contained sections and filled with classic and fresh anecdotes, thoughtful examples, and interesting analogies, The Pragmatic Programmer illustrates the best approaches and major pitfalls of many different aspects of software development. Whether you're a new coder, an experienced programmer, or a manager responsible for software projects, use these lessons daily, and you'll quickly see improvements in personal productivity, accuracy, and job satisfaction. You'll learn skills and develop habits and attitudes that form the foundation for long-term success in your career. You'll become a Pragmatic Programmer. Register your book for convenient access to downloads, updates, and/or corrections as they become available. See inside book for details.

A Novel about Project Management PeoplewareProductive Projects and Teams

PeoplewareProductive Projects and TeamsPearson Education

Understanding Patterns of Project Behavior Addison-Wesley

Often referred to as the "black art" because of its complexity and uncertainty, software estimation is not as difficult or puzzling as people think. In fact, generating accurate estimates is straightforward—once you understand the art of creating them. In his highly anticipated book, acclaimed author Steve McConnell unravels the mystery to successful software estimation—distilling academic information and real-world experience into a practical guide for working software professionals. Instead of arcane treatises and rigid modeling techniques, this guide highlights a proven set of procedures, understandable formulas, and heuristics that individuals and development teams can apply to their projects to help achieve estimation proficiency. Discover how to: Estimate schedule and cost—or estimate the functionality that can be delivered within a given time frame Avoid common software estimation mistakes Learn estimation techniques for you, your team, and your organization * Estimate specific project activities—including development, management, and defect correction Apply estimation

approaches to any type of project—small or large, agile or traditional. Navigate the shark-infested political waters that surround project estimates. When many corporate software projects are failing, McConnell shows you what works for successful software estimation.

Five Core Metrics Macmillan Higher Education

This is the digital version of the printed book (Copyright © 2003). To succeed in the software

industry, managers need to cultivate a reliable development process. By measuring what teams have achieved on previous projects, managers can more accurately set goals, make bids, and ensure the successful completion of new projects. Acclaimed long-time collaborators Lawrence H. Putnam and Ware Myers present simple but powerful measurement techniques to help software

managers allocate limited resources and track project progress. Drawing new findings from an extensive database of software project metrics, the authors demonstrate how readers can control projects with just Five Core Metrics—Time, Effort, Size, Reliability, and Process Productivity. With these metrics, managers can adjust ongoing projects to changing conditions—surprises that would otherwise cause project failure.

Related with Peopleware Productive Projects Teams 3rd:

[© Peopleware Productive Projects Teams 3rd Ap Human Geography Scale Of Analysis](#)

[© Peopleware Productive Projects Teams 3rd Ap Human Geography Practice Test Unit 2](#)

[© Peopleware Productive Projects Teams 3rd Ap Physics 2 Study Guide](#)