Test Strategy Document Example

A hands-on guide to testing techniques that deliver reliable software and systems Testing even a simple system can quickly turn into a potentially infinite task. Faced with tight costs and schedules, testers need to have a toolkit of practical techniques combined with hands-on experience and the right strategies in order to complete a successful project. World-renowned testing expert Rex Black provides you with the proven methods and concepts that test professionals must know. He presents you with the fundamental techniques for testing and clearly shows you how to select and apply successful strategies to test a system with budget and time constraints. Black begins by discussing the goals and tactics of effective and efficient testing. Next, he lays the foundation of his technique for risk-based testing, explaining how to analyze, prioritize, and document risks to the quality of the system using both informal and formal techniques. He then clearly describes how to design, develop, and, ultimately, document various kinds of tests. Because this is a hands-on activity, Black includes realistic, life-sized exercises that illustrate all of the major test techniques with detailed solutions. By the end of this book, you'll know more about the nuts and bolts of testing than most testers learn in an entire career, and you'll be ready to put those ideas into action on your next test project. With the help of real-world examples integrated throughout the chapters, you'll discover how to: Analyze the risks to system quality Allocate your testing effort appropriately based on the level of risk Choose the right testing strategies every time Design tests based on a system's expected behavior (black box) or internal structure (white box) Plan and perform integration testing Explore and attack the system Focus your hard work to serve the needs of the project The author's companion Web site provides exercises, tips, and techniques that can be used to gain valuable experience and effectively test software and systems. Wiley Technology Publishing Timely. Practical. Reliable. Visit the author's Web site at http://www.rexblackconsulting.com/
"This book fills a huge gap in our knowledge of software testing. It does an excellent job describing how test automation differs from other test activities, and clearly lays out what kind of skills and knowledge are needed to automate tests. The book is essential reading for students of testing and a bible for practitioners." –Jeff Offutt, Professor of Software Engineering, George Mason University "This new book naturally expands upon its predecessor, Automated Software Testing, and is the perfect reference for software practitioners applying automated software testing to their development efforts. Mandatory reading for software testing professionals!" –Jeff Rashka, PMP, Coauthor of Automated Software Testing and Quality Web Systems Testing accounts for an increasingly large percentage of the time and cost of new software development. Using automated software testing (AST), developers and software testers can optimize the software testing lifecycle and thus reduce cost. As technologies and development grow increasingly complex, AST becomes even more indispensable. This book builds on some of the proven practices and the automated testing lifecycle methodology (ATLM) described in Automated Software Testing and provides a renewed practical, start-to-finish guide to implementing AST successfully. In Implementing Automated Software Testing, three leading experts explain AST in detail, systematically reviewing its components, capabilities, and limitations. Drawing on their experience deploying AST in both defense and commercial industry, they walk you through the entire implementation process–identifying best practices, crucial success factors, and key pitfalls along with solutions for avoiding them. You will learn how to: Make a realistic business case for AST, and use it to drive your initiative Clarify your testing requirements and develop an automation strategy that reflects them Build efficient test environments and choose the right automation tools and techniques for your environment Use proven metrics to continuously track your

progress and adjust accordingly Whether you're a test professional, QA specialist, project manager, or developer, this book can help you bring unprecedented efficiency to testing–and then use AST to improve your entire development lifecycle.

The book covers all knowledge areas from the BABOK®, Third Edition, and is designed to be a study guide for the CBAP® certification from IIBATM. It includes over 300 sample questions. It is also usable for those seeking the PMI-PBA® certification. This book is a complete business analysis handbook combining the latest standards from the BABOK® case study examples and exercises with solutions. It has usable tools and techniques, as well as templates ready to be used to develop solid requirements to be the cornerstone for any successful product development.

This book teaches test managers what they need to know to achieve advanced skills in test estimation, test planning, test monitoring, and test control. Readers will learn how to define the overall testing goals and strategies for the systems being tested. This hands-on, exercise-rich book provides experience with planning, scheduling, and tracking these tasks. You'll be able to describe and organize the necessary activities as well as learn to select, acquire, and assign adequate resources for testing tasks. You'll learn how to form, organize, and lead testing teams, and master the organizing of communication among the members of the testing teams, and between the testing teams and all the other stakeholders. Additionally, you'll learn how to justify decisions and provide adequate reporting information where applicable. With over thirty years of software and systems engineering experience, author Rex Black is President of RBCS, is a leader in software, hardware, and systems testing, and is the most prolific author practicing in the field of software testing today. He has published a dozen books on testing that have sold tens of thousands of copies worldwide. He is past president of the International Software Testing Qualifications Board (ISTQB) and a director of the American Software Testing Qualifications Board (ASTQB). This book will help you prepare for the ISTQB Advanced Test Manager exam. Included are sample exam questions, at the appropriate level of difficulty, for most of the learning objectives covered by the ISTQB Advanced Level Syllabus. The ISTQB certification program is the leading software tester certification program in the world. With about 300,000 certificate holders and a global presence in over 50 countries, you can be confident in the value and international stature that the Advanced Test Manager certificate can offer you. This second edition has been thoroughly updated to reflect the new ISTQB Advanced Test Manager 2012 Syllabus, and the latest ISTQB Glossary. This edition reflects Rex Black's unique insights into these changes, as he was one of the main participants in the ISTQB Advanced Level Working Group.

Now in its fourth edition, Foundations of Software Testing: ISTQB Certification is the essential guide to software testing and to the ISTQB Foundation qualification. Completely updated to comprehensively reflect the most recent changes to the 2018 ISTQB Foundation Syllabus, the book adopts a practical, hands-on approach, covering the fundamental topics that every system and software tester should know. The authors are themselves developers of the ISTQB syllabus and are highly respected international authorities and teachers within the field of software testing. About ISTQB ISTQB is a multinational body overseeing the development of international qualifications in software testing. It offers an internationally recognized qualification that ensures there is an international, common understanding of software and system testing issues.

Quickly ramp up your practical knowledge of Apache JMeter for software performance testing and focus on actual business problems. This step-by-step guide covers what you will need to know to write and execute test scripts, and verify the results. Pro Apache JMeter covers almost every aspect of Apache JMeter in detail and includes helpful screenshots and a case study. A performance primer chapter provides a high-level summary of terms used in performance testing on a day-to-day basis that also is useful for non-technical readers. A sample web

application Digital Toys has been developed and test scripts are provided for you to try while progressing through the chapters. What You'll Learn Create and execute an Apache JMeter test plan Interpret the results of your test plan Understand distributed testing using Apache JMeter Use Apache JMeter advanced features such as JDBC, REST, FTP, AJAX, SOAP, and mobile performance testing Read a sample case study covering end-to-end planning and execution of a performance testing project Generate and analyze a performance dashboard Who This Book Is For Software performance testing professionals, quality assurance professionals, architects, engineers, project managers, product managers

The quality of a data warehouse (DWH) is the elusive aspect of it, not because it is hard to achieve [once we agree what it is], but because it is difficult to describe. We propose the notion that quality is not an attribute or a feature that a product has to possess, but rather a relationship between that product and each and every stakeholder. More specifically, the relationship between the software quality and the organization that produces the products is explored. Quality of data that populates the DWH is the main concern of the book, therefore we propose a definition for data quality as: "fitness to serve each and every purpose". Methods are proposed throughout the book to help readers achieve data warehouse quality.

The calculus of variations is a classical area of mathematical analysis yet its myriad applications in science and technology continue to keep it an active area of research. Encompassing two volumes, this set brings together leading experts who focus on critical point theory, differential equations, and the variational aspects of optimal control. The books cover monotonicity, nonlinear optimization, the impossible pilot wave, the Lavrentiev phenomenon, and elliptic problems.

"Don's book is a very good addition both to the testing literature and to the literature on quality assurance and software engineering… . [It] is likely to become a standard for test training as well as a good reference for professional testers and developers. I would also recommend this book as background material for negotiating outsourced software contracts. I often work as an expert witness in litigation for software with very poor quality, and this book might well reduce or eliminate these lawsuits…." –Capers Jones, VP and CTO, Namcook Analytics LLC Software and system testers repeatedly fall victim to the same pitfalls. Think of them as "anti-patterns": mistakes that make testing far less effective and efficient than it ought to be. In Common System and Software Testing Pitfalls, Donald G. Firesmith catalogs 92 of these pitfalls. Drawing on his 35 years of software and system engineering experience, Firesmith shows testers and technical managers and other stakeholders how to avoid falling into these pitfalls, recognize when they have already fallen in, and escape while minimizing their negative consequences. Firesmith writes for testing professionals and other stakeholders involved in large or medium-sized projects. His anti-patterns and solutions address both "pure software" applications and "software-reliant systems," encompassing heterogeneous subsystems, hardware, software, data, facilities, material, and personnel. For each pitfall, he identifies its applicability, characteristic symptoms, potential negative consequences and causes, and offers specific actionable recommendations for avoiding it or limiting its consequences. This guide will help you Pinpoint testing processes that need improvement–before, during, and after the project Improve shared understanding and collaboration among all project participants Develop, review, and optimize future project testing programs Make your test documentation far more useful Identify testing risks and appropriate risk-mitigation strategies Categorize testing problems

for metrics collection, analysis, and reporting Train new testers, QA specialists, and other project stakeholders With 92 common testing pitfalls organized into 14 categories, this taxonomy of testing pitfalls should be relatively complete. However, in spite of its comprehensiveness, it is also quite likely that additional pitfalls and even missing categories of pitfalls will be identified over time as testers read this book and compare it to their personal experiences. As an enhancement to the print edition, the author has provided the following location on the web where readers can find major additions and modifications to this taxonomy of pitfalls: http://donald.firesmith.net/home/common-testing-pitfalls Please send any recommended changes and additions to dgf (at) sei (dot) cmu (dot) edu, and the author will consider them for publication both on the website and in future editions of this book.

Teaches readers how to test and analyze software to achieve an acceptable level of quality at an acceptable cost Readers will be able to minimize software failures, increase quality, and effectively manage costs Covers techniques that are suitable for near-term application, with sufficient technical background to indicate how and when to apply them Provides balanced coverage of software testing & analysis approaches By incorporating modern topics and strategies, this book will be the standard software-testing textbook

Good requirements do not come from a tool, or from a customer interview. They come from a repeatable set of processes that take a project from the early idea stage through to the creation of an agreed-upon project and product scope between the customer and the developer.From enterprise analysis and planning requirements gathering to documentation,

Master continuous integration, deployment and automated testing for Android apps. You'll see how to set up and tear down sandbox environments to test the end-user experience, where you'll learn how to manage a mobile device in addition to the build machine. Android Continuous Integration applies a real-world CI pattern that has been thoroughly tested and implemented. This book starts with continuous integration concepts and the tools and code needed to become proficient in continuous integration for Android apps. You'll also follow acceptance test driven development (ATDD) best practice, giving you all the skills you need to become a better, more effective developer. Finally, you'll learn about the Appium mobile automation library and the Jenkins continuous integration tool. What You Will Learn Understand how to build an Android mobile app from source Set up a development or debugging environment for mobile apps Integrate with the Nexus dependency management and application release tool Work with the SonarQube code quality analyzer Use debugging tools in Android Who This Book Is For Product owners/business analysts, QA/test engineers, developers and build/deploy engineers.

Decades of software testing experience condensed into the most important lessons learned. The world's leading software testing experts lend you their wisdom and years of experience to help you avoid the most common mistakes in testing software. Each lesson is an assertion related to software testing, followed by an explanation or example that shows you the how, when, and why of the testing lesson. More than just tips, tricks, and pitfalls to avoid, Lessons Learned in Software Testing speeds you through the critical testing phase of the software development project without the extensive trial and error it normally takes to do so. The ultimate resource for software testers and developers at every level of expertise, this guidebook features: * Over 200 lessons

gleaned from over 30 years of combined testing experience * Tips, tricks, and common pitfalls to avoid by simply reading the book rather than finding out the hard way * Lessons for all key topic areas, including test design, test management, testing strategies, and bug reporting * Explanations and examples of each testing trouble spot help illustrate each lesson's assertion

This revised edition of Software Engineering-Principles and Practices has become more comprehensive with the inclusion of several topics. The book now offers a complete understanding of software engineering as an engineering discipline. Like its previous edition, it provides an in-depth coverage of fundamental principles, methods and applications of software engineering. In addition, it covers some advanced approaches including Computer-aided Software Engineering (CASE), Component-based Software Engineering (CBSE), Clean-room Software Engineering (CSE) and formal methods.Taking into account the needs of both students and practitioners, the book presents a pragmatic picture of the software engineering methods and tools. A thorough study of the software industry shows that there exists a substantial difference between classroom study and the practical industrial application. Therefore, earnest efforts have been made in this book to bridge the gap between theory and practical applications. The subject matter is well supported by examples and case studies representing the situations that one actually faces during the software development process.The book meets the requirements of students enrolled in various courses both at the undergraduate and postgraduate levels, such as BCA, BE, BTech, BIT, BIS, BSc, PGDCA, MCA, MIT, MIS, MSc, various DOEACC levels and so on. It will also be suitable for those software engineers who abide by scientific principles and wish to expand their knowledge. With the increasing demand of software, the software engineering discipline has become important in education and industry. This thoughtfully organized second edition of the book provides its readers a profound knowledge of software engineering concepts and principles in a simple, interesting and illustrative manner.

Achieving Systems Safety contains papers presented at the twentieth annual Safety-critical Systems Symposium, held in Bristol, UK, in February 2012. The Symposium is for engineers, managers and academics in the field of system safety, across all industry sectors, so the papers making up this volume offer a wide-ranging coverage of current safety topics, and a blend of academic research and industrial experience. They include both recent developments in the field and discussion of open issues that will shape future progress. The topics covered by the 20 papers in this volume include vulnerabilities in global navigation satellite systems; safety culture and community; transport safety; cyber-attacks on safety-critical systems; improving our approach to systems safety; accidents; assessment, validation and testing; safety standards and safety levels. The book will be of interest to both academics and practitioners working in the safety-critical systems arena.

Effective Software Testing explores fifty critically important best practices, pitfalls, and solutions. Gleaned from the author's extensive practical experience, these concrete items will enable quality assurance professionals and test managers to immediately enhance their understanding and skills, avoid costly mistakes, and implement a state-of-the-art testing program. This book places special emphasis on the integration of testing into all phases of the software development life cycle--from requirements definition to design and final coding. The fifty lessons provided here focus on the key aspects of software testing: test planning, design,

documentation, execution, managing the testing team, unit testing, automated testing, nonfunctional testing, and more. You will learn to: Base testing efforts on a prioritized feature schedule Estimate test preparation and execution Define the testing team roles and responsibilities Design test procedures as soon as requirements are available Derive effective test cases from requirements Avoid constraints and detailed data elements in test procedures Make unit-test execution part of the build process Use logging to increase system testability Test automated test tools on an application prototype Automate regression tests whenever possible Avoid sole reliance on capture/playback Conduct performance testing with production-sized databases Tailor usability tests to the intended audience Isolate the test environment from the development environment Implement a defect tracking life cycle Throughout the book, numerous real-world case studies and concrete examples illustrate the successful application of these important principles and techniques. Effective Software Testing provides ready access to the expertise and advice of one of the world's foremost software quality and testing authorities. 0201794292B12032002

Management, Computers, Computer networks, Information exchange, Data processing, IT and Information Management: IT Service Management

This book is designed to help students on an 'A' Level, Advanced VCE or similar course to design and implement a Web site using Microsoft FrontPage 2002. FrontPage is a suitable software package for the project component in 'AS' level and Advanced VCE courses. Parts 1 to 2 of the book take the reader through the steps in creating and publishing a Web site both with and without the use of a wizard. Part 3 gives advice on all stages of project work from the definition of a suitable problem through to documentation. The book was written primarily for 'AS' and 'A' Level Information and Communication Technology students and contains in Appendix B the AQA mark scheme for 'AS' Module 3: Coursework. It will also be suitable for students on many other courses at different levels since the mark scheme, with minor variations, is one which applies to projects in many ICT courses. For a second year project in an 'A' Level course, data collected from a Web site can be exported and used in an Access database. This aspect could be developed further to fulfil the requirements of, for example, Module 6 of the AQA ICT specification.A sample project is included to show students how a complete project report may be laid out.

With a view to assisting the evaluation of integrated approaches to testing and assessment (IATA) in regulatory decision-making within OECD Member Countries, this guidance document provides a set of principles for reporting defined approaches to testing and assessment that can be used as one of ...

Successful software depends not only on technical excellence but on how members of the software team work together. Written in easy to understand language by a leading expert in the field, this ground-breaking volume provides an overview of the team culture required to develop quality software. Reflecting the different views on the nature of software

quality, the book helps groups in a software team to communicate more effectively and to overcome the conflict created by their different perceptions of quality. You learn the roles and activities of team members (including customers) throughout the life of a software product, from before the software development starts and during the software development lifecycle, to after the software has been deployed and is in use.

At a time when information systems are becoming ever more complex and quality to market and time to market are critical for many companies, a structured test process is essential. Even more important is a structured test management process to keep testing under control. Nowadays a test manager must have extensive knowledge of and experience with project management, risk assessment, team building, and, process improvement. Based on their long-term industry experience, Pinkster and her coauthors describe a holistic approach to test management that combines test methods, test management, risk assessment and stakeholder management into one integral process, giving test managers, test coordinators, IT project managers, and QA managers a competitive edge in environments where there are numerous unstructured requirements, tough testing schedules and limited resources. This book should be in every test manager's backpack!

Software testing is a critical aspect of the software development process, and this heavily illustrated reference takes professionals on a complete tour of this increasingly important, multi-dimensional area. The book offers a practical understanding of all the most critical software testing topics and their relationships and inter-dependencies. This unique resource utilizes a wealth of graphics that support the discussions to offer a clear overview of software testing, from the definition of testing and the value and purpose of testing, through the complete testing process with all its activities, techniques and documentation, to the softer aspects of people and teams working with testing. Practitioners find numerous examples and exercises presented in each chapter to help ensure a complete understanding of the material. The book supports the ISTQB certification and provides a bridge from this to the ISO 29119 Software Testing Standard in terms of extensive mappings between the two; this is a truly unique feature.

Introducing the Most Helpful and Inexpensive Software Testing Study Guide: Stop yourself trying to figuring out how to succeed in your software testing career. Instead, take benefit of these proven methods and real-life examples. Being a software tester for over 9 years I personally know what it takes to get a job and advance in your software testing/QA career. Each and every page of this book consist of proven advice for handling the day to day software testing activities. Who should use this book? It doesn't matter if you are an undergraduate or graduate student or a fresher looking for a job in software testing or a professional working as a test engineer or a senior QA lead or a test manager, this eBook is designed to be used as the primary textbook and an all-in-one resource for software test engineers and developers. What

You'll learn after reading this eBook... * You should be able to get a job with our comprehensive guide on resume and interview preparation. * Get started in software testing. * Learn best tips on how to become a skilled software tester who finds critical defects in any application * Learn how to manage defects like a pro. * Become a web testing expert. * Learn how to achieve exponential career growth and excel in your career. * Learn how to deal with the developers during uncomfortable project meetings. * Master the art of becoming a good team leader/manager. * Plug-in all real-life tips and examples into almost any of your career situations for a bright software testing career. This eBook strives to strike a perfect balance between theoretical concepts, which are covered rigorously as well as practical contexts thus allowing the readers to build a solid foundation in key methodologies, techniques, tips and tricks in the field of software testing. The clear terminology definitions and comprehensive real-life examples provide an easy way to master various software testing techniques. After reading this eBook you should be able to get started in software testing, learn great tips on how to be an effective tester who finds critical bugs in the application under test, learn how to deal with the developers during uncomfortable project meetings, master the art of how to become a good test team leader/manager and more.

Written by the founder and executive director of the Quality Assurance Institute, which sponsors the most widely accepted certification program for software testing Software testing is a weak spot for most developers, and many have no system in place to find and correct defects quickly and efficiently This comprehensive resource provides step-by-step guidelines, checklists, and templates for each testing activity, as well as a self-assessment that helps readers identify the sections of the book that respond to their individual needs Covers the latest regulatory developments affecting software testing, including Sarbanes-Oxley Section 404, and provides guidelines for agile testing and testing for security, internal controls, and data warehouses CD-ROM with all checklists and templates saves testers countless hours of developing their own test documentation Note: CD-ROM/DVD and other supplementary materials are not included as part of eBook file.

Software Testing and Continuous Quality Improvement, Second Edition, illustrates a quality framework for software testing in traditional structured and unstructured environments. It explains how a continuous quality improvement approach promotes effective testing, and it analyzes the various testing tools and techniques that you can choose.

Get past the myths of testing in agile environments - and implement agile testing the RIGHT way. * * For everyone concerned with agile testing: developers, testers, managers, customers, and other stakeholders. * Covers every key issue: Values, practices, organizational and cultural challenges, collaboration, metrics, infrastructure, documentation, tools, and more. * By two of the world's most experienced agile testing practitioners and consultants. Software testing has always been crucial, but it may be even more crucial in agile environments that rely heavily on repeated iterations of software capable of passing tests. There are, however, many myths associated with testing in agile environments. This book helps agile team members overcome those myths -- and implement testing that truly maximizes software quality and

value. Long-time agile testers Lisa Crispin and Janet Gregory offer powerful insights for three large, diverse groups of readers: experienced testers who are new to agile; members of newly-created agile teams who aren't sure how to perform testing or work with testers; and test/QA managers whose development teams are implementing agile. Readers will learn specific agile testing practices and techniques that can mean the difference between success and failure; discover how to transition 'traditional' test teams to agile; and learn how to integrate testers smoothly into agile teams. Drawing on extensive experience, the authors illuminate topics ranging from culture to test planning to automated tools. They cover every form of testing: business-facing tests, technology-facing tests, exploratory tests, context-driven and scenario tests, load, stability, and endurance tests, and more. Using this book's techniques, readers can improve the effectiveness and reduce the risks of any agile project or initiative.

Conallen introduces architects and designers and client/server systems to issues and techniques of developing software for the Web. He expects readers to be familiar with object-oriented principles and concepts, particularly with UML (unified modeling language), and at least one Web application architecture or environment. The second edition incorporates both technical developments and his experience since 1999. He does not provide a bibliography. Annotation copyrighted by Book News, Inc., Portland, OR

Software testing is traditionally seen as a difficult and time consuming activity that is hard to embed in the software development process. This book provides a different view. It explains to stakeholders how testing can add value to software development and doing business, and provides the tester with practical information. TestGoal is not just another methodology. Several good testing methodologies exist. But, like any other profession, also testing encompasses more than the simple application of a methodology. After all, strict adherence to a specific methodology is no guarantee for success. Success stems from the mindset, enthusiasm, knowledge and skill of the tester. These factors determine whether a methodology is applied successfully and whether testing takes on a result-driven character. And that's what TestGoal is about: a result-driven attitude, testing principles and expertise as fundament, and a hands-on six step plan to enable result driven testing. Derk-Jan de Grood and his colleagues from Collis, an international software testing company, know about the main pitfalls in test projects from their extensive professional experience. TestGoal has emerged from the office floor and captures over a decade of best practice. TestGoal is made by professionals for professionals, and it combines the mindset, knowledge, and skills required to add value with testing and make software development more successful. This book explains in a clear language how you can make testing result-driven. It explains why testing is important and describes all of the activities involved in testing. It is enriched with recognizable examples, practical tips and useful checklists. This makes it a "GO kit" that enables testers to immediately get started and add value to their organization.

The world of M&A has always been complex and nuanced. Corporations encounter their toughest business problems during a divestiture or a merger. At the same time, optimal execution of divestitures can also create high value for the seller as well as the buyer. This book is a collection of leading practices on Divestitures and covers end to end transaction life cycle from readiness through execution including post deal transformation. It contains the synthesis of experiences across a wide array of clients across industries, ranging from $500 million to $100 billion in revenue. Each chapter in this book can stand on its own as an authority on leading practices related to the topic it presents, and together, these chapters provide a comprehensive set of perspectives needed to successfully complete a divestiture. The highlight of the book is valuable real-life examples and references that a business can benefit from, when it is considering, analyzing or implementing a divestiture.

Five years on from its adoption in 1997 by the Object Management Group (OMG), the Uni?ed Modeling Language is the de facto standard for

creating - agrammatic models of software systems. More than 100 books have been written about UML, and it is taught to students throughout the world. The de?nition of UML version 2 is well under way, and should be largely completed within the year. This will not only improve and enhance UML itself, including standard facilities for diagram interchange, but also make it fully integrated with other modeling technologies from the OMG, such as Meta-Object Facility (MOF) and XML Metadata Interchange (XMI). The Object Constraint Language, which has become an important vehicle for communicating detailed insights between UML researchers and practitioners, will have a much expanded speci?cation and be better integrated with the UML. The popularity of UML signi?es the possibility of a shift of immense prop- tions in the practice of software development, at least comparable to the shift from the use of assembly language to "third-generation" or "high-level" p- gramming languages. We dream of describing the behavior of software systems in terms of models, closely related to the needs of the enterprise being served, and being able to routinely translate these models automatically into executing p- grams on distributed computing systems. The OMG is promoting Model-Driven Architecture (MDA) as a signi?cant step towards this vision, and the MDA c- cept has received considerable support within the IT industry.

An updated edition of the best tips and tools to plan, build, and execute a structured test operation In this update of his bestselling book, Rex Black walks you through how to develop essential tools and apply them to your test project. He helps you master the basic tools, apply the techniques to manage your resources, and give each area just the right amount of attention so that you can successfully survive managing a test project! Offering a thorough review of the tools and resources you will need to manage both large and small projects for hardware and software, this book prepares you to adapt the concepts across a broad range of settings. Simple and effective, the tools comply with industry standards and bring you up to date with the best test management practices and tools of leading hardware and software vendors. Rex Black draws from his own numerous testing experiences-- including the bad ones, so you can learn from his mistakes-- to provide you with insightful tips in test project management. He explores such topics as: Dates, budgets, and quality-expectations versus reality Fitting the testing process into the overall development or maintenance process How to choose and when to use test engineers and technicians, contractors and consultants, and external test labs and vendors Setting up and using an effective and simple bug-tracking database Following the status of each test case The companion Web site contains fifty tools, templates, and case studies that will help you put these ideas into action--fast!

Crandall's Power Supply Testing Handbook comes into the marketplace at an optimum time. Now, more than ever, there is an urgency for a comprehensive handbook on power supply testing that will fulfill the reference needs of the wide variety of professionals testing power supplies, including designers, manufacturers, purchasers, and field service organizations.

When implemented correctly, release management can help ensure that quality is integrated throughout the development, implementation, and delivery of services, applications, and infrastructure. This holistic, total cost of ownership approach allows for higher levels of system availability, is more cost effective to maintain, and increases overall s

Written in a clear and thoughtful style, Building a Successful Board-Test Strategy, Second Edition offers an integrated approach to the complicated process of developing the test strategies most suited to a company's profile and philosophy. This book also provides comprehensive coverage of the specifics of electronic test equipment as well as those broader issues of management and marketing that shape a manufacturer's "image of quality." In this new edition, the author adds still more "war stories," relevant examples from his own experience, which will guide his readers in their decisionmaking. He has also updated all technical aspects of the first edition, covering new

device and attachment technologies, new inspection techniques including optical, infrared and x-ray, as well as vectorless methods for detecting surface-mount open-circuit board failures. The chapter on economics has been extensively revised, and the bibliography includes the latest material on this topic. *Discusses ball-grid arrays and other new devices and attachment technologies *Adds a comprehensive new chapter on optical, infrared, and x-ray inspection *Covers vectorless techniques for detecting surface-mount open-circuit board failures
This book covers the ISTQB Expert Level Test Manager syllabus and is a complete, one-stop preparation guide for the reader who is otherwise qualified (based on experience as a test manager) to take the Expert Level Test Manager exam. Included are extensive hands-on exercises and sample exam questions that comply with ISTQB standards for Expert Level exams. p.p1 {margin: 0.0px 0.0px 0.0px 0.0px; font: 11.0px Verdana} p.p2 {margin: 0.0px 0.0px 0.0px 0.0px; font: 11.0px Verdana; min-height: 13.0px} The ISTQB certification program is the leading software tester certification program in the world. With more than 500,000 certificates issued and a global presence in 70 countries, you can be confident in the value and international stature that the ISTQB Expert Level certificate can offer you.
Software Testing Concepts and Tools provide experience-based practices and key concepts that can be used by any organization to implement a successful and efficient testing process. This book provides experience-based practices and key concepts that can be used by an organization to implement a successful and efficient testing process. The prime aim of this book is to provide a distinct collection of technologies and discussions that are directly applicable in software development organizations to improve the quality and avoid major mistakes and human errors.· Software Engineering Evaluation· System Testing Process· WinRunner 8.0· QTP 8.2· LoadRunner 8.0· TestDirector 8.0
Use an Approach Inspired by Domain-Driven Design to Build Documentation That Evolves to Maximize Value Throughout Your Development Lifecycle Software documentation can come to life, stay dynamic, and actually help you build better software. Writing for developers, coding architects, and other software professionals, Living Documentation shows how to create documentation that evolves throughout your entire design and development lifecycle. Through patterns, clarifying illustrations, and concrete examples, Cyrille Martraire demonstrates how to use well-crafted artifacts and automation to dramatically improve the value of documentation at minimal extra cost. Whatever your domain, language, or technologies, you don't have to choose between working software and comprehensive, high-quality documentation: you can have both. · Extract and augment available knowledge, and make it useful through living curation · Automate the creation of documentation and diagrams that evolve as knowledge changes · Use development tools to refactor documentation · Leverage documentation to improve software designs · Introduce living documentation to new and legacy environments

Copyright: 662cf615724ceabe51559e20d5eb71ce