

Peopleware Productive Projects And Teams Ebook Pdf

Yeah, reviewing a books **Peopleware Productive Projects And Teams Ebook Pdf** could amass your near friends listings. This is just one of the solutions for you to be successful. As understood, completion does not recommend that you have astounding points.

Comprehending as without difficulty as bargain even more than further will find the money for each success. adjacent to, the statement as with ease as perspicacity of this Peopleware Productive Projects And Teams Ebook Pdf can be taken as skillfully as picked to act.

Software Project Survival Guide - Steve McConnell 1998

Looks at a successful software project and provides details for software development for clients using object-oriented design and programming.

[Chemical Kinetics and Catalysis](#) -

The Deadline - Tom DeMarco 1997

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.

The Art of Agile Practice - Bhuvan Unhelkar 2016-04-19

The Art of Agile Practice: A Composite Approach for Projects and Organizations presents a consistent, integrated, and strategic approach to achieving "Agility" in your business. Transcending beyond Agile as a software development method, it covers the gamut of methods in an organization-including business processes, governance standards, project ma

The Mythical Man-month - Frederick P. Brooks (Jr.) 1975

The orderly Sweet-Williams are dismayed at their son's fondness for the messy pastime of gardening.

The Pragmatic Programmer - Andrew Hunt 1999-10-20

What others in the trenches say about The Pragmatic Programmer... "The cool thing about this book is that it's great for keeping the programming process fresh. The book helps you to continue to grow and clearly comes from people who have been there." —Kent Beck, author of Extreme Programming Explained: Embrace Change "I found this book to be a great mix of solid advice and wonderful analogies!" —Martin Fowler, author of Refactoring and UML Distilled "I would buy a copy, read it twice, then tell all my colleagues to run out and grab a copy. This is a book I would never loan because I would worry about it being lost." —Kevin Ruland, Management Science, MSG-Logistics "The wisdom and practical experience of the authors is obvious. The topics presented are relevant and useful.... By far its greatest strength for me has been the outstanding analogies—tracer bullets, broken windows, and the fabulous helicopter-based explanation of the need for orthogonality, especially in a crisis situation. I have little doubt that this book will eventually become an excellent source of useful information for journeymen programmers and expert mentors alike." —John Lakos, author of Large-Scale C++ Software Design "This is the sort of book I will buy a dozen copies of when it comes out so I can give it to my clients." —Eric Vought, Software Engineer "Most modern books on software development fail to cover the basics of what makes a great software developer, instead spending their time on syntax or technology where in reality the greatest leverage possible for any software team is in having talented developers who really know their craft well. An

excellent book." —Pete McBreen, Independent Consultant "Since reading this book, I have implemented many of the practical suggestions and tips it contains. Across the board, they have saved my company time and money while helping me get my job done quicker! This should be a desktop reference for everyone who works with code for a living." —Jared Richardson, Senior Software Developer, iRenaissance, Inc. "I would like to see this issued to every new employee at my company...." —Chris Cleeland, Senior Software Engineer, Object Computing, Inc. "If I'm putting together a project, it's the authors of this book that I want. . . . And failing that I'd settle for people who've read their book." —Ward Cunningham Straight from the programming trenches, The Pragmatic Programmer cuts through the increasing specialization and technicalities of modern software development to examine the core process--taking a requirement and producing working, maintainable code that delights its users. It covers topics ranging 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; Avoid the trap of duplicating knowledge; Write flexible, dynamic, and adaptable code; Avoid programming by coincidence; Bullet-proof your code with contracts, assertions, and exceptions; Capture real requirements; Test ruthlessly and effectively; Delight your users; Build teams of pragmatic programmers; and Make your developments more precise with automation. Written as a series of self-contained sections and filled with entertaining anecdotes, thoughtful examples, and interesting analogies, The Pragmatic Programmer illustrates the best practices 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.

Controlling Software Projects - Tom DeMarco 1982

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.

Project Retrospectives - Norman Kerth 2013-07-15

This is the digital copy of the printed book (Copyright © 2001). With detailed scenarios, imaginative illustrations, and step-by-step instructions, consultant and speaker Norman L. Kerth guides readers through productive, empowering retrospectives of project performance. Whether your shop calls them postmortems or postpartums or something else, project retrospectives offer organizations a formal method for preserving the valuable lessons learned from the successes and failures of every project. These lessons and the changes identified by the community will foster stronger teams and savings on subsequent efforts. For a retrospective to be effective and successful, though, it needs to be safe. Kerth shows facilitators and participants how to defeat the fear of retribution and establish an air of mutual trust. One tool is Kerth's Prime Directive: Regardless of what we discover, we must understand and truly believe that everyone did the best job he or she could, given what was known at the time, his or her skills and abilities, the resources available, and the situation at hand. Applying years of experience as a project retrospective facilitator for software organizations, Kerth reveals his secrets for managing the sensitive, often emotionally charged issues that arise as teams relive and learn from each project.

Team Geek - Brian W. Fitzpatrick 2012-07-06

Annotation In this book, Brian Fitzpatrick and Ben Collins-Sussman cover basic patterns and anti-patterns for working with other people, teams and users while trying to develop software.

Adrenaline Junkies and Template Zombies - Tom DeMarco 2013-07-15

This is the digital version of the printed book (Copyright © 2008). Adrenaline junkies, dead fish, project sluts, true believers, Lewis and Clark, template zombies . . . Most developers, testers, and managers on IT projects are pretty good at recognizing patterns of behavior and gut-level hunches, as in, "I sense that this project is headed for disaster." But it has always been more difficult to transform these patterns and hunches into a usable form, something a team can debate, refine, and use. Until now. In *Adrenaline Junkies and Template Zombies*, the six principal consultants of The Atlantic Systems Guild present the patterns of behavior they most often observe at the dozens of IT firms they transform each year, around the world. The result is a quick-read guide to identifying nearly ninety typical scenarios, drawing on a combined one-hundred-and-fifty years of project management experience. Project by project, you'll improve the accuracy of your hunches and your ability to act on them. The patterns are presented in an easy-reference format, with names designed to ease communication with your teammates. In just a few words, you can describe what's happening on your project. Citing the patterns of behavior can help you quickly move those above and below you to the next step on your project. You'll find classic patterns such as these: News Improvement Management by Mood Ring Piling On Rattle Yer Dags Natural Authority Food++ Fridge Door and more than eighty more! Not every pattern will be evident in your organization, and not every pattern is necessarily good or bad. However, you'll find many patterns that will apply to your current and future assignments, even in the most ambiguous circumstances. When you assess your situation and follow your next hunch, you'll have the collective wisdom of six world-class consultants at your side.

Measuring and Managing Performance in Organizations - Robert D. Austin 2013-07-15

This is the digital version of the printed book (Copyright © 1996). Based on an award-winning doctoral thesis at Carnegie Mellon University, *Measuring and Managing Performance in Organizations* presents a captivating analysis of the perils of performance measurement systems. In the book's foreword, Peopleware authors Tom DeMarco and Timothy Lister rave, "We believe this is a book that needs to be on the desk of just about anyone who manages anything." Because people often react with unanticipated sophistication when they are being measured, measurement-based management systems can become dysfunctional, interfering with achievement of intended results. Fortunately, as the author shows, measurement dysfunction follows a pattern that can be identified and avoided. The author's findings are bolstered by interviews with eight recognized experts in the use of measurement to manage computer software development: David N. Card, of Software Productivity Solutions; Tom DeMarco, of the Atlantic Systems Guild; Capers Jones, of Software Productivity Research; John Musa, of AT&T Bell Laboratories; Daniel J. Paulish, of Siemens Corporate Research; Lawrence H. Putnam, of Quantitative Software Management; E. O. Tilford, Sr., of Fissure; plus the anonymous Expert X. A practical model for analyzing measurement projects solidifies the text—don't start without it!

Waltzing with Bears - Tom DeMarco 2013

This is the digital version of the printed book (Copyright © 2003). If There's No Risk On Your Next Project, Don't Do It. Greater risk brings greater reward, especially in software development. A company that runs away from risk will soon find itself lagging behind its more adventurous competition. By ignoring the threat of negative outcomes—in the name of positive thinking or a can-do attitude—software managers drive their organizations into the ground. In *Waltzing with Bears*, Tom DeMarco and Timothy Lister—the best-selling authors of *Peopleware*—show readers how to identify and embrace worthwhile risks. Developers are then set free to push the limits. The authors present the benefits of risk management, including that it makes aggressive risk-taking possible, protects management from getting blindsided, provides minimum-cost downside protection, reveals invisible transfers of responsibility, isolates the failure of a subproject. Readers are armed with strategies for confronting the most common risks that software projects face: schedule flaws, requirements inflation, turnover, specification breakdown, and under-performance. *Waltzing with Bears* will help you mitigate the risks—before they turn into project-killing problems. Risks are out there—and they should be there—but there is a way to manage them.

Developer Hegemony - Erik Dietrich

It's been said that software is eating the planet. The modern economy—the world itself—relies on technology. Demand for the people who can produce it far outweighs the supply. So why do developers occupy largely subordinate roles in the corporate structure? *Developer Hegemony* explores the past, present, and future of the corporation and what it means for developers. While it outlines problems with the modern corporate structure, it's ultimately a play-by-play of how to leave the corporate carnival and control your own destiny. And it's an emboldening, specific vision of what software development looks like in the world of developer hegemony—one where developers band together into partner firms of "efficiencers," finally able to command the pay, respect, and freedom that's earned by solving problems no one else can. Developers, if you grow tired of being treated like geeks who can only be trusted to take orders and churn out code, consider this your call to arms. Bring about the autonomous future that's rightfully yours. It's time for developer hegemony.

The Soul of Design - Robert Austin 2012-09-05

What makes the Apple iPhone cool? Bang & Olufsen and Samsung's televisions beautiful? Any of a wide variety of products and services special? The answer is not simply functionality or technology, for competitors' products are often as good. *The Soul of Design* explores the uncanny power of some products to grab and hold attention—to create desire. To understand what sets a product apart in this way, authors Lee Devin and Robert Austin push past personal taste and individual response to adopt a more conceptual approach. They carefully explore the hypothesis that there is something within a "special" product that makes it—well, special. They argue that this *je ne sais quoi* arises from "plot"—the shape that emerges as a product or service arouses and then fulfills expectations. Marketing a special product is, then, a matter of helping its audience perceive its plot and comprehend its qualities. Devin and Austin provide keys to understanding why some products and services stand out in a crowd and how the companies that make them create these hits. Part One of the book introduces the authors' definition of plot in this context; Part Two breaks down the components needed to build a plot; Part Three describes what makes a plot coherent; Part Four takes on the challenges of making coherent products and services attractive to consumers. Part Four also presents detailed casework, which shows how innovators and makers have successfully brought special products to market. Readers will come away with a sensible and clear approach to conceiving of artful products and services. This book will help managers and designers think about engaging with plot, taking aesthetic factors into account to provide consumers with more special things.

The Art of Systems Architecting, Third Edition - Mark W. Maier 2009-01-06

If engineering is the art and science of technical problem solving, systems architecting happens when you don't yet know what the problem is. The third edition of a highly respected bestseller, *The Art of Systems Architecting* provides in-depth coverage of the least understood part of systems design: moving from a vague concept and limited resources to a satisfactory and feasible system concept and an executable program. The book provides a practical, heuristic approach to the "art" of systems architecting. It provides methods for embracing, and then taming, the growing complexity of modern systems. New in the Third Edition: Five major case studies illustrating successful and unsuccessful practices Information on architecture frameworks as standards for architecture descriptions New methods for integrating business strategy and architecture and the role of architecture as the technical embodiment of strategy Integration of process guidance for organizing and managing architecture projects Updates to the rapidly changing fields of software and systems-of-systems architecture Organization of heuristics around a simple and practical process model A Practical Heuristic Approach to the Art of Systems Architecting Extensively rewritten to reflect the latest developments, the text explains how to create a system from scratch, presenting invention/design rules together with clear explanations of how to use them. The author supplies practical guidelines for avoiding common systematic failures while implementing new mandates. He uses a heuristics-based approach that provides an organized attack on very ill-structured engineering problems. Examining architecture as more than a set of diagrams and documents, but as a set of decisions that either drive a system to success or doom it to failure, the book provides methods for integrating business strategy with technical architectural decision making.

Managing and Leading Software Projects - Richard E. Fairley 2011-09-20

The book is organized around basic principles of software project management: planning and estimating, measuring and controlling, leading and communicating, and managing risk. Introduces software development methods, from traditional (hacking, requirements to code, and waterfall) to iterative (incremental build, evolutionary, agile, and spiral). Illustrates and emphasizes tailoring the development process to each project, with a foundation in the fundamentals that are true for all development methods. Topics such as the WBS, estimation, schedule networks, organizing the project team, and performance reporting are integrated, rather than being relegated to appendices. Each chapter in the book includes an appendix that covers the relevant topics from CMMI-DEV-v1.2, IEEE/ISO Standards 12207, IEEE Standard 1058, and the PMI® Body of Knowledge. (PMI is a registered mark of Project Management Institute, Inc.)

Project Management for Information Professionals - Margot Note 2015-11-03

Aimed at practitioners, this handbook imparts guidance on project management techniques in the cultural heritage sector. Information professionals often direct complex endeavors with limited project management training or resources. Project Management for Information Professionals demystifies the tools and processes essential to successful project management and advises on how to manage the interpersonal dynamics and organizational culture that influence the effectiveness of these methods. With this book, readers will gain the knowledge to initiate, plan, execute, monitor, and close projects. offers guidance based on real-world experience prepares readers without prior project management knowledge or experience provides lean, easy-to-read, and jargon-free instructions aimed at information professionals working in libraries, archives, museums

Facts and Fallacies of Software Engineering - Robert L. Glass 2003

Regarding the controversial and thought-provoking assessments in this handbook, many software professionals might disagree with the authors, but all will embrace the debate. Glass identifies many of the key problems hampering success in this field. Each fact is supported by insightful discussion and detailed references.

Trends and Applications in Information Systems and Technologies - Álvaro Rocha 2021-03-28

This book is composed of a selection of articles from The 2021 World Conference on Information Systems and Technologies (WorldCIST'21), held online between 30 and 31 of March and 1 and 2 of April 2021 at Hangra de Heroismo, Terceira Island, Azores, Portugal. WorldCIST is a global forum for researchers and practitioners to present and discuss recent results and innovations, current trends, professional experiences and challenges of modern information systems and technologies research, together with their technological development and applications. The main topics covered are: A) Information and Knowledge Management; B) Organizational Models and Information Systems; C) Software and Systems Modeling; D) Software Systems, Architectures, Applications and Tools; E) Multimedia Systems and Applications; F) Computer Networks, Mobility and Pervasive Systems; G) Intelligent and Decision Support Systems; H) Big Data Analytics and Applications; I) Human-Computer Interaction; J) Ethics, Computers & Security; K) Health Informatics; L) Information Technologies in Education; M) Information Technologies in Radiocommunications; N) Technologies for Biomedical Applications.

Dinosaur Brains - Albert J. Bernstein 1990

Offers advice on getting along in the working world, revealing the methods of staying in control, dealing with irrationality in others, and recognizing your own rationality

Managing the Unmanageable - Mickey W. Mantle 2012-09-16

“Mantle and Lichty have assembled a guide that will help you hire, motivate, and mentor a software development team that functions at the highest level. Their rules of thumb and coaching advice are great blueprints for new and experienced software engineering managers alike.” —Tom Conrad, CTO, Pandora “I wish I’d had this material available years ago. I see lots and lots of ‘meat’ in here that I’ll use over and over again as I try to become a better manager. The writing style is right on, and I love the personal anecdotes.” —Steve Johnson, VP, Custom Solutions, DigitalFish All too often, software development is deemed unmanageable. The news is filled with stories of projects that have run catastrophically over schedule and budget. Although adding some formal discipline to the development process has improved the situation, it has by no means solved the problem. How can it be, with so much time and money spent to get software development under control, that it remains so unmanageable? In *Managing the Unmanageable: Rules,*

Tools, and Insights for Managing Software People and Teams, Mickey W. Mantle and Ron Lichty answer that persistent question with a simple observation: You first must make programmers and software teams manageable. That is, you need to begin by understanding your people—how to hire them, motivate them, and lead them to develop and deliver great products. Drawing on their combined seventy years of software development and management experience, and highlighting the insights and wisdom of other successful managers, Mantle and Lichty provide the guidance you need to manage people and teams in order to deliver software successfully. Whether you are new to software management, or have already been working in that role, you will appreciate the real-world knowledge and practical tools packed into this guide.

Herding Cats - Hank Rainwater 2008-01-01

This self-help guide is for programmers who need to improve their management and leadership skills.

Artful Making - Robert Daniel Austin 2003

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.

Driving Technical Change - Terrence Ryan 2010

New technologies are popping up every day. Convincing co-workers to adopt them is the hard part. Adobe software evangelist Ryan breaks down the patterns and types of resistance technologists face in many organizations.

Communicating Project Management - Benjamin Lauren 2018-03-22

Communicating Project Management argues that the communication practices of project managers have necessarily become participatory, made up of complex strategies and processes solidly grounded in rhetorical concepts. The book draws on case studies across organizational contexts and combines individual experiences to investigate how project management relies on communication as teams develop products, services, and internal processes. The case studies also provide examples of how project managers can be understood and studied as writers, further arguing project managers must approach communication as designed experience that must be intentionally inclusive. Author Benjamin Lauren illustrates to readers how teams work together to manage projects through complex coordinative communication practices, and highlights how project managers are constantly learning and evolving by analyzing where they succeed and fail. He concludes that technical and professional communicators have a pivotal role in supporting and facilitating participative approaches to communicating project management.

Crystal Clear - Alistair Cockburn 2004-10-19

Carefully researched over ten years and eagerly anticipated by the agile community, *Crystal Clear: A Human-Powered Methodology for Small Teams* is a lucid and practical introduction to running a successful agile project in your organization. Each chapter illuminates a different important aspect of orchestrating agile projects. Highlights include Attention to the essential human and communication aspects of successful projects Case studies, examples, principles, strategies, techniques, and guiding properties Samples of work products from real-world projects instead of blank templates and toy problems Top strategies used by software teams that excel in delivering quality code in a timely fashion Detailed introduction to emerging best-practice techniques, such as Blitz Planning, Project 360°, and the essential Reflection Workshop Question-and-answer with the author about how he arrived at these recommendations, including where they fit with CMMI, ISO, RUP, XP, and other methodologies A detailed case study, including an ISO auditor's analysis of the project Perhaps the most important contribution this book offers is the Seven Properties of Successful Projects. The author has studied successful agile projects and identified common traits they share. These properties lead your project to success; conversely, their absence endangers your project.

Slack - Tom DeMarco 2002-04-09

If your company's goal is to become fast, responsive, and agile, more efficiency is not the answer--you need more slack. Why is it that today's superefficient organizations are ailing? Tom DeMarco, a leading management consultant to both Fortune 500 and up-and-coming companies, reveals a counterintuitive principle that explains why efficiency efforts can slow a company down. That principle is the value of slack, the degree of freedom in a company that allows it to change. Implementing slack could be as simple as adding an assistant to a department and letting high-priced talent spend less time at the photocopier and

more time making key decisions, or it could mean designing workloads that allow people room to think, innovate, and reinvent themselves. It means embracing risk, eliminating fear, and knowing when to go slow. Slack allows for change, fosters creativity, promotes quality, and, above all, produces growth. With an approach that works for new- and old-economy companies alike, this revolutionary handbook debunks commonly held assumptions about real-world management, and gives you and your company a brand-new model for achieving and maintaining true effectiveness.

[Understanding the Agile Manifesto](#) - Larry Apke 2015-08-10

Agile is one of the most popular software development methodologies used by organizations all over the world. It is characterized by adaptability, flexibility and self-organization, but what does it mean to truly "be" Agile instead of just "doing" Agile? This book offers in-depth commentary and explanations on the Agile methodology's foundation, the Agile Manifesto. Larry Apke, a seasoned Agile coach, uses his own experiences to provide a clear, understandable path to implementing and succeeding with Agile for organizations and individuals.

Team Writing - Joanna Wolfe 2017-07-01

Built around real group interactions, Team Writing is a flexible, hybrid resource that pairs videos with a brief print book. Based on research revealing major problems at all stages of peer group work, the book shows how written communication can help technical writing students contribute to team projects in a meaningful way — and provides strategies for dealing with the breakdowns that can derail a project's success. Numerous examples highlight the kind of written communication that helps teams thrive. Short, Web-based videos depict student teams in action, going beyond the textbook to show what real collaboration looks and sounds like.

The Productivity Project - Chris Bailey 2016-01-05

A fresh, personal, and entertaining exploration of a topic that concerns all of us: how to be more productive at work and in every facet of our lives. Chris Bailey turned down lucrative job offers to pursue a lifelong dream—to spend a year performing a deep dive experiment into the pursuit of productivity, a subject he had been enamored with since he was a teenager. After obtaining his business degree, he created a blog to chronicle a year-long series of productivity experiments he conducted on himself, where he also continued his research and interviews with some of the world's foremost experts, from Charles Duhigg to David Allen. Among the experiments that he tackled: Bailey went several weeks with getting by on little to no sleep; he cut out caffeine and sugar; he lived in total isolation for 10 days; he used his smartphone for just an hour a day for three months; he gained ten pounds of muscle mass; he stretched his work week to 90 hours; a late riser, he got up at 5:30 every morning for three months—all the while monitoring the impact of his experiments on the quality and quantity of his work. The Productivity Project—and the lessons Chris learned—are the result of that year-long journey. Among the counterintuitive insights Chris Bailey will teach you: · slowing down to work more deliberately; · shrinking or eliminating the unimportant; · the rule of three; · striving for imperfection; · scheduling less time for important tasks; · the 20 second rule to distract yourself from the inevitable distractions; · and the concept of productive procrastination. In an eye-opening and thoroughly engaging read, Bailey offers a treasure trove of insights and over 25 best practices that will help you accomplish more.

[Peopleware](#) - Tom DeMarco 2013

Most software project problems are sociological, not technological. Peopleware is a book on managing software projects.

The Art of Agile Development - James Shore 2008

For those considering Extreme Programming, this book provides no-nonsense advice on agile planning, development, delivery, and management taken from the authors' many years of experience. While plenty of books address the what and why of agile development, very few offer the information users can apply directly.

[Why Does Software Cost So Much?](#) - Tom DeMarco 1995

Known for his ability to find provocative answers to the most puzzling questions, Tom DeMarco explores a wide range of issues in twenty-four masterful essays. The offerings range from the wise to the kooky -- in fact, many of them defy categorization. But all are marked by the author's eye-opening perspectives on

topics that demand your professional attention. Drawing together several essays published in such journals as IEEE Software and American Programmer, plus ten all-new papers never seen beyond his circle of colleagues, Tom DeMarco tackles a multitude of tough subjects and wrestles fresh insight out of them. Here's a compact, compelling edition of this acclaimed consultant's views on software engineering. Subjects include management-aided engineering, documentation, desktop video, productivity, software factories, teams, measurement, icons, and more! Essays Include* Why Does Software Cost So Much?* Mad About Measurement* Software Productivity: The Covert Agenda* The Choir and the Team* Management-Aided Software Engineering (with Sheila Brady of Apple Computer)* Lean and Mean* Software Development: State of the Art vs. State of the Practice (with Tim Lister)* Twenty Years of Software Engineering: Looking Forward, Looking Back* "If We Did Only One Thing to Improve . . ."-- plus fifteen more!

[The Effective Engineer](#) - Edmond Lau 2015-03-19

Introducing The Effective Engineer--the only book designed specifically for today's software engineers, based on extensive interviews with engineering leaders at top tech companies, and packed with hundreds of techniques to accelerate your career.

[Debugging Teams](#) - Brian W. Fitzpatrick 2015-10-13

In the course of their 20+-year engineering careers, authors Brian Fitzpatrick and Ben Collins-Sussman have picked up a treasure trove of wisdom and anecdotes about how successful teams work together. Their conclusion? Even among people who have spent decades learning the technical side of their jobs, most haven't really focused on the human component. Learning to collaborate is just as important to success. If you invest in the "soft skills" of your job, you can have a much greater impact for the same amount of effort. The authors share their insights on how to lead a team effectively, navigate an organization, and build a healthy relationship with the users of your software. This is valuable information from two respected software engineers whose popular series of talks—including "Working with Poisonous People"—has attracted hundreds of thousands of followers.

Making Things Happen - Scott Berkun 2008-03-25

Offers a collection of essays on philosophies and strategies for defining, leading, and managing projects. This book explains to technical and non-technical readers alike what it takes to get through a large software or web development project. It does not cite specific methods, but focuses on philosophy and strategy.

[The Problem with Software](#) - Adam Barr 2018-10-23

An industry insider explains why there is so much bad software—and why academia doesn't teach programmers what industry wants them to know. Why is software so prone to bugs? So vulnerable to viruses? Why are software products so often delayed, or even canceled? Is software development really hard, or are software developers just not that good at it? In *The Problem with Software*, Adam Barr examines the proliferation of bad software, explains what causes it, and offers some suggestions on how to improve the situation. For one thing, Barr points out, academia doesn't teach programmers what they actually need to know to do their jobs: how to work in a team to create code that works reliably and can be maintained by somebody other than the original authors. As the size and complexity of commercial software have grown, the gap between academic computer science and industry has widened. It's an open secret that there is little engineering in software engineering, which continues to rely not on codified scientific knowledge but on intuition and experience. Barr, who worked as a programmer for more than twenty years, describes how the industry has evolved, from the era of mainframes and Fortran to today's embrace of the cloud. He explains bugs and why software has so many of them, and why today's interconnected computers offer fertile ground for viruses and worms. The difference between good and bad software can be a single line of code, and Barr includes code to illustrate the consequences of seemingly inconsequential choices by programmers. Looking to the future, Barr writes that the best prospect for improving software engineering is the move to the cloud. When software is a service and not a product, companies will have more incentive to make it good rather than "good enough to ship."

[The Principles of Project Management \(SitePoint\)](#) - Meri Williams 2008-03-13

Presents information on principles that can be used in planning, managing, and leading business projects.

[Corporate Real Estate Asset Management](#) - Barry Haynes 2010-10-28

It is important for those studying and practicing in real estate and property management to learn to manage property assets effectively, to be able to provide their companies with effective property and facilities solutions. This book raises the awareness of how real estate management can support business, transform the workplace and impact upon people and productivity, ensuring that costs are minimized and profit maximized. Written for advanced undergraduate students on property related courses, it provides

them with a rounded understanding by aligning the subject with estates management, facilities management and business strategy. Case studies and action plans provide real insight and make this book an essential reference for those at the start of their careers in real estate and facilities management.
The Elements of Scrum - Chris Sims 2011
A practical field guide to the practice of scrum, an agile software project management methodology.