
Read PDF Scope Document Template For Software Project

Getting the books **Scope Document Template For Software Project** now is not type of inspiring means. You could not forlorn going when book heap or library or borrowing from your contacts to right to use them. This is an extremely simple means to specifically get lead by on-line. This online statement Scope Document Template For Software Project can be one of the options to accompany you following having additional time.

It will not waste your time. allow me, the e-book will totally heavens you other event to read. Just invest little mature to entry this on-line publication **Scope Document Template For Software Project** as well as review them wherever you are now.

APJWLE - GOODMAN BAUTISTA

SCM practices are recognised as core functional areas in assisting a project team to identify, control, audit, and report on all configuration items of a project. Consequently they are then better able to control changes to the working environment. Moreira presents a totally unique book, offering a "how-to" guide for SCM implementation for commercial and technology fields. A thoroughly practical approach; this guide includes examples and instruction of SCM tasks. This book has an easy to follow set of tasks that can be customized to assist a SCM professional in implementing SCM in a more efficient and expedient manner while also imparting SCM knowledge. Provides a customisable step-by-step process in implementing SCM Discusses typical SCM activities at project level and includes source control, change control, problem management, etc. An accompanying website contains templates, procedures and other materials to aid understanding and encourage the practical applications of the material discussed throughout

www.wiley.com/go/moreira_software/
Anyone who has to implement SCM in his/her company at every level will need this book and find its practical approach useful

An aspiring business analyst has to go through the rigors of the interview process in order to prove his knowledge, skill, ability, and worth to a prospective employer. The intent of this book is to provide a comprehensive guide to help aspiring as well as experienced business analysts prepare for interviews for suitable roles. The Q&A format of the book seeks to guide readers in planning and organizing their thoughts in a focused and systematic manner. Additionally, this book also aims to not only clarify existing concepts but also help candidates to enhance their understanding of the field. Thus, the book can also be used for preparing for professional certification exams offered by various leading institutes across the globe.

This is the digital version of the printed book (Copyright © 1996). Written in a remarkably clear style, Creating a Software Engineering Culture presents a comprehensive approach to improving

the quality and effectiveness of the software development process. In twenty chapters spread over six parts, Wiegers promotes the tactical changes required to support process improvement and high-quality software development. Throughout the text, Wiegers identifies scores of culture builders and culture killers, and he offers a wealth of references to resources for the software engineer, including seminars, conferences, publications, videos, and on-line information. With case studies on process improvement and software metrics programs and an entire part on action planning (called "What to Do on Monday"), this practical book guides the reader in applying the concepts to real life. Topics include software culture concepts, team behaviors, the five dimensions of a software project, recognizing achievements, optimizing customer involvement, the project champion model, tools for sharing the vision, requirements traceability matrices, the capability maturity model, action planning, testing, inspections, metrics-based project estimation, the cost of quality, and much more! Principles from Part 1 Never let your boss or your customer talk you into doing a bad job. People need to feel the work they do is appreciated. Ongoing education is every team member's responsibility. Customer involvement is the most critical factor in software quality. Your greatest challenge is sharing the vision of the final product with the customer. Continual improvement of your software development process is both possible and essential. Written software development procedures can help build a shared culture of best practices. Quality is the top priority; long-term productivity is a natural consequence of high quality. Strive to have a peer, rather than a customer, find a defect. A key to software quality is to iter-

ate many times on all development steps except coding: Do this once. Managing bug reports and change requests is essential to controlling quality and maintenance. If you measure what you do, you can learn to do it better. You can't change everything at once. Identify those changes that will yield the greatest benefits, and begin to implement them next Monday. Do what makes sense; don't resort to dogma.

This publication is the Project Plan for a community-type society. A societal-level project plan describes the organized thinking and execution of a socio-technical environment; the societal structuring of community. This project plan identifies humanity's project to create a global community-type society for the fulfillment of that which everyone has mutually in common. This is a planned project for a configuration of society that may be tested in its results at optimally meeting all human life requirements at the global scale. This is a planning and work proposal for an open-source, societal-level project. This document describes and explains a unified approach to actions and results that is likely, given what is known and accessible, to improve all of humanity. This is the plan for societal navigation that specifies an approach, direction, and execution to socio-technical life. The project plan has three core sections: (1) Approach to project execution, (2) Direction of project execution, and (3) Execution of project execution. The standard details the complete, plannable information set for the society's operation, including its approach to action, its direction of action, and its execution and adaptation of action. Herein, these concepts, their relationships and understandings, are defined and modeled. Discursive reasoning is provided for this specific configuration of a project plan, as

opposed to the selection and encoding of other configurations. A project plan provides for the formalized project-based development operation of a society, organized in time and with available resources, coordinated to become a societal service system for human fulfillment and ecological well-being.

Policies and procedures are the foundation of internal controls for organizations. Taking a complicated subject and breaking it into manageable components, this book enables you to hit the ground running and significantly accelerate your completion of a solid policies and procedures program. Comprehensive and practical, this useful book provides you with sample documents you can personalize and customize to meet your company's needs.

PMBOK® Guide is the go-to resource for project management practitioners. The project management profession has significantly evolved due to emerging technology, new approaches and rapid market changes. Reflecting this evolution, The Standard for Project Management enumerates 12 principles of project management and the PMBOK® Guide – Seventh Edition is structured around eight project performance domains. This edition is designed to address practitioners' current and future needs and to help them be more proactive, innovative and nimble in enabling desired project outcomes. This edition of the PMBOK® Guide:

- Reflects the full range of development approaches (predictive, adaptive, hybrid, etc.);
- Provides an entire section devoted to tailoring the development approach and processes;
- Includes an expanded list of models, methods, and artifacts;
- Focuses on not just delivering project outputs but also enabling outcomes; and
- Integrates with

PMIstandards+™ for information and standards application content based on project type, development approach, and industry sector.

Make the most of OTS systems in operator training and engineering Key Features Learn OTS project delivery best practices from the author's 30 years of experience Explore use cases to understand how your OTS systems can maximize ROI for users Discover how to best develop OTS training models for developers and users Book Description Operator training simulators in the process industry have been around since the 1970s, but you may not find a book that documents the development of these systems and the standard best practices. The Operator Training Simulator Handbook covers best practices for OTS engineering and OTS training development and delivery, starting from the basic the jargon and the different types of OTS systems. It will take you through the best approaches to project specification as well as building, maintenance, planning, and delivering these systems by sharing real-life experiences and dos and don'ts. As you advance, you'll uncover the various challenges in the planning and delivery of operator training models and understand how to address those by working through real-world projects. This book helps in specifying the best fit for purpose, choosing a cost-effective system when acquiring an OTS. You'll also learn how you can turn your OTS projects into digital twins before finally learning all about documentation in a typical OTS project, covering the sample structure that you can use as a starting point in your projects. By the end of the book, you'll have learned best practices for developing operator training simulator systems and have a reference guide to overcome common challenges. What you will

learn Become familiar with the OTS jargon to set a base for understanding OTS aspects Implement training planning methods that have been tried and tested in the industry for many years Get to grips with writing well-planned documentation for your OTS project Review new model suggestions to maximize benefits of the OTS systems and the actual ICSS control systems to maximize ROI for users Understand Cloud OTS systems as a new way to address some of the common issues that developers and users face Create digital twins of your OTS projects Who this book is for This book is for suppliers who build and deliver OTS systems, OTS buyers, or companies looking to invest in these systems. Anyone with an interest in OTS systems, including university students or graduates who will work on these systems, will find this book useful. Basic knowledge of either OTS systems, ICSS control systems, or process engineering will help you grasp the concepts covered in this book.

A software architecture manifests the major early design decisions, which determine the system's development, deployment and evolution. Thus, making better architectural decisions is one of the large challenges in software engineering. Software architecture knowledge management is about capturing practical experience and translating it into generalized architectural knowledge, and using this knowledge in the communication with stakeholders during all phases of the software lifecycle. This book presents a concise description of knowledge management in the software architecture discipline. It explains the importance of sound knowledge management practices for improving software architecture processes and products, and makes clear the role of knowledge management in software architecture and

software development processes. It presents many approaches that are in use in software companies today, approaches that have been used in other domains, and approaches under development in academia. After an initial introduction by the editors, the contributions are grouped in three parts on "Architecture Knowledge Management", "Strategies and Approaches for Managing Architectural Knowledge", and "Tools and Techniques for Managing Architectural Knowledge". The presentation aims at information technology and software engineering professionals, in particular software architects and software architecture researchers. For the industrial audience, the book gives a broad and concise understanding of the importance of knowledge management for improving software architecture process and building capabilities in designing and evaluating better architectures for their mission- and business-critical systems. For researchers, the book will help to understand the applications of various knowledge management approaches in an industrial setting and to identify research challenges and opportunities.

Software engineering is playing an increasingly significant role in computing and informatics, necessitated by the complexities inherent in large-scale software development. To deal with these difficulties, the conventional life-cycle approaches to software engineering are now giving way to the "process system" approach, encompassing development methods, infrastructure, organization, and management. Until now, however, no book fully addressed process-based software engineering or set forth a fundamental theory and framework of software engineering processes. *Software Engineering Processes: Principles and Appli-*

cations does just that. Within a unified framework, this book presents a comparative analysis of current process models and formally describes their algorithms. It systematically enables comparison between current models, avoidance of ambiguity in application, and simplification of manipulation for practitioners. The authors address a broad range of topics within process-based software engineering and the fundamental theories and philosophies behind them. They develop a software engineering process reference model (SEPRM) to show how to solve the problems of different process domains, orientations, structures, taxonomies, and methods. They derive a set of process benchmarks-based on a series of international surveys-that support validation of the SEPRM model. Based on their SEPRM model and the unified process theory, they demonstrate that current process models can be integrated and their assessment results can be transformed between each other. Software development is no longer just a black art or laboratory activity. It is an industrialized process that requires the skills not just of programmers, but of organization and project managers and quality assurance specialists. Software Engineering Processes: Principles and Applications is the key to understanding, using, and improving upon effective engineering procedures for software development.

From System Designers to Top Management, Everyone loves a good story Once upon a time, it was well understood that stories teach better than plain facts. Why then are most software requirements documents a baffling hodge-podge of diagrams, data dictionaries, and bullet points, held together by little more than a name and a staple? Telling Stories teaches you to combine proven

standards of requirements analysis with the most ancient and effective tool for sharing information, the narrative. Telling Stories simplifies and refines the classic methods of Structured Analysis, providing organization, design, and old-fashioned writing advice. Whether you're just getting started or an experienced requirements writer, Telling Stories can help you turn dull, detailed material into an engaging, logical, and readable story, a story that can make the difference for your project and your career. Learn why readers believe and remember what they learn from stories Work with team members to gather content, tell their stories, and win their support Use stories to find every requirement Create diagrams that almost tell the story on their own (while looking clear and professional) Explain everything important about a process Use precise language to remove the ambiguity from requirements Write a forceful executive summary that stands on its own and sells a project to senior management Summarize often to keep the reader focused on key issues Structure the document so every part has a clear place and purpose

The seasoned programmer and novice alike find this reference the ideal resource for getting a project off to the right start. Friendly, practical advice is combined with the latest software in this ...For Dummies edition. Follow your expert guide through planning, development, testing, and implementation -- the first steps to your project's success. Then get your hands on scheduling, assigning resources and estimating costs, and best of all, making your software happen. The book's CD-ROM includes trial versions of Microsoft Project 2000, Soffrant TRACK, and Cost Xpert as well as templates and a wealth of other plann-

ing tools.

Change is the law of life. This precept holds for individuals and organizations alike. Radical change, called transformation, is not so frequent. Organizational transformation entails fundamental changes that encompass the entire gamut of the organization. A successful transformation exercise begins with formulating a set of Objectives at the beginning and ends up with realizing the desired Outcomes. This book delves deep into the intricacies of the life cycle of organizational transformation and explains how it can be managed effectively adopting a systematic and step-by-step methodology. In this well-researched text, the author, J. Satyanarayana, with his vast and varied experience in the field of management and governance, gives a brilliant exposition of managing transformation. The book postulates the O2O (Objectives to Outcomes) framework as a seven-step methodology. The methodology, in a sense, is a meta-framework for transformation. It harmonizes and synthesizes altogether 40 sub-frameworks, which include some well-known concepts, such as Porter's Value Chain Framework, Kaplan and Norton's Balanced Scorecard, and Hammer's BPR framework. The author illustrates the O2O framework through a Case Study on Passport Seva Project, designed by him, and being implemented all over India to transform the quality of passport-related services provided to the citizens. This text, with its blend of theory and practice, would prove extremely valuable to the students of management and commerce. It should also be a valuable reference book to the administrators and management consultants. **KEY FEATURES :** By expounding the O2O methodology, the book addresses the managerial requirements of conceptualizing, designing and imple-

menting major transformation projects. Focuses on the four principal dimensions of transformation, namely, Process, People, Technology, and Business Model while explaining the major concepts. Interspersed with plenty of examples to illustrate the concepts. Includes chapter-end review questions to drill the students in self-study.

The Quality Special Interest Group of the British Computer Society presents the edited proceedings of their sixth International Conference on Software Quality Management (SQM'98) held in April 1998 in Amsterdam. The objective of this series of annual conferences is to promote international co-operation among those concerned with software quality and process improvement, by creating a greater understanding of software quality issues and by sharing current research and industrial experience. The papers cover a broad spectrum of practical experience and research. The topic areas include process improvement, maintaining a quality management system, quality metrics, human factors, project management issues, software tools and approaches to systems development. The organisers would like to thank Origin for their sponsorship of the proceedings. The editors are indebted to the members of the International Advisory Committee for their support and for refereeing the abstracts and the final papers, as well as to the authors who have contributed to the success of this conference.

A flat organization believes the formal processes and controls used by many hierarchical organizations are too involved, require too much overhead cost, and are too complex and/or time consuming. Project Management for Flat Organizations provides common sense solutions to the unique challenges of organi-

zations with flat hierarchical structures. It explains project management theory and offers simple and cost effective project management processes, tools, and techniques that can be applied immediately. This guide includes instruction and templates required to deliver projects efficiently and successfully with minimal risk and investment. It also enables users to develop a framework specific to the needs of their organization. This is a go-to guide you will want to keep on your desk for easy reference when working on projects. This book is ideal for the project manager, team member, manager, or project sponsor with limited or no formal project management experience working within a flat organization. It offers clear, understandable discussions about project management processes; practical ideas and suggestions; answers common questions; and explains ways to address common pitfalls.

The increase in project outsourcing has forced traditional programmers to take on the role of project managers and quickly learn how to manage software projects. The author discusses all of the essentials in widely accepted project management methodology, from managing programmers to assessing and eliminating risk. The book covers the iterative development model, using Microsoft Project 2003, as well as a variety of methodologies including eXtreme, open source, SQA testing, software life cycle management, and more. The companion Web site contains tools, case studies and other resources to help even novices get up and running.

Discover exciting behind-the-scenes opportunities and challenges in technology today with Schwalbe's unique INFORMATION TECHNOLOGY PROJECT MANAGEMENT, REVISED 7E. This one-of-a-kind book demonstrates the principles distinc-

tive to managing information technology (IT) projects that extend well beyond standard project management requirements. No book offers more up-to-the-minute insights and software tools for IT project management success, including updates that reflect the latest PMBOK Guide, 5th edition, the global standard for managing projects and earning certification. The book weaves today's theory with successful practices for an understandable, integrated presentation that focuses on the concepts, tools, and techniques that are most effective today. INFORMATION TECHNOLOGY PROJECT MANAGEMENT is the only book to apply all ten project management knowledge areas to IT projects. You master skills in project integration, scope, time, cost, quality, human resource, communications, risk, procurement, and stakeholder management as well as all five process groups--initiating, planning, executing, monitoring and controlling, and closing. Intriguing examples from familiar companies featured in today's news, a new Agile case, opportunities with MindView software, and a new chapter on project stakeholder management further ensure you are equipped to manage information technology projects with success. The REVISED Seventh Edition has updated Appendix A for Microsoft Project 2013. Important Notice: Media content referenced within the product description or the product text may not be available in the ebook version.

Project managers, sponsors, team members, and involved stakeholders know when things aren't going well. A frequent first indication is a missing or errant process. Project Health Assessment presents an innovative approach for assessing project processes through a set of ten critical success factors based on

PMI's PMBOK® Guide knowledge areas. The findings from such assessments can help project managers reduce project risk, improve stakeholder satisfaction, and increase the likelihood of project success, as demonstrated by 30+ assessments done over 15 years of putting this approach into practice. Project Health Assessment breaks down each PMBOK® Guide knowledge area into its process steps, inputs, and outputs and then creates critical success factor questions that evaluate its effectiveness and potential risk. These questions can be used by project managers to establish sufficient project processes or by external entities to evaluate a project and assess its overall risk. The book illustrates critical success factor points through numerous case studies, including a step-by-step example of how to conduct a project health assessment from engagement acquisition through startup, initial assessment, and periodic follow-up assessments. The book provides several downloadable document, spreadsheet, and scheduling templates that practitioners can customize and use in their projects. Using these tools, you can avoid or minimize the cost of failed projects to your organization.

It is often assumed that software testing is based on clearly defined requirements and software development standards. However, testing is typically performed against changing, and sometimes inaccurate, requirements. The third edition of a bestseller, *Software Testing and Continuous Quality Improvement, Third Edition* provides a continuous quality framework for the software testing process within traditionally structured and unstructured environments. This framework aids in creating meaningful test cases for systems with evolving requirements. This completely revised reference provides a comprehensive look at software testing

as part of the project management process, emphasizing testing and quality goals early on in development. Building on the success of previous editions, the text explains testing in a Service Oriented Architecture (SOA) environment, the building blocks of a Testing Center of Excellence (COE), and how to test in an agile development. Fully updated, the sections on test effort estimation provide greater emphasis on testing metrics. The book also examines all aspects of functional testing and looks at the relation between changing business strategies and changes to applications in development. Includes New Chapters on Process, Application, and Organizational Metrics. All IT organizations face software testing issues, but most are unprepared to manage them. *Software Testing and Continuous Quality Improvement, Third Edition* is enhanced with an up-to-date listing of free software tools and a question-and-answer checklist for choosing the best tools for your organization. It equips you with everything you need to effectively address testing issues in the most beneficial way for your business.

Each and every chapter covers the contents up to a reasonable depth necessary for the intended readers in the field. The book consists in all about 1200 exercises based on the topics and sub-topics covered. Keeping in view the emerging trends in newly emerging scenario with new dimension of software engineering, the book specially includes the following chapters, but not limited to these only. This book explains all the notions related to software engineering in a very systematic way, which is of utmost importance to the novice readers in the field of software Engineering.

Practical Support for Lean Six Sigma Software Process Definition: Using IEEE Software Engineering Standards addresses

the task of meeting the specific documentation requirements in support of Lean Six Sigma. This book provides a set of templates supporting the documentation required for basic software project control and management and covers the integration of these templates for their entire product development life cycle. Find detailed documentation guidance in the form of organizational policy descriptions, integrated set of deployable document templates, artifacts required in support of assessment, organizational delineation of process documentation.

This book constitutes the proceedings of the 16th International Conference on Relational and Algebraic Methods in Computer Science, RAMiCS 2017, held in Lyon, France, in May 2017. The 17 revised full papers and 2 invited papers presented together with 1 invited abstract were carefully selected from 28 submissions. Topics covered range from mathematical foundations to applications as conceptual and methodological tools in computer science and beyond.

A game design document (GDD) is a software design document that serves as a blueprint from which your game is to be built. It helps you define the scope of your game and sets the general direction for the project, keeping the entire team on the same page. This is a companion piece, intended to accompany the Lazy Designer series. There are two sections -- a sample planning document followed by a "how it went" discussion on the actual implementation.

Develop a strong understanding of IT project management as you learn to apply today's most effective project management tools and techniques with the unique approach found in Schwalbe's INFORMATION TECHNOLOGY PROJECT MANAGEMENT, 9E. Examine the latest devel-

opments and skills as you prepare for the Project Management Professional (PMP) or Certified Associate in Project Management (CAPM) exams. This edition reflects content from the latest PMBOK Guide, 6E and the Agile Practice Guide while providing a meaningful context for understanding project management. Hundreds of timely examples highlight IT projects, while discussion, exercises and cases reinforce learning. Examples from familiar companies featured in today's news, and a guide to using Microsoft Project 2016 help you master IT project management skills that are marketable across the globe. Important Notice: Media content referenced within the product description or the product text may not be available in the ebook version. Discusses ways to run meetings effectively and efficiently.

Now in its third edition, this classic guide to software requirements engineering has been fully updated with new topics, examples, and guidance. Two leaders in the requirements community have teamed up to deliver a contemporary set of practices covering the full range of requirements development and management activities on software projects. Describes practical, effective, field-tested techniques for managing the requirements engineering process from end to end. Provides examples demonstrating how requirements "good practices" can lead to fewer change requests, higher customer satisfaction, and lower development costs. Fully updated with contemporary examples and many new practices and techniques. Describes how to apply effective requirements practices to agile projects and numerous other special project situations. Targeted to business analysts, developers, project managers, and other software project stake-

holders who have a general understanding of the software development process. Shares the insights gleaned from the authors' extensive experience delivering hundreds of software-requirements training courses, presentations, and webinars. New chapters are included on specifying data requirements, writing high-quality functional requirements, and requirements reuse. Considerable depth has been added on business requirements, elicitation techniques, and non-functional requirements. In addition, new chapters recommend effective requirements practices for various special project situations, including enhancement and replacement, packaged solutions, outsourced, business process automation, analytics and reporting, and embedded and other real-time systems projects.

Agile software development approaches have had significant impact on industrial software development practices. Today, agile software development has penetrated to most IT companies across the globe, with an intention to increase quality, productivity, and profitability. Comprehensive knowledge is needed to understand the architectural challenges involved in adopting and using agile approaches and industrial practices to deal with the development of large, architecturally challenging systems in an agile way. Agile Software Architecture focuses on gaps in the requirements of applying architecture-centric approaches and principles of agile software development and demystifies the agile architecture paradox. Readers will learn how agile and architectural cultures can co-exist and support each other according to the context. Moreover, this book will also provide useful leads for future research in architecture and agile to bridge such gaps by developing appropriate approaches that in-

corporate architecturally sound practices in agile methods. Presents a consolidated view of the state-of-art and state-of-practice as well as the newest research findings. Identifies gaps in the requirements of applying architecture-centric approaches and principles of agile software development and demystifies the agile architecture paradox. Explains whether or not and how agile and architectural cultures can co-exist and support each other depending upon the context. Provides useful leads for future research in both architecture and agile to bridge such gaps by developing appropriate approaches, which incorporate architecturally sound practices in agile methods.

A classic treatise that defined the field of applied demand analysis, *Consumer Demand in the United States: Prices, Income, and Consumption Behavior* is now fully updated and expanded for a new generation. Consumption expenditures by households in the United States account for about 70% of America's GDP. The primary focus in this book is on how households adjust these expenditures in response to changes in price and income. Econometric estimates of price and income elasticities are obtained for an exhaustive array of goods and services using data from surveys conducted by the Bureau of Labor Statistics, providing a better understanding of consumer demand. Practical models for forecasting future price and income elasticities are also demonstrated. Fully revised with over a dozen new chapters and appendices, the book revisits the original Taylor-Houthakker models while examining new material as well, such as the use of quantile regression and the stationarity of consumer preference. It also explores the emerging connection between neuroscience and consumer behavior, integrat-

ing the economic literature on demand theory with psychology literature. The most comprehensive treatment of the topic to date, this volume will be an essential resource for any researcher, student or professional economist working on consumer behavior or demand theory, as well as investors and policymakers concerned with the impact of economic fluctuations.

"If you're looking for solid, easy-to-follow advice on estimation, requirements gathering, managing change, and more, you can stop now: this is the book for you."--Scott Berkun, Author of *The Art of Project Management*

What makes software projects succeed? It takes more than a good idea and a team of talented programmers. A project manager needs to know how to guide the team through the entire software project. There are common pitfalls that plague all software projects and rookie mistakes that are made repeatedly--sometimes by the same people! Avoiding these pitfalls is not hard, but it is not necessarily intuitive. Luckily, there are tried and true techniques that can help any project manager. In *Applied Software Project Management*, Andrew Stellman and Jennifer Greene provide you with tools, techniques, and practices that you can use on your own projects right away. This book supplies you with the information you need to diagnose your team's situation and presents practical advice to help you achieve your goal of building better software. Topics include: Planning a software project Helping a team estimate its workload Building a schedule Gathering software requirements and creating use cases Improving programming with refactoring, unit testing, and version control Managing an outsourced project Testing software Jennifer Greene and Andrew Stellman have been building

software together since 1998. Andrew comes from a programming background and has managed teams of requirements analysts, designers, and developers. Jennifer has a testing background and has managed teams of architects, developers, and testers. She has led multiple large-scale outsourced projects. Between the two of them, they have managed every aspect of software development. They have worked in a wide range of industries, including finance, telecommunications, media, nonprofit, entertainment, natural-language processing, science, and academia. For more information about them and this book, visit stellman-greene.com

Zero in on key project-initiation tasks—and build a solid foundation for successful software development. In this concise guide, critically-acclaimed author Karl E. Wiegers fills a void in project management literature by focusing on the activities that are essential—but often overlooked—for launching any project. Drawing on his extensive experience, Karl shares lessons learned, proven practices, and tools for getting your project off to the right start—and steering it to ultimate success. Lay a foundation for project success—discover how to: Effectively charter a project Define meaningful criteria for project success and product releases Negotiate achievable commitments for project teams and stakeholders Identify and document potential barriers to success—and manage project risks Apply the Wideband Delphi method for more accurate estimation Measure project performance and avoid common metrics traps Systematically apply lessons learned to future projects Companion Web site includes: Worksheets from inside the book Project document templates Resources for pro-

ject initiation and process improvement. This book provides the software engineering fundamentals, principles and skills needed to develop and maintain high quality software products. It covers requirements specification, design, implementation, testing and management of software projects. It is aligned with the SWEBOK, Software Engineering Undergraduate Curriculum Guidelines and ACM Joint Task Force Curricula on Computing. This classroom-tested new edition features expanded coverage of the basics and test automation frameworks, with new exercises and examples.

Learn how to: § Select the best ERP software for your organization § Choose the most effective wrap around software to enhance the performance of an existing ERP system § Align software selection with business goals and objectives § Budget for the software and the hidden costs involved in its implementation. At times a daring, maddening, and even frightening process, finding and implementing a suitable software package is never an easy task. The cost of the software package is often a fraction of the overall expense. Unless carefully selected, a major software package implementation can consume a considerable amount of your organization's time and energy. An ill-informed purchase can cost your organization its customers, dollars, and reputation. *Maximizing Business Performance through Software Packages: Best Practices for Justification, Selection, and Implementation* explores the business challenges involved in justifying, selecting, and implementing software packages. It contains practical advice and insights on how to select "good fitting" software packages, how to justify them in terms of their ability to enable business process change or improvement, and most importantly, how to implement them suc-

cessfully. Selecting and implementing enterprise architecture technology software solutions involves a large expenditure across all the resources of an organization. The process has become increasingly complex as business functions have become increasingly integrated. *Maximizing Business Performance through Software Packages: Best Practices for Justification, Selection, and Implementation* provides a definitive source that will help you select the solutions that best fit your business needs.

Incomplete or missed requirements, omissions, ambiguous product features, lack of user involvement, unrealistic customer expectations, and the proverbial scope creep can result in cost overruns, missed deadlines, poor product quality, and can very well ruin a project. *Project Scope Management: A Practical Guide to Requirements for Engineering, Product, Construction, IT and Enterprise Projects* describes how to elicit, document, and manage requirements to control project scope creep. It also explains how to manage project stakeholders to minimize the risk of an ever-growing list of user requirements. The book begins by discussing how to collect project requirements and define the project scope. Next, it considers the creation of work breakdown structures and examines the verification and control of the scope. Most of the book is dedicated to explaining how to collect requirements and how to define product and project scope inasmuch as they represent the bulk of the project scope management work undertaken on any project regardless of the industry or the nature of the work involved. The book maintains a focus on practical and sensible tools and techniques rather than academic theories. It examines five different projects and traces their devel-

opment from a project scope management perspective—from project initiation to the end of the execution and control phases. The types of projects considered include CRM system implementation, mobile number portability, port upgrade, energy-efficient house design, and airport check-in kiosk software. After reading this book, you will learn how to create project charters, high-level scope, detailed requirements specifications, re-

quirements management plans, traceability matrices, and a work breakdown structure for the projects covered.

This book is a comprehensive, step-by-step guide to software engineering. This book provides an introduction to software engineering for students in undergraduate and post graduate programs in computers.

Project management software.