A Discipline For Software Engineering is written by Watts S. Humphrey in English language. Release on 1995-01-10, this book has 816 page count that enclose valuable information with lovely reading experience. The book was publish by Addison-Wesley Professional, it is one of best computers & technology book genre that gave you everything love about reading. You can find A Discipline For Software Engineering book with ISBN 0201546108.
This book from Watts Humphrey broadens his disciplined approach to software engineering. In his earlier book, Managing the Software Process, Humphrey developed concrete methods for managing software development and maintenance. These methods, now commonly practiced, provide programmers and managers specific steps for evaluating and improving their software capabilities. In this book, he scales down those methods to a personal level, helping software practitioners develop the skills and habits they need to plan, track, and analyze large and complex projects more carefully and successfully.

**Discipline Software Engineering Watts Humphrey Related Books**

**Watts My Line?: The Life and Work of Editorial Artist, Lawrie Watts**

This is the story of Lawrie Watts and his amazing technical artworks, illustrations, and cutaway drawings of motorcycles, motorcars, aircraft, and farm machinery. He was drawing amazingly complex machinery with meticulous attention to detail way before the development of CAD. Lawrie is not just an artist; he's a designer too. An example of his designs was the Enfield-powered Dreamliner.


Software Engineering: The Current Practice teaches students basic software engineering skills and helps practitioners refresh their knowledge and explore recent developments in the field, including software changes and iterative processes of software development. After a historical overview and an introduction to software technology and models, the book discusses the software change and its phases, including concept location, impact analysis, refactoring, actualization, and verification. It then...

**Software Architecture: Perspectives on an Emerging Discipline**

Good software developers often adopt one or several architectural patterns as strategies for system organization. But, although they use these patterns purposefully, they often use them informally and nearly unconsciously. This book organizes this substantial emerging "folklore" of system design -- with its rich language of system description -- and closes the gap between the useful abstractions (constructs and patterns) of system design and the current models, notations and tools. ...

**Fundamentals of Dependable Computing for Software Engineers (Chapman & Hall/CRC Innovations in Software Engineering and Software Development Series)**

Fundamentals of Dependable Computing for Software Engineers presents the essential elements of computer system dependability. The book describes a comprehensive dependability-engineering process and explains the roles of software and software engineers in computer system dependability. Readers will learn: Why dependability matters What it means for a system to be dependable How to build a dependable software system How to assess whether a software system is adequately dependable The author...

**Model-Driven Software Engineering in Practice (Synthesis Lectures on Software Engineering)**

This book discusses how model-based approaches can improve the daily practice of software professionals. This is known as Model-Driven Software Engineering (MDSE) or, simply, Model-Driven Engineering (MDE). MDSE practices have proved to increase efficiency and effectiveness in software development, as demonstrated by various quantitative and qualitative studies. MDSE adoption in the software industry is foreseen to grow exponentially in the near future, e.g., due to the convergence of software d...


Taking a learn-by-doing approach, Software Engineering Design: Theory and Practice uses examples, review questions, chapter exercises, and case study assignments to provide students and practitioners with the understanding required to design complex software systems. Explaining the concepts that are immediately relevant to software designers, it begins with a review of software design fundamentals. The text presents a formal top-down design process that consists of several design activities with ...
Object-Oriented Software Engineering: Practical Software Development Using UML and Java

The authors' focus in this book is to deliver software engineering knowledge and skills that readers can put into immediate practical use. The book provides the essential topic coverage required by students of software engineering, from the nuts and bolts of objects to software architecture, from writing code to testing, from software development processes to project management. Working through nine contemporary themes in Software Engineering, students are given an awareness of key issues from u...


Emphasizing leadership principles and practices, Antipatterns: Managing Software Organizations and People, Second Edition catalogs 49 business practices that are often precursors to failure. This updated edition of a bestseller not only illustrates bad management approaches, but also covers the bad work environments and cultural traits commonly found in IT, software development, and other business domains. For each antipattern, it describes the situation and symptoms, gives examples, and offers ...


The award-winning and highly influential Software Architecture in Practice, Third Edition, has been substantially revised to reflect the latest developments in the field. In a real-world setting, the book once again introduces the concepts and best practices of software architecture—how a software system is structured and how that systems elements are meant to interact. Distinct from the details of implementation, algorithm, and data representation, an architecture holds the key to achieving ...

Data Structures and Algorithms (Software Engineering and Knowledge Engineering, 13)

This is an excellent, up-to-date and easy-to-use text on data structures and algorithms that is intended for undergraduates in computer science and information science. The thirteen chapters, written by an international group of experienced teachers, cover the fundamental concepts of algorithms and most of the important data structures as well as the concept of interface design. The book contains many examples and diagrams. Whenever appropriate, program codes are included to facilitate learning....