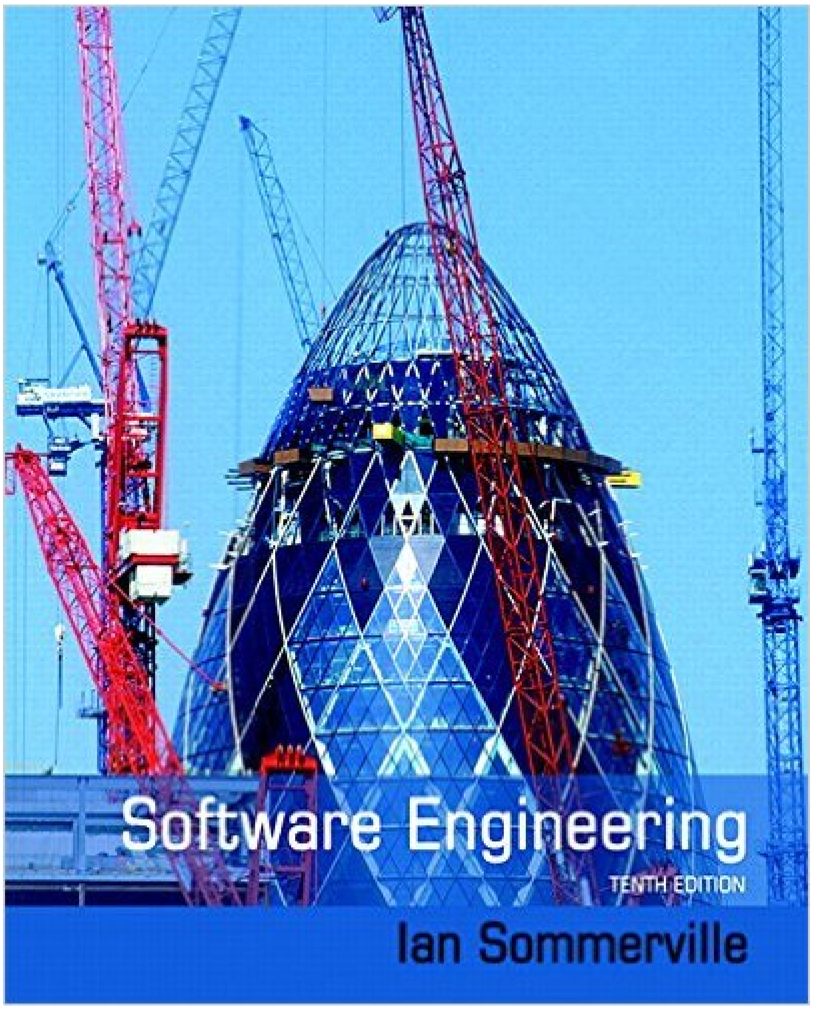
Software Engineering (10th Edition)



Synopsis

For courses in computer science and software engineering Â The Fundamental Practice of Software Engineering Software Engineering introduces readers 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 worldâ ™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. Â The Tenth Edition contains new information that highlights various technological updates of recent years, providing readers with highly relevant and current information. Sommervilleâ ™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.

Book Information

Hardcover: 816 pages

Publisher: Pearson; 10 edition (April 3, 2015)

Language: English

ISBN-10: 0133943038

ISBN-13: 978-0133943030

Product Dimensions: 7.6 x 1.2 x 9.1 inches

Shipping Weight: 2.8 pounds (View shipping rates and policies)

Average Customer Review: 3.3 out of 5 starsÂ Â See all reviewsÂ (46 customer reviews)

Best Sellers Rank: #20,911 in Books (See Top 100 in Books) #19 inÂ Books > Textbooks > Computer Science > Software Design & Engineering #34 inÂ Books > Computers & Technology > Programming > Software Design, Testing & Engineering > Software Development #81 inÂ Books > Textbooks > Computer Science > Programming Languages

Customer Reviews

I had the displeasure of using this book for a graduate level course on Software Engineering. We did not use the web site or the online learning tools associated with the book. The professor did use the author’s supplied slides.Pros:* Very nice UML diagrams.* Might get recycled into toilet paper.Cons:* Very repetitive. I lost track of how many times a legacy system was defined. This made reading this book extremely boring.* Despite the 2011 copyright date, the material is dated. Computers are still single core, smart phones aren’t on the scene, and Sun still owns Java (cue

Oracle lawyers).* The back of the book claims it has been updated with new material on open source development. That new material consists of a few paragraphs on the legal issues of incorporating open source into a traditional project. There is nothing on developing software for open source.* The power point slides that accompany the book have problems with the graphics starting about chapter 5. The image quality of the embedded diagrams takes a nose dive and the images are barely readable.* The topics covered seemed very shallow. I'm not sure you'll get much more out of this book then you would reading through wikipedia articles.* The author has a habit of using acronyms without defining them. COTS (Commercial Off The Shelf) was used for several chapters before it was spelled out.* Electronic version (Kindle) has random spaces removed (as noted by other reviewers)* The project schedule charts presented are Gantt charts. It seems the author has never heard of Gantt and just refers to them as bar charts.* Some diagrams are mislabeled.* There was at least one sample XML file that was used for a few problems.

Download to continue reading...

Non-Functional Requirements in Software Engineering (International Series in Software Engineering) Software Engineering Classics: Software Project Survival Guide/ Debugging the Development Process/ Dynamics of Software Development (Programming/General) Software Engineering (10th Edition) Software Architecture in Practice (3rd Edition) (SEI Series in Software Engineering) Software Components With Ada: Structures, Tools, and Subsystems (The Benjamin/Cummings Series in Ada and Software Engineering) Global Software Development Handbook (Applied Software Engineering Series) Software Failure: Management Failure: Amazing Stories and Cautionary Tales (Wiley Series in Software Engineering Practice) Error-Free Software: Know-How and Know-Why of Program Correctness (Wiley Series in Software Engineering Practice) Constraint-Based Design Recovery for Software Reengineering: Theory and Experiments (International Series in Software Engineering) Re-Engineering Software: How to Re-Use Programming to Build New, State-of-the-Art Software Practical Software Reuse (Wiley Series in Software Engineering Practice) Object-oriented software development: Engineering software for reuse Software Reuse: Guidelines and Methods (Software Science and Engineering) Enterprise Software Platform: A Textbook for Software Engineering Students Object-Oriented Software Engineering: Practical Software Development Using UML and Java Surreptitious Software: Obfuscation, Watermarking, and Tamperproofing for Software Protection: Obfuscation, Watermarking, and Tamperproofing for Software Protection Object-Oriented Software Engineering Using UML, Patterns, and Java (3rd Edition) [Economy Edition] Energy Audit of Building Systems: An Engineering Approach, Second Edition (Mechanical and Aerospace Engineering Series) Orbital