Get Free Agile Model Based Systems Engineering Cookbook

Thank you unquestionably much for downloading **Agile Model Based Systems Engineering Cookbook**.Maybe you have knowledge that, people have see numerous times for their favorite books later than this Agile Model Based Systems Engineering Cookbook, but stop going on in harmful downloads.

Rather than enjoying a fine book taking into account a cup of coffee in the afternoon, then again they juggled taking into consideration some harmful virus inside their computer. **Agile Model Based Systems Engineering Cookbook** is user-friendly in our digital library an online entry to it is set as public hence you can download it instantly. Our digital library saves in combination countries, allowing you to get the most less latency era to download any of our books subsequent to this one. Merely said, the Agile Model Based Systems Engineering Cookbook is universally compatible following any devices to read.

KEY=SYSTEMS - CAREY CASSIUS

Agile Model-Based Systems Engineering Cookbook Improve system development by applying proven recipes for effective agile systems engineering Packt Publishing Ltd **The Agile Model-Based Systems Engineering Cookbook distills the most relevant MBSE workflows and work products into a set of easy-to-follow recipes, complete with examples of their application. This book serves as a quick and reliable practical reference for systems engineers looking to apply agile MBSE to real-world projects. AGILE MODEL-BASED SYSTEMS ENGINEERING COOKBOOK - Improve System Development by Applying Proven... Recipes for Effective Agile Systems Engineering Agile Systems Engineering** Morgan Kaufmann **Agile Systems Engineering presents a vision of systems engineering where precise specification of requirements, structure, and behavior meet larger concerns as such as safety, security, reliability, and performance in an agile engineering context. World-renown author and speaker Dr. Bruce Powel Douglass incorporates agile methods and model-based systems engineering (MBSE) to define the properties of entire systems while avoiding errors that can occur when using traditional textual specifications. Dr. Douglass covers the lifecycle of systems development, including requirements, analysis, design, and the handoff to specific engineering disciplines. Throughout, Dr. Douglass couples agile methods with SysML and MBSE to arm system engineers with the conceptual and methodological tools they need to avoid specification defects and improve system quality while simultaneously reducing the effort and cost of systems engineering. Identifies how the concepts and techniques of agile methods can be effectively applied in systems engineering context Shows how to perform model-based functional analysis and tie these analyses back to system requirements and stakeholder needs, and forward to system architecture and interface definition Provides a means by which the quality and correctness of systems engineering data can be assured (before the entire system is built!) Explains agile system architectural specification and allocation of functionality to system components Details how to transition engineering specification data to downstream engineers with no loss of fidelity Includes detailed examples from across industries taken through their stages, including the "Waldo" industrial exoskeleton as a complex system Effective Model-Based Systems Engineering** Springer **This textbook presents a proven, mature Model-Based Systems Engineering (MBSE) methodology that has delivered success in a wide range of system and enterprise programs. The authors introduce MBSE as the state of the practice in the vital Systems Engineering discipline that manages complexity and integrates technologies and design approaches to achieve effective, affordable, and balanced system solutions to the needs of a customer organization and its personnel. The book begins with a summary of the background and nature of MBSE. It summarizes the theory behind Object-Oriented Design applied to complex system architectures. It then walks through the phases of the MBSE methodology, using system examples to illustrate key points. Subsequent chapters broaden the application of MBSE in Service-Oriented Architectures (SOA), real-time systems, cybersecurity, networked enterprises, system simulations, and prototyping. The vital subject of system and architecture governance completes the discussion. The book features exercises at the end of each chapter intended to help readers/students focus on key points, as well as extensive appendices that furnish additional detail in particular areas. The self-contained text is ideal for students in a range of courses in systems architecture and MBSE as well as for practitioners seeking a highly practical presentation of MBSE principles and techniques. Model Based Systems Engineering Fundamentals and Methods** John Wiley & Sons **This book is a contribution to the definition of a model based system engineering (MBSE) approach, designed to meet the objectives laid out by the INCOSE. After pointing out the complexity that jeopardizes a lot of system developments, the book examines fundamental aspects of systems under consideration. It goes on to address methodological issues and proposes a methodic approach of MBSE that provides, unlike current practices, systematic and integrated model-based engineering processes. An annex describes relevant features of the VHDL-AMS language supporting the methodological issues described in the book. Real-Time Agility The Harmony/ESW Method for Real-Time and Embedded Systems Development** Pearson Education **Real-time and embedded systems face the same development challenges as traditional software: shrinking budgets and shorter timeframes. However, these systems can be even more difficult to successfully develop due to additional requirements for timeliness, safety, reliability, minimal resource use, and, in some cases, the need to support rigorous industry standards. In Real-Time Agility, leading embedded-systems consultant Bruce Powel Douglass reveals how to leverage the best practices of agile development to address all these challenges. Bruce introduces the Harmony/ESW process: a proven, start-to-finish approach to software development that can reduce costs, save time, and eliminate potential defects. Replete with examples, this book provides an ideal tutorial in agile methods for real-time and embedded-systems developers. It also serves as an invaluable “in the heat of battle” reference guide for developers working to advance projects, both large and small. Coverage includes How Model-Driven Development (MDD) and agile methods work synergistically The Harmony/ESW process, including roles, workflows, tasks, and work products Phases in the Harmony/ESW microcycle and their implementation Initiating a real-time agile project, including the artifacts you may (or may not) need Agile analysis, including the iteration plan, clarifying requirements, and validation The three levels of agile design: architectural, mechanistic, and detailed Continuous integration strategies and end-of-the-microcycle validation testing How Harmony/ESW’s agile process self-optimizes by identifying and managing issues related to schedule, architecture, risks, workflows, and the process itself A Primer for Model-Based Systems Engineering** Lulu.com **This primer addresses the basic concepts of model-based systems engineering. It covers the Model, Language, Behavior, Process, Architecture, and Verification and Validation. It is a call to consider the foundational principles behind those concepts. It is not designed to present novel insights into MBSE so much as to provide a guided tour of the touchstones of systems design. It is a guide to the new MBSE acolyte and a reminder to the experienced practitioner. It is our hope that you find this primer valuable. We welcome your comments and suggestions about improving it. Much of what we have learned about how it should be organized and presented has come from thoughtful contributions from the readers of the 1st edition. Systems Engineering Demystified A practitioner's handbook for developing complex systems using a model-based approach** Packt Publishing Ltd **Get to grips with systems engineering life cycles, processes, and best practices and discover techniques to successfully develop complex systems Key FeaturesDiscover how to manage increased complexity and understand systems better via effective communicationAdopt a proven model-based approach for systems engineering in your organizationApply proven techniques for requirements, design, validation and verification, and systems engineering managementBook Description Systems engineering helps us to understand, specify, and develop complex systems, and is applied across a wide set of disciplines. As systems and their associated problems become increasingly complex in this evermore connected world, the need for more rigorous, demonstrable, and repeatable techniques also increases. Written by Professor Jon Holt – an internationally recognized systems engineering expert – this book provides a blend of technical and business aspects you need to understand in order to develop successful systems. You'll start with systems engineering basics and understand the complexity, communication, and different stakeholders' views of the system. The book then covers essential aspects of model-based systems engineering, systems, life cycles, and processes, along with techniques to develop systems. Moving on, you'll explore system models and visualization techniques, focusing on the SysML, and discover how solutions can be defined by developing effective system design, verification, and validation techniques. The book concludes by taking you through key management processes and systems engineering best practices and guidelines. By the end of this systems engineering book, you'll be able to confidently apply modern model-based systems engineering techniques to your own systems and projects. What you will learnUnderstand the three evils of systems engineering - complexity, ambiguous communication, and lack of understandingRealize successful systems using model-based systems engineeringUnderstand the concept of life cycles and how they control the evolution of a systemExplore processes and related concepts such as activities, stakeholders, and resourcesDiscover how needs fit into the systems life cycle and which processes are relevant and how to comply with themFind out how design, verification, and validation fit into the life cycle and processesWho this book is for This book is for aspiring systems engineers, engineering managers, or anyone looking to apply systems engineering practices to their systems and projects. While a well-structured, model-based approach to systems engineering is an essential skill for engineers of all disciplines, many companies are finding that new graduates have little understanding of systems engineering. This book helps you acquire this skill with the help of a simple and practical approach to developing successful systems. No prior knowledge of systems engineering or modeling is required to get started with this book. Verification and Validation in Systems Engineering Assessing UML/SysML Design Models** Springer Science & Business Media **At the dawn of the 21st century and the information age, communication and c-puting power are becoming ever increasingly available, virtually pervading almost every aspect of modern socio-economical interactions. Consequently, the potential for realizing a signi?cantly greater number of technology-mediated activities has emerged. Indeed, many of our modern activity ?elds are heavily dependant upon various underlying systems and software-intensive platforms. Such technologies are commonly used in everyday activities such as commuting, traf?c control and m- agement, mobile computing, navigation, mobile communication. Thus, the correct function of the forenamed computing systems becomes a major concern. This is all the more important since, in spite of the numerous updates, patches and ?rmware revisions being constantly issued, newly discovered logical bugs in a wide range of modern software platforms (e. g. , operating systems) and software-intensive systems (e. g. , embedded systems) are just as frequently being reported. In addition, many of today’s products and services are presently being deployed in a highly competitive environment wherein a product or service is succeeding in most of the cases thanks to its quality to price ratio for a given set of features. Accordingly, a number of critical aspects have to be considered, such as the ab- ity to pack as many features as needed in a given product or service while c- currently maintaining high quality, reasonable price, and short time -to- market. System Requirements Engineering A SysML Supported Requirements Engineering Method** John Wiley & Sons **Practical Model-Based Systems Engineering** Artech House **This comprehensive resource provides systems engineers and practitioners with the analytic, design and modeling tools of the Model-Based Systems Engineering (MBSE) methodology of Integrated Systems Engineering (ISE) and Pipelines of**

Processes in Object Oriented Architectures (PPOOA) methodology. This methodology integrates model based systems and software engineering approaches for the development of complex products, including aerospace, robotics and energy domains applications. Readers learn how to synthesize physical architectures using design heuristics and trade-off analysis. The book provides information about how to identify, classify and specify the system requirements of a new product or service. Using Systems Modeling Language (SysML) constructs, readers will be able to apply ISE & PPOOA methodology in the engineering activities of their own systems. Writing Effective Use Cases Pearson Education This guide will help readers learn how to employ the significant power of use cases to their software development efforts. It provides a practical methodology, presenting key use case concepts. SYSMOD - The Systems Modeling Toolbox - Pragmatic MBSE with SysML Lulu.com SYSMOD is an MBSE toolbox for pragmatic modeling of systems. It is well-suited to be used with SysML. The book provides a set of methods with roles and outputs. Concrete guidances and examples show how to apply the methods with SysML. * Requirements modeling * System Context * Use Cases * Functional, Physical, Logical and Product Architectures * Guidances how to create a SysML model * Full-fledged SysML example * Complete definition of a profile for SYSMOD This book is also available as an eBook at leanpub.com/sysmod. INCOSE Systems Engineering Handbook A Guide for System Life Cycle Processes and Activities John Wiley & Sons A detailed and thorough reference on the discipline and practice of systems engineering The objective of the International Council on Systems Engineering (INCOSE) Systems Engineering Handbook is to describe key process activities performed by systems engineers and other engineering professionals throughout the life cycle of a system. The book covers a wide range of fundamental system concepts that broaden the thinking of the systems engineering practitioner, such as system thinking, system science, life cycle management, specialty engineering, system of systems, and agile and iterative methods. This book also defines the discipline and practice of systems engineering for students and practicing professionals alike, providing an authoritative reference that is acknowledged worldwide. The latest edition of the INCOSE Systems Engineering Handbook: Is consistent with ISO/IEC/IEEE 15288:2015 Systems and software engineering—System life cycle processes and the Guide to the Systems Engineering Body of Knowledge (SEBoK) Has been updated to include the latest concepts of the INCOSE working groups Is the body of knowledge for the INCOSE Certification Process This book is ideal for any engineering professional who has an interest in or needs to apply systems engineering practices. This includes the experienced systems engineer who needs a convenient reference, a product engineer or engineer in another discipline who needs to perform systems engineering, a new systems engineer, or anyone interested in learning more about systems engineering. SysML Distilled A Brief Guide to the Systems Modeling Language Pearson Education SysML Distilled is a go-to reference for everyone who wants to start creating accurate and useful system models with SysML. Drawing on his pioneering experience creating models for Lockheed Martin and NASA, Lenny Delligatti illuminates SysML's core components, and shows how to use them even under tight deadlines and other constraints. The reader needn't know all of SysML to create effective models: SysML Distilled quickly teaches what does need to be known, and helps deepen the reader's knowledge incrementally as the need arises. SysML in Action with Cameo Systems Modeler Elsevier System engineering (SE) using models (MBSE) is currently in vogue in the community of SE practitioners, whether they are analysts, architects, developers or testers. INCOSE has contributed greatly to the definition of a language for the community, henceforth standardized under ISO-19514: SysML. However, this language is not associated by default with any particular MBSE procedure. This is a major difficulty hampering its implementation. In order to overcome this difficulty, this book describes, in addition to the SysML notation, a generic approach based on the main principles of SE and relative standards, serving as the basis for a specific MBSE approach to be built. This is in order to respond to the specificities of the field of projects in which the practitioners evolve. In order to carry out the procedure in a pragmatic way, a simplified but realistic example serves as a guideline from the initial requirements to the validation of the system, putting into action the SysML modeling tool Cameo Systems Modeler by No Magic. Based on a realistic example and simplified, yet still useful for professionals (no ATM or traffic lights) Explores everything from requirements to validation to cover the classical V cycle Utilizes a generic approach, fully suitable to SysML, to apply major system engineering principles and standards Helps users learn to make their own model by transcribing their needs and taking advantage of the tool features, Conserves time by using recommended workarounds to develop custom processes for this tool, before deploying successfully on real industrial projects Model-Based Engineering of Collaborative Embedded Systems Extensions of the SPES Methodology Springer Nature This Open Access book presents the results of the "Collaborative Embedded Systems" (CrESt) project, aimed at adapting and complementing the methodology underlying modeling techniques developed to cope with the challenges of the dynamic structures of collaborative embedded systems (CESs) based on the SPES development methodology. In order to manage the high complexity of the individual systems and the dynamically formed interaction structures at runtime, advanced and powerful development methods are required that extend the current state of the art in the development of embedded systems and cyber-physical systems. The methodological contributions of the project support the effective and efficient development of CESs in dynamic and uncertain contexts, with special emphasis on the reliability and variability of individual systems and the creation of networks of such systems at runtime. The project was funded by the German Federal Ministry of Education and Research (BMBF), and the case studies are therefore selected from areas that are highly relevant for Germany’s economy (automotive, industrial production, power generation, and robotics). It also supports the digitalization of complex and transformable industrial plants in the context of the German government's "Industry 4.0" initiative, and the project results provide a solid foundation for implementing the German government's high-tech strategy "Innovations for Germany" in the coming years. A Practical Guide to SysML The Systems Modeling Language Morgan Kaufmann A Practical Guide to SysML: The Systems Modeling Language is a comprehensive guide to SysML for systems and software engineers. It provides an advanced and practical resource for modeling systems with SysML. The source describes the modeling language and offers information about employing SysML in transitioning an organization or project to model-based systems engineering. The book also presents various examples to help readers understand the OMG Systems Modeling Professional (OCSMP) Certification Program. The text is organized into four parts. The first part provides an overview of systems engineering. It explains the model-based approach by comparing it with the document-based approach and providing the modeling principles. The overview of SYsML is also discussed. The second part of the book covers a comprehensive description of the language. It discusses the main concepts of model organization, parametrics, blocks, use cases, interactions, requirements, allocations, and profiles. The third part presents examples that illustrate how SysML supports different model-based procedures. The last part discusses how to transition and deploy SysML into an organization or project. It explains the integration of SysML into a systems development environment. Furthermore, it describes the category of data that are exchanged between a SysML tool and other types of tools, and the types of exchange mechanisms that can be used. It also covers the criteria that must be considered when selecting a SysML. Software and systems engineers, programmers, IT practitioners, experts, and non-experts will find this book useful. *The authoritative guide for understanding and applying SysML *Authored by the foremost experts on the language *Language description, examples, and quick reference guide included The Art of Systems Engineering A How-To Guide for Systems Engineers Rjm This text is designed to provide a step-by-step development methodology for systems engineering. The text will allow those not familiar with the domain to work through examples and concepts, enabling them to become adept at the tools and methodologies of the systems engineering domain. This text is the only known publication that provides a how-to approach to the challenging topic of systems engineering. Scaling Agile with Jira Align A practical guide to strategically scaling agile across teams, programs, and portfolios in enterprises Packt Publishing Ltd Accelerate business value delivery with Jira Align, the enterprise agile planning platform, by connecting strategy with execution to maximize outcomes Key FeaturesImprove coordination and transparency between multiple programs, products, and business portfoliosIncrease customer satisfaction by responding quickly to ever-evolving customer needsDeliver higher quality products faster and more predictably with real-time insights and OKR trackingBook Description Jira Align is a platform purpose-built for enterprises to connect strategy with execution and drive transparency, consistency, and predictability at all levels of scale. The platform supports business value delivery in agile frameworks such as LeSS, DAD, and SAFe. It also caters to organizations that mix agile with waterfall to support scaled bimodal delivery. Starting with an introduction to the platform and its features, this book takes you through the foundational building blocks of Jira Align. You’ll learn how an organization can benefit from implementing Jira Align and understand how to connect dimensions such as people, work, time, and outcomes. The book takes you through the typical steps for implementing Jira Align for maximizing outcomes and helps you solve common team, program, and portfolio-level challenges by enhancing visibility, tracking dependencies and risks, and using reports for real-time, distributed decision making. Throughout the book, you’ll explore features such as remote agile ceremonies, live roadmaps, and objectives and key results (OKRs). You’ll also get to grips with lean portfolio management, financial reporting, and using the program board for planning and execution. By the end of this book, you’ll be well versed in the key features of Jira Align and be able to leverage them to support all levels of agile at scale. What you will learnUnderstand Jira Align’s key factors for successFind out how you can connect people, work, time, and outcomes with Jira AlignNavigate and collaborate in Jira AlignScale team agility to the portfolio and enterpriseDelve into planning and execution, including roadmaps and predictability metricsImplement lean portfolio management and OKRsGet to grips with handling bimodal and hybrid deliveryEnable advanced data security and analytics in Jira AlignWho this book is for This book is for portfolio managers, program managers, product managers, product owners, executives, release train engineers, and scrum masters who want to empower their teams to deliver the right things at the right time and quickly respond to changes in the market. Familiarity with agile frameworks and Jira Software is necessary; the book will teach you the rest. 97 Things Every Cloud Engineer Should Know "O'Reilly Media, Inc." If you create, manage, operate, or configure systems running in the cloud, you're a cloud engineer--even if you work as a system administrator, software developer, data scientist, or site reliability engineer. With this book, professionals from around the world provide valuable insight into today's cloud engineering role. These concise articles explore the entire cloud computing experience, including fundamentals, architecture, and migration. You'll delve into security and compliance, operations and reliability, and software development. And examine networking, organizational culture, and more. You're sure to find 1, 2, or 97 things that inspire you to dig deeper and expand your own career. "Three Keys to Making the Right Multicloud Decisions," Brendan O'Leary "Serverless Bad Practices," Manases Jesus Galindo Bello "Failing a Cloud Migration," Lee Atchison "Treat Your Cloud Environment as If It Were On Premises," Iyana Garry "What Is Toil, and Why Are SREs Obsessed with It?", Zachary Nickens "Lean QA: The QA Evolving in the DevOps World," Theresa Neate "How Economies of Scale Work in the Cloud," Jon Moore "The Cloud Is Not About the Cloud," Ken Corless "Data Gravity: The Importance of Data Management in the Cloud," Geoff Hughes "Even in the Cloud, the Network Is the Foundation," David Murray "Cloud Engineering Is About Culture, Not Containers," Holly Cummins Verification, Validation, and Testing of Engineered Systems John Wiley & Sons Systems' Verification Validation and Testing (VVT) are carried out throughout systems' lifetimes. Notably, quality-cost expended on performing VVT activities and correcting system defects consumes about half of the overall engineering cost. Verification, Validation and Testing of Engineered Systems provides a comprehensive compendium of VVT activities and corresponding VVT methods for implementation throughout the entire lifecycle of an engineered system. In addition, the book strives to alleviate the fundamental testing conundrum, namely: What should be tested? How should one test? When should one test? And, when should one stop testing? In other words, how should one select a VVT strategy and how it be optimized? The book is organized in three parts: The first part provides introductory material about systems and VVT concepts. This part presents a comprehensive explanation of the role of VVT in the process of engineered systems (Chapter-1). The second part describes 40 systems' development VVT activities (Chapter-2) and 27 systems' post-development activities (Chapter-3). Corresponding to these activities, this part also describes 17 non-testing systems' VVT methods (Chapter-4) and 33 testing systems' methods (Chapter-5). The third part of the book describes ways to model systems’ quality cost, time and risk (Chapter-6), as well as ways to acquire quality data and optimize the VVT strategy in the face of funding, time and other

resource limitations as well as different business objectives (Chapter-7). Finally, this part describes the methodology used to validate the quality model along with a case study describing a system’s quality improvements (Chapter-8). Fundamentally, this book is written with two categories of audience in mind. The first category is composed of VVT practitioners, including Systems, Test, Production and Maintenance engineers as well as first and second line managers. The second category is composed of students and faculties of Systems, Electrical, Aerospace, Mechanical and Industrial Engineering schools. This book may be fully covered in two to three graduate level semesters; although parts of the book may be covered in one semester. University instructors will most likely use the book to provide engineering students with knowledge about VVT, as well as to give students an introduction to formal modeling and optimization of VVT strategy. Architecting Spacecraft With Sysml A Model-based Systems Engineering Approach Createspace Independent Publishing Platform **A Guide to Apply a Model-based Systems Engineering Approach with SysML to Specify and Architect Systems. This book provides a straightforward guide to develop an architecture model of aSmall Satellite using the Systems Modeling Language (SysML(r)). SysML is a general-purpose modeling language used to specify and architect systems. Model-based Systems Engineering (MBSE) is intended to produce an integratedsystem model using SysML which reflects multiple views of the system to specify theinteraction and interconnection of its components, and their functions, states,interfaces, and performance and physical characteristics. The system model canenhance quality, reuse, and shared understanding of the system. This book can be used by instructors and students to learn how to apply MBSE with SysML, as wellas practitioners of MBSE and organizations as a reference approach for their application. System Architecture Strategy and Product Development for Complex Systems** Prentice Hall **Architecture and Function of Complex Systems Systems Architecture sheds light on the increasingly important study of electronic and computer system design. The text teaches programmers and engineering professionals how to examine the DNA of a system to understand its basis for competitive advantage. Building on the idea of architecture as a specialized field, theFirst Edition sets the precedent for studying systems architecture as a “science”. The material is highly connected to real world examples--many of them involving the participation of its authors. Focusing on how functions work together to create a coherent system, the text examines systems architecture in the disciplines of communication, robotics, exploration, medicine, and farm and space equipment. Model-based System and Architecture Engineering with the Arcadia Method** Elsevier **This book presents ARCADIA—a tooled method devoted to systems and architecture engineering, especially for those dealing with strong constraints to be reconciled (cost, performance, safety, security, reuse, consumption, weight). The book describes the detailed reasoning necessary to: understand the real customer need; define and share the product architecture among all engineering stakeholders; early validate its design and justify it; and ease and master integration, validation, verification and qualification (IVVQ). Offers a comprehensive examination of systems engineering, including the use of models to support it Not only yet another book on modeling, but rather a journey in systems engineering, enlightening the use of models to support it. Focuses on solitary modeling tasks while also covering prime collaborations between engineering stakeholders Examines modeling techniques to capture and share architecture and to early verify it against need and non-functional constraints Addresses subjects not usually covered by model-based system engineering (MBSE) methods, such as co-engineering with specialties, system/sub-system co-engineering, integration verification and validation Features a powerful, dedicated tool (Capella) Covers a range of topics, including an introduction to system engineering issues, an introduction to MBSE, a presentation of the method for beginners and a handy reference manual for advanced users Simple SysML for Beginners Using Sparx Enterprise Architect Simple SysML for Beginners Using Sparx Enterprise Architect is for beginners. The book assumes that you have just purchased a copy of Enterprise Architect and are anxious to get started, but otherwise don't know too much about SysML and don't have much experience using Enterprise Architect or any other similar tool. There are several good books on the market about SysML. However, these books show only finished diagrams. They don't cover the steps needed to construct the models and the diagrams. These steps can be remarkably complicated; the sequence of steps needed to construct the underlying model for a diagram is often less than obvious when using a real SysML tool. The purpose of this book is to help you get through the initial learning curve and get you on your way to becoming proficient at SysML modeling. MITRE Systems Engineering Guide Agile IT Organization Design For Digital Transformation and Continuous Delivery** Addison-Wesley Professional **Design IT Organizations for Agility at Scale Aspiring digital businesses need overall IT agility, not just development team agility. In Agile IT Organization Design , IT management consultant and ThoughtWorks veteran Sriram Narayan shows how to infuse agility throughout your organization. Drawing on more than fifteen years’ experience working with enterprise clients in IT-intensive industries, he introduces an agile approach to “Business–IT Effectiveness” that is as practical as it is valuable. The author shows how structural, political, operational, and cultural facets of organization design influence overall IT agility—and how you can promote better collaboration across diverse functions, from sales and marketing to product development, and engineering to IT operations. Through real examples, he helps you evaluate and improve organization designs that enhance autonomy, mastery, and purpose: the key ingredients for a highly motivated workforce. You’ll find “close range” coverage of team design, accountability, alignment, project finance, tooling, metrics, organizational norms, communication, and culture. For each, you’ll gain a deeper understanding of where your organization stands, and clear direction for making improvements. Ready to optimize the performance of your IT organization or digital business? Here are practical solutions for the long term, and for right now. Govern for value over predictability Organize for responsiveness, not lowest cost Clarify accountability for outcomes and for decisions along the way Strengthen the alignment of autonomous teams Move beyond project teams to capability teams Break down tool-induced silos Choose financial practices that are free of harmful side effects Create and retain great teams despite today’s “talent crunch” Reform metrics to promote (not prevent) agility Evolve culture through improvements to structure, practices, and leadership—and careful, deliberate interventions Agile Project Management with Scrum** Microsoft Press **The rules and practices for Scrum—a simple process for managing complex projects—are few, straightforward, and easy to learn. But Scrum’s simplicity itself—its lack of prescription—can be disarming, and new practitioners often find themselves reverting to old project management habits and tools and yielding lesser results. In this illuminating series of case studies, Scrum co-creator and evangelist Ken Schwaber identifies the real-world lessons—the successes and failures—culled from his years of experience coaching companies in agile project management. Through them, you’ll understand how to use Scrum to solve complex problems and drive better results—delivering more valuable software faster. Gain the foundation in Scrum theory—and practice—you need to: Rein in even the most complex, unwieldy projects Effectively manage unknown or changing product requirements Simplify the chain of command with self-managing development teams Receive clearer specifications—and feedback—from customers Greatly reduce project planning time and required tools Build—and release—products in 30-day cycles so clients get deliverables earlier Avoid missteps by regularly inspecting, reporting on, and fine-tuning projects Support multiple teams working on a large-scale project from many geographic locations Maximize return on investment! The DevOps Handbook How to Create World-Class Agility, Reliability, and Security in Technology Organizations** IT Revolution **Increase profitability, elevate work culture, and exceed productivity goals through DevOps practices. More than ever, the effective management of technology is critical for business competitiveness. For decades, technology leaders have struggled to balance agility, reliability, and security. The consequences of failure have never been greater—whether it's the healthcare.gov debacle, cardholder data breaches, or missing the boat with Big Data in the cloud. And yet, high performers using DevOps principles, such as Google, Amazon, Facebook, Etsy, and Netflix, are routinely and reliably deploying code into production hundreds, or even thousands, of times per day. Following in the footsteps of The Phoenix Project, The DevOps Handbook shows leaders how to replicate these incredible outcomes, by showing how to integrate Product Management, Development, QA, IT Operations, and Information Security to elevate your company and win in the marketplace. Crystal Clear A Human-Powered Methodology for Small Teams** Pearson Education **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 360º, 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. Becoming an Agile Software Architect Strategies, practices, and patterns to help architects design continually evolving solutions** Packt Publishing Ltd **A guide to successfully operating in a lean-agile organization for solutions architects and enterprise architects Key Features Develop the right combination of processes and technical excellence to address architectural challenges Explore a range of architectural techniques to modernize legacy systems Discover how to design and continuously improve well-architected sustainable software Book Description Many organizations have embraced Agile methodologies to transform their ability to rapidly respond to constantly changing customer demands. However, in this melee, many enterprises often neglect to invest in architects by presuming architecture is not an intrinsic element of Agile software development. Since the role of an architect is not pre-defined in Agile, many organizations struggle to position architects, often resulting in friction with other roles or a failure to provide a clear learning path for architects to be productive. This book guides architects and organizations through new Agile ways of incrementally developing the architecture for delivering an uninterrupted, continuous flow of values that meets customer needs. You'll explore various aspects of Agile architecture and how it differs from traditional architecture. The book later covers Agile architects' responsibilities and how architects can add significant value by positioning themselves appropriately in the Agile flow of work. Through examples, you'll also learn concepts such as architectural decision backlog,the last responsible moment, value delivery, architecting for change, DevOps, and evolutionary collaboration. By the end of this Agile book, you'll be able to operate as an architect in Agile development initiatives and successfully architect reliable software systems. What you will learn Acquire clarity on the duties of architects in Agile development Understand architectural styles such as domain-driven design and microservices Identify the pitfalls of traditional architecture and learn how to develop solutions Understand the principles of value and data-driven architecture Discover DevOps and continuous delivery from an architect's perspective Adopt Lean-Agile documentation and governance Develop a set of personal and interpersonal qualities Find out how to lead the transformation to achieve organization-wide agility Who this book is for This agile study guide is for architects currently working on agile development projects or aspiring to work on agile software delivery, irrespective of the methodology they are using. You will also find this book useful if you're a senior developer or a budding architect looking to understand an agile architect's role by embracing agile architecture strategies and a lean-agile mindset. To understand the concepts covered in this book easily, you need to have prior knowledge of basic agile development practices. Microservice Architecture Aligning Principles, Practices, and Culture** "O'Reilly Media, Inc." **Have you heard about the tremendous success Amazon and Netflix have had by switching to a microservice architecture? Are you wondering how this can benefit your company? Or are you skeptical about how it might work? If you've answered yes to any of these questions, this practical book will benefit you. You'll learn how to take advantage of the microservice architectural style for building systems, and learn from the experiences of others to adopt and execute this approach most successfully. The Art of Software Testing** John Wiley & Sons **The classic, landmark work on software testing The hardware and software of computing have changed markedly in the three decades since the first edition of The Art of Software Testing, but this book's powerful underlying analysis has stood the test of time. Whereas most books on software testing target particular development techniques, languages, or testing methods, The Art of Software Testing,**

Third Edition provides a brief but powerful and comprehensive presentation of time-proven software testing approaches. If your software development project is mission critical, this book is an investment that will pay for itself with the first bug you find. The new Third Edition explains how to apply the book's classic principles to today's hot topics including: Testing apps for iPhones, iPads, BlackBerrys, Androids, and other mobile devices Collaborative (user) programming and testing Testing for Internet applications, e-commerce, and agile programming environments Whether you're a student looking for a testing guide you'll use for the rest of your career, or an IT manager overseeing a software development team, The Art of Software Testing, Third Edition is an expensive book that will pay for itself many times over. Hands-On Microservices with Kubernetes Build, deploy, and manage scalable microservices on Kubernetes Packt Publishing Ltd **Enhance your skills in building scalable infrastructure for your cloud-based applications Key FeaturesLearn to design a scalable architecture by building continuous integration (CI) pipelines with KubernetesGet an in-depth understanding of role-based access control (RBAC), continuous deployment (CD), and observabilityMonitor a Kubernetes cluster with Prometheus and GrafanaBook Description Kubernetes is among the most popular open-source platforms for automating the deployment, scaling, and operations of application containers across clusters of hosts, providing a container-centric infrastructure. Hands-On Microservices with Kubernetes starts by providing you with in-depth insights into the synergy between Kubernetes and microservices. You will learn how to use Delinkcious, which will serve as a live lab throughout the book to help you understand microservices and Kubernetes concepts in the context of a real-world application. Next, you will get up to speed with setting up a CI/CD pipeline and configuring microservices using Kubernetes ConfigMaps. As you cover later chapters, you will gain hands-on experience in securing microservices, and implementing REST, gRPC APIs, and a Delinkcious data store. In addition to this, you'll explore the Nuclio project, run a serverless task on Kubernetes, and manage and implement data-intensive tests. Toward the concluding chapters, you'll deploy microservices on Kubernetes and learn to maintain a well-monitored system. Finally, you'll discover the importance of service meshes and how to incorporate Istio into the Delinkcious cluster. By the end of this book, you'll have gained the skills you need to implement microservices on Kubernetes with the help of effective tools and best practices. What you will learnUnderstand the synergy between Kubernetes and microservicesCreate a complete CI/CD pipeline for your microservices on KubernetesDevelop microservices on Kubernetes with the Go kit framework using best practicesManage and monitor your system using Kubernetes and open-source toolsExpose your services through REST and gRPC APIsImplement and deploy serverless functions as a serviceExternalize authentication, authorization and traffic shaping using a service meshRun a Kubernetes cluster in the cloud on Google Kubernetes EngineWho this book is for This book is for developers, DevOps engineers, or anyone who wants to develop large-scale microservice-based systems on top of Kubernetes. If you are looking to use Kubernetes on live production projects or want to migrate existing systems to a modern containerized microservices system, then this book is for you. Coding skills, together with some knowledge of Docker, Kubernetes, and cloud concepts will be useful. Agile Development with ICONIX Process People, Process, and Pragmatism** Apress ***Describes an agile process that works on large projects *Ideal for hurried developers who want to develop software in teams *Incorporates real-life C#/.NET web project; can compare this with cases in book Software Quality Assurance Consistency in the Face of Complexity and Change** Springer **This textbook offers undergraduate students an introduction to the main principles and some of the most popular techniques that constitute 'software quality assurance'. The book seeks to engage students by placing an emphasis on the underlying foundations of modern quality-assurance techniques , using these to highlight why techniques work, as opposed to merely focussing on how they work. In doing so it provides readers with a comprehensive understanding of where software quality fits into the development lifecycle (spoiler: everywhere), and what the key quality assurance activities are. The book focuses on quality assurance in a way that typical, more generic software engineering reference books do not. It is structured so that it can (and should) be read from cover to cover throughout the course of a typical university module. Specifically, it is Concise: it is small enough to be readable in its entirety over the course of a typical software engineering module. Explanatory: topics are discussed not merely in terms of what they are, but also why they are the way they are – what events, technologies, and individuals or organisations helped to shape them into what they are now. Applied: topics are covered with a view to giving the reader a good idea of how they can be applied in practice, and by pointing, where possible, to evidence of their efficacy. The book starts from some of the most general notions (e.g. quality and development process), and gradually homes-in on the more specific activities, assuming knowledge of the basic notions established in prior chapters. Each chapter concludes with a "Key Points" section, summarising the main issues that have been covered in the chapter. Throughout the book there are exercises that serve to remind readers of relevant parts in the book that have been covered previously, and give them the opportunity to reflect on a particular topic and refer to related references. Low Earth Orbit Satellite Design** Springer **In recent decades, the number of satellites being built and launched into Earth's orbit has grown immensely, alongside the field of space engineering itself. This book offers an in-depth guide to engineers and professionals seeking to understand the technologies behind Low Earth Orbit satellites. With access to special spreadsheets that provide the key equations and relationships needed for mastering spacecraft design, this book gives the growing crop of space engineers and professionals the tools and resources they need to prepare their own LEO satellite designs, which is especially useful for designers of small satellites such as those launched by universities. Each chapter breaks down the various mathematics and principles underlying current spacecraft software and hardware designs. Hands-On Software Engineering with Python Move beyond basic programming and construct reliable and efficient software with complex code** Packt Publishing Ltd **Explore various verticals in software engineering through high-end systems using Python Key FeaturesMaster the tools and techniques used in software engineeringEvaluates available database options and selects one for the final Central Office system-componentsExperience the iterations software go through and craft enterprise-grade systemsBook Description Software Engineering is about more than just writing code—it includes a host of soft skills that apply to almost any development effort, no matter what the language, development methodology, or scope of the project. Being a senior developer all but requires awareness of how those skills, along with their expected technical counterparts, mesh together through a project's life cycle. This book walks you through that discovery by going over the entire life cycle of a multi-tier system and its related software projects. You'll see what happens before any development takes place, and what impact the decisions and designs made at each step have on the development process. The development of the entire project, over the course of several iterations based on real-world Agile iterations, will be executed, sometimes starting from nothing, in one of the fastest growing languages in the world—Python. Application of practices in Python will be laid out, along with a number of Python-specific capabilities that are often overlooked. Finally, the book will implement a high-performance computing solution, from first principles through complete foundation. What you will learnUnderstand what happens over the course of a system's life (SDLC)Establish what to expect from the pre-development life cycle stepsFind out how the development-specific phases of the SDLC affect developmentUncover what a real-world development process might be like, in an Agile wayFind out how to do more than just write the codeIdentify the existence of project-independent best practices and how to use themFind out how to design and implement a high-performance computing processWho this book is for Hands-On Software Engineering with Python is for you if you are a developer having basic understanding of programming and its paradigms and want to skill up as a senior programmer. It is assumed that you have basic Python knowledge. Encyclopedia of Information Science and Technology** IGI Global Snippet **"This set of books represents a detailed compendium of authoritative, research-based entries that define the contemporary state of knowledge on technology"--Provided by publisher.**