Synopsis

For courses in Software Engineering, Software Development, or Object-Oriented Design and Analysis at the Junior/Senior or Graduate level. This text can also be utilized in short technical courses or in short, intensive management courses. Object-Oriented Software Engineering Using UML, Patterns, and Java, 3e, shows readers how to use both the principles of software engineering and the practices of various object-oriented tools, processes, and products. Using a step-by-step case study to illustrate the concepts and topics in each chapter, Bruegge and Dutoit emphasize learning object-oriented software engineer through practical experience: readers can apply the techniques learned in class by implementing a real-world software project. The third edition addresses new trends, in particular agile project management (Chapter 14 Project Management) and agile methodologies (Chapter 16 Methodologies).

Book Information

Hardcover: 800 pages
Publisher: Pearson; 3 edition (August 8, 2009)
Language: English
ISBN-10: 0136061257
Product Dimensions: 7.2 x 1.8 x 9.3 inches
Shipping Weight: 3.2 pounds (View shipping rates and policies)
Average Customer Review: 3.8 out of 5 stars Â See all reviews (25 customer reviews)

Customer Reviews

This is NOT a book on Unified Modeling Language (UML). It’s not a book on Object Constraint Language (OCL). It’s also not a book on Capability Maturity Models (CMM), Class-Responsibilities-Collaborators (CRC) cards, Decision Representation Language (DRL), Extreme Programming (XP), Gantt charts, Issue-Based Information Systems (IBIS), Joint Application Design (JAD), Key Process Areas (KPA), the Liskov Substitution Principle, Model-View-Controller (MVC) architectural styles, Nonfunctional Requirements (NFR) Frameworks, Object Design Documents (ODD), PERT charts, the Questions-Options-Criteria (QOC) model, Requirements
Analysis Documents (RAD), Royce's methodology, Software Configuration Management Plans (SCMP), System Design Documents (SDD), Software Project Management Plans (SPMP), the Unified Software Development Process, User Manuals, V-Models, Work Breakdown Structures (WBS), or any of the myriad other tools introduced in the book. This IS a book to introduce newly-minted programmers to the kind of things, tools, and processes they can look forward to (with either anticipation or dread) in the real world of software development. As the authors state on page viii of the Preface: "We have observed that students are taught programming and software engineering techniques in isolation, often using small problems as examples. As a result, they are able to solve well-defined problems efficiently, but are overwhelmed by the complexity of their first real development experience, when many different techniques and tools need to be used and different people need to collaborate." It's been many years since I was involved in major software development projects (and those were all in the military).

Download to continue reading...
