

Object Oriented Software Engineering Kung

Object-Oriented Software Engineering: An Agile Unified Methodology by David Kung presents a step-by-step methodology that integrates modeling and design, UML, patterns, test-driven development, quality assurance, configuration management, and agile principles throughout the life cycle.

Read Book Object Oriented Software Engineering Kung

The overall approach is casual and easy to follow, with many practical examples that show the theory at work. The author uses his experiences as well as real-world stories to help the reader understand software design principles, patterns, and other software engineering concepts. The book also provides stimulating exercises that go far beyond the type of question that can be answered by simply copying portions of the text.

Read Book Object Oriented Software Engineering Kung

A superior primer on software testing and quality assurance, from integration to execution and automation This important new work fills the pressing need for a user-friendly text that aims to provide software engineers, software quality professionals, software developers, and students with the fundamental developments in testing theory and common testing practices. Software Testing and Quality Assurance: Theory and Practice equips readers with

Read Book Object Oriented Software Engineering Kung

a solid understanding of: Practices that support the production of quality software Software testing techniques Life-cycle models for requirements, defects, test cases, and test results Process models for units, integration, system, and acceptance testing How to build test teams, including recruiting and retaining test engineers Quality Models, Capability Maturity Model, Testing Maturity Model, and Test Process Improvement Model Expertly

Read Book Object Oriented Software Engineering Kung

balancing theory with practice, and complemented with an abundance of pedagogical tools, including test questions, examples, teaching suggestions, and chapter summaries, this book is a valuable, self-contained tool for professionals and an ideal introductory text for courses in software testing, quality assurance, and software engineering.

This book presents a selection of subjects which the authors deem to be

Read Book Object Oriented Software Engineering Kung

important for information systems engineers. The book is intended for introductory teaching. We have tried to write the book in such a way that students with only fragmented knowledge of computers are able to read the book without too many difficulties. Students who have had only an introductory course in computer programming should be able to read most of the book. We have tried to achieve simplicity without compromising on depth in our

Read Book Object Oriented Software Engineering Kung

discussions of the various aspects of information systems engineering. So it is our hope that also those who have deeper knowledge in computing may find pleasure in reading parts of the book. The writing of a textbook is a major undertaking for its authors. One is quite often forced to reexamine truisms in the subject area, and must be prepared to reevaluate one's opinions and priorities as one learns more. In particular this is so in new fields,

Read Book Object Oriented Software Engineering Kung

where formalisms have been scarcely used, and where consensus has not yet emerged either on what constitutes the subject area or on how practical problems within the field shall be approached. Contemporary practice in computer applications is confronted with an increasingly complex world, both in a technical sense and in the complexity of problems that are solved by computer.

This book contains both relevant real-

Read Book Object Oriented Software Engineering Kung

world research, as well as reviews of different areas of interest in the software engineering literature, such as clone identification. The contents of the various sections will provide a better understanding of known problems and detailed treatment of advanced topics. Consequently, the book consolidates the work and findings from leading researchers in the software research community in key areas such as maintainability, architectural

Read Book Object Oriented Software Engineering Kung

recovery, code analysis, software migration, and tool support.

M.E.I.S. Center, University of Minnesota, Minneapolis, 1983, November 16-17, 1983

Web Information Systems -- WISE 2004 10th International Symposium, CBSE 2007, Medford, MA, USA, July 9-11, 2007, Proceedings

Systems Analysis and Design in a Changing World

Dependable Computing - EDDC-3

Read Book Object Oriented Software Engineering Kung

Curriculum for Test Technology

Object orientation has become a ?must know? subject for managers, researchers, and software practitioners interested in the design, evolution, reuse and management of efficient software components. The book contains technical papers reflecting both theoretical and practical contributions from researchers in the field of object-oriented (OO) databases and software engineering systems. The book

Read Book Object Oriented Software Engineering Kung

identifies actual and potential areas of integration of OO and database technologies, current and future research directions in software methodologies, and reflections about the OO paradigm. In providing current research and relevant information about this promising and rapidly growing field of object-oriented databases and software engineering systems, this book is invaluable to research scientists, practitioners, and graduate students

Read Book Object Oriented Software Engineering Kung

working in the areas of databases and software engineering.

Extensively class-tested, this textbook takes an innovative approach to software testing: it defines testing as the process of applying a few well-defined, general-purpose test criteria to a structure or model of the software. It incorporates the latest innovations in testing, including techniques to test modern types of software such as OO, web applications,

Read Book Object Oriented Software Engineering Kung

and embedded software. The book contains numerous examples throughout. An instructor's solution manual, PowerPoint slides, sample syllabi, additional examples and updates, testing tools for students, and example software programs in Java are available on an extensive website.

This book constitutes the refereed proceedings of the 6th International Conference on Information Systems, Technology and Management, ICISTM 2012,

Read Book Object Oriented Software Engineering Kung

held in Grenoble, France, in March 2012. The 38 revised papers were carefully reviewed and selected from 85 submissions. The papers are organized in topical sections on information systems; information technology; information management; business intelligence; management science and education; applications; workshop on program protection and reverse engineering.

The practice of building software is a

Read Book Object Oriented Software Engineering Kung

“new kid on the block” technology. Though it may not seem this way for those who have been in the field for most of their careers, in the overall scheme of professions, software builders are relative “newbies.” In the short history of the software field, a lot of facts have been identified, and a lot of fallacies promulgated. Those facts and fallacies are what this book is about. There's a problem with those facts-and, as you might imagine, those

Read Book Object Oriented Software Engineering Kung

fallacies. Many of these fundamentally important facts are learned by a software engineer, but over the short lifespan of the software field, all too many of them have been forgotten. While reading Facts and Fallacies of Software Engineering , you may experience moments of "Oh, yes, I had forgotten that," alongside some "Is that really true?" thoughts. The author of this book doesn't shy away from controversy. In fact, each of the facts and

Read Book Object Oriented Software Engineering Kung

fallacies is accompanied by a discussion of whatever controversy envelops it. You may find yourself agreeing with a lot of the facts and fallacies, yet emotionally disturbed by a few of them! Whether you agree or disagree, you will learn why the author has been called “the premier curmudgeon of software practice.” These facts and fallacies are fundamental to the software building field—forget or neglect them at your peril!

Read Book Object Oriented Software Engineering Kung

Emerging Methods, Technologies, and
Process Management in Software
Engineering

Object-oriented Technology for Database
and Software Systems

A Practitioner's Approach

Software Testing and Quality Assurance

Object-Oriented and Classical Software
Engineering

The Compiler Design Handbook

Programming has become a significant part of
connecting theoretical development and

Read Book Object Oriented Software Engineering Kung

scientific application computation. Computer programs and processes that take into account the goals and needs of the user meet with the greatest success, so it behooves software engineers to consider the human element inherent in every line of code they write.

Research Anthology on Recent Trends, Tools, and Implications of Computer Programming is a vital reference source that examines the latest scholarly material on trends, techniques, and uses of various programming applications and examines the benefits and challenges of these computational developments. Highlighting a range of topics

Read Book Object Oriented Software Engineering Kung

such as coding standards, software engineering, and computer systems development, this multi-volume book is ideally designed for programmers, computer scientists, software developers, analysts, security experts, IoT software programmers, computer and software engineers, students, professionals, and researchers.

Classical and Object-Oriented Software Engineering, 5/e is designed for an introductory software engineering course.

This book provides an excellent introduction to software engineering fundamentals, covering both traditional and object-oriented

Read Book Object Oriented Software Engineering Kung

techniques. Schach's unique organization and style makes it excellent for use in a classroom setting. It presents the underlying software engineering theory in Part I and follows it up with the more practical life-cycle material in Part II. Many software engineering books are more like reference books, which do not provide the appropriate fundamentals before inundating students with implementation details. In this edition, more practical material has been added to help students understand how to use what they are learning. This has been done through the use of "How To" boxes and greater implementation

Read Book Object Oriented Software Engineering Kung

detail in the case study. Additionally, the new edition contains the references to the most current literature and includes an overview of extreme programming. The website in this edition will be more extensive. It will include Solutions, PowerPoints that incorporate lecture notes, newly developed self-quizz questions, and source code for the term project and case study.

Partial Contents: Formal/Analytic Methods for Dependability; Reliability Modeling; Survivability & Security; Formal Methods; Design for High Assurance; Fault Analysis & Predication; Testing & Analysis of High-

Read Book Object Oriented Software Engineering Kung

Assurance Systems; High-Assurance in
Intelligent Systems

The widespread use of object-oriented languages and Internet security concerns are just the beginning. Add embedded systems, multiple memory banks, highly pipelined units operating in parallel, and a host of other advances and it becomes clear that current and future computer architectures pose immense challenges to compiler designers-
challenges th

Proceedings : Third IEEE International High-Assurance Engineering Symposium : November 13-15, 1998, Washington, D.C.

Read Book Object Oriented Software Engineering Kung

An Introduction

History of Programming Languages

Object-Oriented Design Knowledge: Principles,

Heuristics and Best Practices

Java How to Program

Emerging Web Services Technology

A high-level introduction to new technologies and methods in the field of software engineering. Recent years have witnessed rapid evolution of software engineering methodologies, and until now, there has been no single-source introduction to emerging technologies

Read Book Object Oriented Software Engineering Kung

in the field. Written by a panel of experts and divided into four clear parts, Emerging Methods, Technologies, and Process Management in SoftwareEngineering covers: Software Architectures - Evolution of software composition mechanisms; compositionality in software product lines; and teaching design patterns Emerging Methods - The impact of agent-oriented software engineering in service-oriented computing; testing

Read Book Object Oriented Software Engineering Kung

object-oriented software; the UML and formal methods; and modern Web application development Technologies for Software Evolution - Migrating to Web services and software evolution analysis and visualization Process Management - Empirical experimentation in software engineering and foundations of agile methods Emerging Methods, Technologies, and Process Management in Software Engineering is a one-stop resource for software engineering

Read Book Object Oriented Software Engineering Kung

practitioners and professionals, and also serves as an ideal textbook for undergraduate and graduate students alike.

John Carroll shows how a pervasive but underused element of design practice, the scenario, can transform information systems design. Difficult to learn and awkward to use, today's information systems often change our activities in ways that we do not need or want. The problem lies in the software

Read Book Object Oriented Software Engineering Kung

development process. In this book John Carroll shows how a pervasive but underused element of design practice, the scenario, can transform information systems design. Traditional textbook approaches manage the complexity of the design process via abstraction, treating design problems as if they were composites of puzzles. Scenario-based design uses concretization. A scenario is a concrete story about use. For example: "A person turned on a

Read Book Object Oriented Software Engineering Kung

computer; the screen displayed a button labeled Start; the person used the mouse to select the button." Scenarios are a vocabulary for coordinating the central tasks of system development—understanding people's needs, envisioning new activities and technologies, designing effective systems and software, and drawing general lessons from systems as they are developed and used. Instead of designing software by listing

Read Book Object Oriented Software Engineering Kung

requirements, functions, and code modules, the designer focuses first on the activities that need to be supported and then allows descriptions of those activities to drive everything else. In addition to a comprehensive discussion of the principles of scenario-based design, the book includes in-depth examples of its application.

This work aims to provide the reader with sound engineering principles,

Read Book Object Oriented Software Engineering Kung

whilst embracing relevant industry practices and technologies, such as object orientation and requirements engineering. It includes a chapter on software architectures, covering software design patterns.

This book constitutes the proceedings of the 26th International Conference on Computer Aided Verification, CAV 2014, held as part of the Vienna Summer of Logic, VSL 2014, in Vienna, Austria, in July 2014. The 46 regular papers and 11

Read Book Object Oriented Software Engineering Kung

short papers presented in this volume were carefully reviewed and selected from a total of 175 regular and 54 short paper submissions. The contributions are organized in topical sections named: software verification; automata; model checking and testing; biology and hybrid systems; games and synthesis; concurrency; SMT and theorem proving; bounds and termination; and abstraction.

Software Engineering

Read Book Object Oriented Software Engineering Kung

*26th International Conference, CAV
2014, Held as Part of the Vienna Summer
of Logic, VSL 2014, Vienna, Austria,
July 18-22, 2014, Proceedings
Classical and Object-oriented Software
Engineering*

*6th International Conference, ICISTM
2012, Grenoble, France, March 28-30.
Proceedings*

*Introduction to Software Testing
Facts and Fallacies of Software
Engineering*

Read Book Object Oriented Software Engineering Kung

Requirements engineering is the process by which the requirements for software systems are gathered, analyzed, documented, and managed throughout their complete lifecycle. Traditionally it has been concerned with technical goals for, functions of, and constraints on software systems. Aurum and Wohlin, however, argue that it is no longer appropriate for software systems professionals to focus only on functional and non-functional aspects of the intended system and to somehow assume that organizational context and needs are outside their remit. Instead, they call for a broader perspective in order to gain a better understanding of the interdependencies between enterprise stakeholders, processes, and

Read Book Object Oriented Software Engineering Kung

software systems, which would in turn give rise to more appropriate techniques and higher-quality systems. Following an introductory chapter that provides an exploration of key issues in requirements engineering, the book is organized in three parts. Part 1 presents surveys of state-of-the art requirements engineering process research along with critical assessments of existing models, frameworks and techniques. Part 2 addresses key areas in requirements engineering, such as market-driven requirements engineering, goal modeling, requirements ambiguity, and others. Part 3 concludes the book with articles that present empirical evidence and experiences from practices in industrial projects.

Read Book Object Oriented Software Engineering Kung

Its broader perspective gives this book its distinct appeal and makes it of interest to both researchers and practitioners, not only in software engineering but also in other disciplines such as business process engineering and management science.

This book comprises the refereed proceedings of the International Conferences, ASEA and DRBC 2012, held in conjunction with GST 2012 on Jeju Island, Korea, in November/December 2012. The papers presented were carefully reviewed and selected from numerous submissions and focus on the various aspects of advanced software engineering and its applications, and disaster recovery and business continuity.

Read Book Object Oriented Software Engineering Kung

Refined and streamlined, SYSTEMS ANALYSIS AND DESIGN IN A CHANGING WORLD, 7E helps students develop the conceptual, technical, and managerial foundations for systems analysis design and implementation as well as project management principles for systems development. Using case driven techniques, the succinct 14-chapter text focuses on content that is key for success in today's market. The authors' highly effective presentation teaches both traditional (structured) and object-oriented (OO) approaches to systems analysis and design. The book highlights use cases, use diagrams, and use case descriptions required for a modeling approach, while demonstrating their application to

Read Book Object Oriented Software Engineering Kung

traditional, web development, object-oriented, and service-oriented architecture approaches. The Seventh Edition's refined sequence of topics makes it easier to read and understand than ever. Regrouped analysis and design chapters provide more flexibility in course organization. Additionally, the text's running cases have been completely updated and now include a stronger focus on connectivity in applications. Important Notice: Media content referenced within the product description or the product text may not be available in the ebook version.

History of Programming Languages presents information pertinent to the technical aspects of the

Read Book Object Oriented Software Engineering Kung

language design and creation. This book provides an understanding of the processes of language design as related to the environment in which languages are developed and the knowledge base available to the originators. Organized into 14 sections encompassing 77 chapters, this book begins with an overview of the programming techniques to use to help the system produce efficient programs. This text then discusses how to use parentheses to help the system identify identical subexpressions within an expression and thereby eliminate their duplicate calculation. Other chapters consider FORTRAN programming techniques needed to produce optimum object programs. This book discusses as well the

Read Book Object Oriented Software Engineering Kung

developments leading to ALGOL 60. The final chapter presents the biography of Adin D. Falkoff. This book is a valuable resource for graduate students, practitioners, historians, statisticians, mathematicians, programmers, as well as computer scientists and specialists.

Life Cycle Solutions

Testing and Quality Assurance for Component-based Software

Principles and Practice

Case Studies

Scenario-Based Design of Human-Computer Interactions

Object-Oriented Software Engrg?

Read Book Object Oriented Software Engineering Kung

Papers from an October 2002 symposium describe research in areas including algorithms, artificial intelligence, computer graphics, computer networks, databases, evolutionary computation, graph theory, image processing, multimedia technology, software engineering, and software performance engineering. Some specific topics are packet selection in a deflection routing algorithm, honeycomb subdivision, a new image-based lighting method, visualizing transition diagrams of action language programs, and solution stability in evolutionary computation. Other subjects include control of lightpaths in heterogeneous optical

Read Book Object Oriented Software Engineering Kung

networks, exploiting semantic constraints in a database browser, and bandwidth allocation in bluetooth scatternets. There is no subject index. Annotation copyrighted by Book News, Inc., Portland, OR

"The software engineering community has advanced greatly in recent years and we currently have numerous defined items of knowledge, such as standards, methodologies, methods, metrics, techniques, languages, patterns, knowledge related to processes, concepts, etc. The main objective of this book is to give a unified and global vision about Micro-

Read Book Object Oriented Software Engineering Kung

Architectural Design Knowledge, analyzing the main techniques, experiences and methods"--Provided by publisher.

More than ever, mission-critical and business-critical applications depend on object-oriented (OO) software. Testing techniques tailored to the unique challenges of OO technology are necessary to achieve high reliability and quality. "Testing Object-Oriented Systems: Models, Patterns, and Tools" is an authoritative guide to designing and automating test suites for OO applications. This comprehensive book explains why testing must be model-based and provides in-depth

Read Book Object Oriented Software Engineering Kung

coverage of techniques to develop testable models from state machines, combinational logic, and the Unified Modeling Language (UML). It introduces the test design pattern and presents 37 patterns that explain how to design responsibility-based test suites, how to tailor integration and regression testing for OO code, how to test reusable components and frameworks, and how to develop highly effective test suites from use cases. Effective testing must be automated and must leverage object technology. The author describes how to design and code specification-based assertions to offset testability losses due to inheritance and

Read Book Object Oriented Software Engineering Kung

polymorphism. Fifteen micro-patterns present oracle strategies--practical solutions for one of the hardest problems in test design. Seventeen design patterns explain how to automate your test suites with a coherent OO test harness framework. The author provides thorough coverage of testing issues such as: The bug hazards of OO programming and differences from testing procedural code How to design responsibility-based tests for classes, clusters, and subsystems using class invariants, interface data flow models, hierarchic state machines, class associations, and scenario analysis How to support reuse by

Read Book Object Oriented Software Engineering Kung

effective testing of abstract classes, generic classes, components, and frameworks How to choose an integration strategy that supports iterative and incremental development How to achieve comprehensive system testing with testable use cases How to choose a regression test approach How to develop expected test results and evaluate the post-test state of an object How to automate testing with assertions, OO test drivers, stubs, and test frameworks Real-world experience, world-class best practices, and the latest research in object-oriented testing are included. Practical examples illustrate test design and

Read Book Object Oriented Software Engineering Kung

test automation for Ada 95, C++, Eiffel, Java, Objective-C, and Smalltalk. The UML is used throughout, but the test design patterns apply to systems developed with any OO language or methodology. 0201809389B04062001

The idea of creating the European Dependable Computing Conference (EDCC) was born at the moment when the Iron Curtain fell. A group of enthusiasts, who were previously involved in research and teaching in the field of fault tolerant computing in different European countries, agreed that there is no longer any point in keeping previously independent

Read Book Object Oriented Software Engineering Kung

activities apart and created a steering committee which took the responsibility for preparing the EDCC calendar and appointing the chairs for the individual conferences. There is no single European or global professional organization that took over the responsibility for this conference, but there are three national interest groups that sent delegates to the steering committee and support its activities, especially by promoting the conference materials. As can be seen from these materials, they are the SEE Working Group "Dependable Computing" (which is a successor organization of AFCET) in France, the GI/ITG/GMATec

Read Book Object Oriented Software Engineering Kung

Technical Committee on Dependability and Fault Tolerance in Germany, and the AICA Working Group "Dependability of Computer Systems" in Italy. In addition, committees of several global professional organizations, such as IEEE and IFIP, support this conference. Prague has been selected as a conference venue for several reasons. It is an easily accessible location that may attract many visitors by its beauty and that has a tradition in organizing international events of this kind (one of the last FTSD conferences took place here).

Component-based Software Development

Read Book Object Oriented Software Engineering Kung

Trends in Software Engineering

Object-Oriented Software Engineering: An Agile
Unified Methodology

Information Systems Engineering

High-assurance Systems Engineering Symposium
A Process-Oriented Approach

- First book of its kind (case studies in CBD) - Covers
different kinds of components - Covers different
component models/technologies - Includes a wide
scope of CBD topics - Covers both theoretical and
practical work - Includes both formal and informal
approaches - Provides a snapshot of current concerns

Read Book Object Oriented Software Engineering Kung

and pointers to future trends

The Deitels' groundbreaking How to Program series offers unparalleled breadth and depth of object-oriented programming concepts and intermediate-level topics for further study. This survey of Java programming contains an optional extensive OOD/UML 2 case study on developing and implementing the software for an automated teller machine.

Volume 54 presents six chapters on the changing face of software engineering-the process by which we build reliable software systems. We are constantly building faster and less expensive processors, which allow us to

Read Book Object Oriented Software Engineering Kung

use different processes to try and conquer the "bug" problem facing all developments-how to build reliable systems with few errors at low or at least manageable cost. The first three chapters of this volume emphasize components and the impact that object-oriented design is having on the program development process (a current "hot topic"). The final three chapters present additional aspects of the software development process, including maintenance, purchasing strategies, and secure outsourcing of scientific computations.

For over 20 years, *Software Engineering: A Practitioner's Approach* has been the best selling

Read Book Object Oriented Software Engineering Kung

guide to software engineering for students and industry professionals alike. The sixth edition continues to lead the way in software engineering. A new Part 4 on Web Engineering presents a complete engineering approach for the analysis, design, and testing of Web Applications, increasingly important for today's students. Additionally, the UML coverage has been enhanced and significantly increased in this new edition. The pedagogy has also been improved in the new edition to include sidebars. They provide information on relevant software tools, specific workflow for specific kinds of projects, and additional information on various topics. Additionally, Pressman

Read Book Object Oriented Software Engineering Kung

provides a running case study called "Safe Home" throughout the book, which provides the application of software engineering to an industry project. New additions to the book also include chapters on the Agile Process Models, Requirements Engineering, and Design Engineering. The book has been completely updated and contains hundreds of new references to software tools that address all important topics in the book. The ancillary material for the book includes an expansion of the case study, which illustrates it with UML diagrams. The On-Line Learning Center includes resources for both instructors and students such as checklists, 700 categorized web references,

Read Book Object Oriented Software Engineering Kung

Powerpoints, a test bank, and a software engineering library-containing over 500 software engineering papers.TAKEAWY HERE IS THE FOLLOWING:1. AGILE PROCESS METHODS ARE COVERED EARLY IN CH. 42. NEW PART ON WEB APPLICATIONS --5 CHAPTERS

Advances in Software Engineering

Testing Object-oriented Systems

International Conferences, ASEA and DRBC 2012,

Held in Conjunction with GST 2012, Jeju Island, Korea,
November 28-December 2, 2012. Proceedings

Encyclopedia of Information Science and Technology

Principles, Heuristics and Best Practices

Read Book Object Oriented Software Engineering Kung

Making Use

Presenting the state of the art in component-based software testing, this cutting-edge resource offers you an in-depth understanding of the current issues, challenges, needs and solutions in this critical area. The book discusses the very latest advances in component-based testing and quality assurance in an accessible tutorial format, making the material easy to comprehend and benefit from no matter what your professional level. important, and how it differs from traditional software testing. From an introduction to software components, testing component-based software and validation methods for software components, to performance testing and

Read Book Object Oriented Software Engineering Kung

measurement, standards and certification and verification of quality for component-based systems, you get a revealing snapshot of the key developments in this area, including important research findings. This volume also serves as a textbook for related courses at the advanced undergraduate or graduate level.

This book contains a collection of selected and revised papers originally presented at the Workshop on Emerging Web Service Technology (WEWST) held in conjunction with the 4th European Conference on Web Services (ECOWS'06) in Zurich, Switzerland, December 2006. It details the latest innovations, developments and results in Web Services research. In addition, the book records the

Read Book Object Oriented Software Engineering Kung

evolution of important ideas emerging in the Web Services field.

Providing all the latest on a topic of extreme commercial relevance, this book contains the refereed proceedings of the 10th International ACM SIGSOFT Symposium on Component-Based Software Engineering, held in Medford, MA, USA in July 2007. The 19 revised full papers presented were carefully reviewed and selected from 89 submissions. The papers feature new trends in global software services and distributed systems architectures to push the limits of established and tested component-based methods, tools and platforms.

"This set of books represents a detailed compendium of

Read Book Object Oriented Software Engineering Kung

authoritative, research-based entries that define the contemporary state of knowledge on technology"--Provided by publisher.

Models, Patterns, and Tools

An Agile:unified Methodology

**Third European Dependable Computing Conference,
Prague, Czech Republic, September 15-17, 1999,**

Proceedings

Information Systems, Technology and Management

Optimizations and Machine Code Generation

Theory and Practice

Addressing various aspects of object-oriented software techniques with respect to their

Read Book Object Oriented Software Engineering Kung

impact on testing, this text argues that the testing of object-oriented software is not restricted to a single phase of software development. The book concentrates heavily on the testing of classes and of components or sub-systems, and a major part is devoted to this subject. C++ is used throughout this book that is intended for software practitioners, managers, researchers, students, or anyone interested in object-oriented technology and its impacts throughout the software engineering life-cycle.

We have described the development of a new

Read Book Object Oriented Software Engineering Kung

micro-payment system, NetPay, featuring different ways of managing electronic money, or e-coins. NetPay provides an off-line, anonymous protocol that supports high-volume, low-cost electronic transactions over the Internet. We developed three kinds of e-wallets to manage coins in a NetPay-based system: a server-side e-wallet allowing multiple computer access to coins; a client-side e-wallet allowing customer PC management of the e-coins, and a cookie-based e-wallet cache to improve performance of the client-side e-wallet communication overhead. Experiences to date with NetPay prototypes

Read Book Object Oriented Software Engineering Kung

have demonstrated it provides an effective micro-payment strategy and customers welcome the ability to manage their electronic coins in different ways. References 1. Dai, X. and Lo, B.: NetPay - An Efficient Protocol for Micropayments on the WWW. Fifth Australian World Wide Web Conference, Australia (1999) 2. Dai, X., Grundy, J. and Lo, B.: Comparing and contrasting micro-payment models for-commerce systems, International Conferences of Info-tech and Info-net (ICII), China (2001) 3. Dai, X., Grundy, J.: Architecture of a Micro-Payment System for Thin-Client Web Applications. In Proceedings of the 2002

Read Book Object Oriented Software Engineering Kung

International Conference on Internet Computing, Las Vegas, CSREA Press, June 24-27, 444--450 4. Dai, X. and Grundy J.: "Customer Perception of a Thin-client Micro-payment System Issues and Experiences", Journal of End User Computing, 15(4), pp 62-77, (2003).

Printed in full color. Faced with a software project of epic proportions? Tired of over-committing and under-delivering? Enter the dojo of the agile samurai, where agile expert Jonathan Rasmusson shows you how to kick-start, execute, and deliver your agile projects. Combining cutting-edge tools with

Read Book Object Oriented Software Engineering Kung

classic agile practices, The Agile Samurai gives you everything you need to deliver something of value every week and make rolling your software into production a non-event. Get ready to kick some software project butt. By learning the ways of the agile samurai you will discover: how to create plans and schedules your customer and your team can believe in what characteristics make a good agile team and how to form your own how to gather requirements in a fraction of the time using agile user stories what to do when you discover your schedule is wrong, and how to look like a pro correcting it how

Read Book Object Oriented Software Engineering Kung

to execute fiercely by leveraging the power of agile software engineering practices By the end of this book you will know everything you need to set up, execute, and successfully deliver agile projects, and have fun along the way. If you're a project lead, this book gives you the tools to set up and lead your agile project from start to finish. If you are an analyst, programmer, tester, usability designer, or project manager, this book gives you the insight and foundation necessary to become a valuable agile team member. The Agile Samurai slices away the fluff and theory that make other books less-than-agile.

Read Book Object Oriented Software Engineering Kung

It's packed with best practices, war stories, plenty of humor and hands-on tutorial exercises that will get you doing the right things, the right way. This book will make a difference.

Based on the needs of the educational community, and the software professional, this book takes a unique approach to teaching software testing. It introduces testing concepts that are managerial, technical, and process oriented, using the Testing Maturity Model (TMM) as a guiding framework. The TMM levels and goals support a structured presentation of fundamental and advanced test-

Read Book Object Oriented Software Engineering Kung

related concepts to the reader. In this context, the interrelationships between theoretical, technical, and managerial concepts become more apparent. In addition, relationships between the testing process, maturity goals, and such key players as managers, testers and client groups are introduced. Topics and features: - Process/engineering-oriented text - Promotes the growth and value of software testing as a profession - Introduces both technical and managerial aspects of testing in a clear and precise style - Uses the TMM framework to introduce testing concepts in a systematic,

Read Book Object Oriented Software Engineering Kung

evolutionary way to facilitate understanding - Describes the role of testing tools and measurements, and how to integrate them into the testing process Graduate students and industry professionals will benefit from the book, which is designed for a graduate course in software testing, software quality assurance, or software validation and verification Moreover, the number of universities with graduate courses that cover this material will grow, given the evolution in software development as an engineering discipline and the creation of degree programs in software engineering.

Read Book Object Oriented Software Engineering Kung

International Symposium on Computer and
Information Sciences

Practical Software Testing

Comprehension, Evaluation, and Evolution

Computer Aided Verification

Testing Object-Oriented Software

The Agile Samurai

**Object-oriented programming increases
software reusability, extensibility,
interoperability, and reliability.**

**Software testing is necessary to realize
these benefits. Software testing aims to
uncover as many programming errors as**

Read Book Object Oriented Software Engineering Kung

possible at a minimum cost. A major challenge to the software engineering community remains how to reduce the cost and improve the quality of software testing. The requirements for testing object-oriented programs differ from those for testing conventional programs. Testing Object-Oriented Software illustrates these differences and discusses object-oriented software testing problems, focusing on the difficulties and challenges testers face. The book provides a general framework for class- and system-level testing and

Read Book Object Oriented Software Engineering Kung

examines object-oriented design criteria and high testability metrics. It offers object-oriented testing techniques, ideas and methods for unit testing, and object-oriented program integration-testing strategy. Readers are shown how they can drastically reduce regression test costs, presented with steps for object-oriented testing, and introduced to object-oriented test tools and systems. In addition to software testing problems, the text covers various test methods developers can use during the design phase to generate

Read Book Object Oriented Software Engineering Kung

programs with good testability. The book's intended audience includes object-oriented program testers, program developers, software project managers, and researchers working with object-oriented testing.

Computer Applications for Software Engineering, Disaster Recovery, and Business Continuity

How Agile Masters Deliver Great Software
5th International Conference on Web Information Systems Engineering, Brisbane, Australia, November 22-24, 2004, Proceedings

Read Book Object Oriented Software Engineering Kung

**Component-Based Software Engineering
Research Anthology on Recent Trends,
Tools, and Implications of Computer
Programming
Engineering and Managing Software
Requirements**