

Libri Ingegneria Software

UML e ingegneria del software dalla teoria alla pratica Tecniche Nuove Ingegneria del software Pearson Italia S.p.a. Lean and Digitize An Integrated Approach to Process Improvement Routledge

"This thoroughly updated text teaches students or industry R & D practitioners to successfully negotiate the terrain for building and maintaining large, complex software systems. The authors introduce the basic skills needed for a developer to apply software engineering techniques. Next, they focus on methods and technologies that enable developers to specify, design, and implement complex systems. Finally, the authors show how to support the system changes throughout the software life cycle."--BOOK

JACKET. Title Summary field provided by Blackwell North America, Inc. All Rights Reserved

Traduzione della seconda edizione inglese, questo testo presenta, con uno stile conciso e accurato, i principi fondamentali dell'ingegneria del software, illustrandone l'applicazione durante le differenti fasi dello sviluppo di un prodotto applicativo. Il leit-motiv che lega la trattazione dei diversi capitoli è l'enfasi che gli autori pongono sull'importanza di un approccio rigoroso e formale. Il libro è pensato sia per i corsi della laurea triennale sia per quelli della laurea specialistica, ma gli argomenti trattati possono essere utilizzati anche per la preparazione di corsi professionali sui vari aspetti dell'ingegneria del software, e consentono anche un percorso di auto-apprendimento. Annotation Supplied by Informazioni Editoriali

Lean and Six Sigma initiatives are designed to enable sustained improvements in your company or organization's efficiency and competitiveness. As with other improvement strategies they are dependent on two things, effective management and your ability to automate or digitize elements of your business process. Lean and Digitize provides you with a convincing picture of each of these elements (process improvement, digitization and the management of both) to help you eliminate waste, improve process and service, and better align your information and communications technology with your strategic objectives. Bernardo Nicoletti analyses and reviews the development of automation and telecommunications systems in the context of quality management and process improvement. He uses case examples to illustrate organizational and management approaches to implementation. These, along with his practical guidance, will help you make sense of the complexity, benefits and interrelations between these different elements. The text shows you on the one hand, how to integrate information and communication systems into your process improvement projects and, on the other, how to align information and communication projects with your quality strategy. Without a holistic approach to technology and quality improvement, your initiatives run the risk of being misdirected or simply running out of steam. Changes of this kind will never be easy but at least if you follow the advice in Lean and Digitize you will significantly increase your chances of success.

* Covers three years of the best essays. * Essays range from technical to humorous, but are always tangible. *

Beautifully written and extremely timely. * Google lists 183,000 links for "Joel on Software". * Spolsky is one of the most popular programmers around today, with legions of followers.

The overwhelming majority of a software system's lifespan is spent in use, not in design or implementation. So, why does conventional wisdom insist that software engineers focus primarily on the design and development of large-scale computing systems? In this collection of essays and articles, key members of Google's Site Reliability Team explain how and why their commitment to the entire lifecycle has enabled the company to successfully build, deploy, monitor, and maintain some of the largest software systems in the world. You'll learn the principles and practices that enable Google engineers to make systems more scalable, reliable, and efficient—lessons directly applicable to your organization. This book is divided into four sections: Introduction—Learn what site reliability engineering is and why it differs from conventional IT industry practices Principles—Examine the patterns, behaviors, and areas of concern that influence the work of a site reliability engineer (SRE) Practices—Understand the theory and practice of an SRE's day-to-day work: building and operating large distributed computing systems Management—Explore Google's best practices for training, communication, and meetings that your organization can use

SQL (Structured Query Language) è un linguaggio di programmazione ideato per gestire e interrogare basi di dati. Questo libro svela gli errori più comuni e i trabocchetti in cui si imbattono gli sviluppatori di software alle prese con i database, spesso considerati a torto come una delle principali fonti di problemi delle applicazioni. Grazie ai suggerimenti pratici e alle tecniche presentate in queste pagine, il lettore avrà in mano gli strumenti per ottenere dai database risultati più efficaci e per trasformare in soluzioni ottimali anche le strutture più complesse. Strutturato in maniera da rispondere alle esigenze sia dei programmatori più esperti sia di quelli alle prime armi, il libro insegna a progettare e costruire un database, a scrivere query sempre più efficaci e a integrare al meglio il linguaggio SQL nel processo di programmazione delle applicazioni. Infine, non mancano indicazioni sulla progettazione di un codice sicuro e resistente agli attacchi esterni.

La domanda, qual è il contributo che la qualità del software e dei sistemi possono apportare al successo del business di un'azienda e come possiamo garantire la giusta qualità dei sistemi e dei prodotti basati sul software, è una domanda retorica. Come in molti altri settori, anche l'industria del software è in continua trasformazione. Le innovazioni e le nuove soluzioni scaturiscono da nuove esigenze provenienti dai mercati e dalla disponibilità di nuove tecnologie. Già da molti anni l'industria del software è influenzata da un proprio alto tasso di innovazione che ha avuto un impatto su tutte le fasi del ciclo di vita di un software e di un sistema. I cambiamenti che vediamo nel software includono anche la gestione e la garanzia della qualità nel suo insieme. Sebbene alcune best practice siano già ora messe in pratica, c'è molto spazio per migliorare. A nostro avviso, non è ancora presente nel settore IT un approccio olistico alla qualità dei prodotti e dei sistemi software. Deve essere definita in maniera chiara la giusta qualità dei sistemi e dei prodotti basati sul software.

Per questo motivo, diamo uno sguardo ai "sistemi integrati" e ai "sistemi IT", impariamo qualcosa da entrambi e discutiamo il nostro approccio per la giusta qualità dei software e dei sistemi (RiSSQ).

La progettazione e la verifica di qualsiasi tipo di struttura, dalla più elementare alla più complessa, vanno orientandosi ormai definitivamente verso l'utilizzo sempre più massiccio e intensivo dei software di calcolo a modellazione tridimensionale, che presentano fasi finali di "post-processing" dai risultati anche molto articolati. Il progettista è quindi proiettato verso una situazione abbastanza complessa in cui la possibilità offertagli dai moderni strumenti di calcolo automatico, se da una parte lo sgrava da migliaia di calcolazioni, dall'altro lo obbliga a una maggiore consapevolezza e conoscenza degli algoritmi di soluzione, forzandolo a una rigorosa analisi critica dei risultati. In questo panorama, il presente lavoro dà evidenza a quanto espressamente richiesto al par. 10 delle NTC: l'adozione e l'utilizzo di semplici formulazioni pratiche che permettano non solo la validazione in sé, ma anche una rapida stima dell'ordine di grandezza di ciò che ci si accinge a progettare o a verificare.

Un'eccellente introduzione agli algoritmi, alla loro struttura, a come modificano i dati, alla computabilità e alla complessità, il libro è scritto in una forma allo stesso tempo elegante e schietto che fa sì che possa essere considerato sia un valido testo per un corso introduttivo di Informatica, sia un tesoro da custodire per i programmatori provetti e i progettisti di software.

Questo è il primo libro sulle presentazioni che classifica e organizza i pattern, i mattoni, che è necessario saper padroneggiare per comunicare in modo efficace con strumenti come PowerPoint e Keynote. I pattern sono modelli che, una volta appresi, potrete mescolare e utilizzare come preferite per costruire presentazioni avvincenti nei più svariati contesti, lavorativi e non: dalle riunioni commerciali alle dimostrazioni tecniche, passando per le esposizioni accademiche. Il testo, ripercorrendo le fasi della creazione di una presentazione – pianificazione, realizzazione, esposizione – insegna a mettere a fuoco il messaggio, relazionarsi con il pubblico, affrontare gli imprevisti ed evitare le trappole e gli errori più frequenti (i cosiddetti antipattern). L'originale punto di vista degli autori, maturato nel campo dello sviluppo software, presto diventerà indispensabile sia che leggete questa guida dall'inizio alla fine, sia che decidiate di consultarla al volo, per migliorare il contenuto e l'impatto della vostra ultima presentazione poco prima della consegna.

Il Convitato di Vetro è il software, impiegato ubiquamente dagli ingegneri almeno da vent'anni, ma sconosciuto ai più, ed in specie ai normatori, nelle sue caratteristiche logiche e formali essenziali. In questo libro Roberto Spagnuolo, grande esperto di sviluppo software per l'ingegneria strutturale e titolare di una nota software house italiana, ci fa vedere come sia illusorio credere che il software ed il formalismo matematico ad esso spesso associato possano cancellare le incertezze e i problemi, sostituendo ad essi certezze che di fatto poi si rivelano come illusioni. Il libro ripercorre anche l'esperienza del suo autore dalla metà degli anni '80 sino ai giorni nostri, ed è un racconto godibilissimo e a tratti esilarante, sempre molto intenso e pieno di riferimenti originali, qualche volta utilmente provocatorio, della progressiva trasformazione del software da nuovo strumento tutto da esplorare, quale era all'inizio degli anni '80, a predittore assoluto di precisioni impossibili ed inesistenti quale è percepito in questi anni. Molto interessante è anche la critica all'abuso del linguaggio matematico e formale, visto come possibile metodo per dare crisma di

infallibilità ad ipotesi a volte azzardate, se non addirittura infondate. Ne consegue una profonda riflessione sull'impianto che è stato dato alle normative più recenti, anche alla luce della sostanziale insostituibilità del Convitato di Vetro. La lettura di questo libro può molto aiutare i non addetti ai lavori a comprendere meglio la vasta gamma di problemi e di questioni che sono associate allo sviluppo software, favorendone quindi indirettamente un uso più attento e consapevole.

“Il futuro è sotto i nostri occhi e riguarda il declino del lavoro manuale a favore del lavoro creativo nei diversi campi delle tecnologie dell'informazione e della comunicazione, ma soprattutto nello sviluppo del software, motore insostituibile di tutte le tecnologie hardware esistenti. Dunque, il primo obiettivo è spostare il fuoco dell'attenzione di chi decide le sorti del Paese e dell'Europa dall'universo materiale a quello immateriale, cioè dalle tecnologie hardware alle tecnologie software. Infatti, mentre le piattaforme hardware tendono a diventare commodity, è la tecnologia software il componente vitale e critico di ogni sistema reale, sia perché rappresenta la quota maggioritaria dei costi di progettazione, realizzazione e gestione dei sistemi complessi, sia perché l'industria del software, anche per ritardi e difficoltà nell'ingegneria del processo produttivo, è ad alta intensità di lavoro e di conoscenza e quindi rappresenta la principale opportunità di lavoro per le giovani generazioni dei Paesi sviluppati e anche dei Paesi in ritardo di sviluppo, ma con un sistema formativo evoluto”. Perché torno a scrivere? Che contributo posso ancora dare che sia utile e costruttivo? L'età avanzata mi obbliga a trovare la forza per uscire dal silenzio e dalla rassegnazione. Nel corso della mia lunga vita professionale sono convinto di essere entrato in contatto con le persone, le aziende e le istituzioni che hanno costruito e poi smantellato l'Italia informatica, quindi di aver preso cognizione degli errori compiuti nel passato e anche, in parte, delle possibili vie d'uscita verso il futuro. Occorre recuperare il tempo perduto, affinché i treni tecnologici si possano fermare anche alla stazione Italia.

"This book manages to convey the practical use of UML 2 in clear and understandable terms with many examples and guidelines. Even for people not working with the Unified Process, the book is still of great use. UML 2 and the Unified Process, Second Edition is a must-read for every UML 2 beginner and a helpful guide and reference for the experienced practitioner." --Roland Leibundgut, Technical Director, Zuehlke Engineering Ltd. "This book is a good starting point for organizations and individuals who are adopting UP and need to understand how to provide visualization of the different aspects needed to satisfy it. " --Eric Naiburg, Market Manager, Desktop Products, IBM Rational Software This thoroughly revised edition provides an indispensable and practical guide to the complex process of object-oriented analysis and design using UML 2. It describes how the process of OO analysis and design fits into the software development lifecycle as defined by the Unified Process (UP). UML 2 and the Unified Process contains a wealth of practical, powerful, and useful techniques that you can apply immediately. As you progress through the text, you will learn OO analysis and

design techniques, UML syntax and semantics, and the relevant aspects of the UP. The book provides you with an accurate and succinct summary of both UML and UP from the point of view of the OO analyst and designer. This book provides Chapter roadmaps, detailed diagrams, and margin notes allowing you to focus on your needs Outline summaries for each chapter, making it ideal for revision, and a comprehensive index that can be used as a reference New to this edition: Completely revised and updated for UML 2 syntax Easy to understand explanations of the new UML 2 semantics More real-world examples A new section on the Object Constraint Language (OCL) Introductory material on the OMG's Model Driven Architecture (MDA) The accompanying website provides A complete example of a simple e-commerce system Open source tools for requirements engineering and use case modeling Industrial-strength UML course materials based on the book

For courses in computer science and software engineering The Fundamental Practice of Software Engineering Software Engineering introduces students to the overwhelmingly important subject of software programming and development. In the past few years, computer systems have come to dominate not just our technological growth, but the foundations of our world's major industries. This text seeks to lay out the fundamental concepts of this huge and continually growing subject area in a clear and comprehensive manner. The Tenth Edition contains new information that highlights various technological updates of recent years, providing students with highly relevant and current information. Sommerville's experience in system dependability and systems engineering guides the text through a traditional plan-based approach that incorporates some novel agile methods. The text strives to teach the innovators of tomorrow how to create software that will make our world a better, safer, and more advanced place to live.

"Unified Modeling Language (UML), Unified Process (UP), and other information modeling methods are addressed in this scholarly consideration of the analysis, design, and development of web-based and enterprise applications. The most current research on conceptual, theoretical, and empirical issues of modeling for online business and static information is provided."

[Copyright: 32ac4fc604e2448d69729c267470edd7](https://www.pdfdrive.com/unified-modeling-language-unified-process-and-other-information-modeling-methods-are-addressed-in-this-scholarly-consideration-of-the-analysis-design-and-development-of-web-based-and-enterprise-applications-p123456789.html)