

软件工程



[软件工程_下载链接1](#)

著者:(美)弗莱格//阿特利

出版者:高等教育

出版时间:2009-12

装帧:

isbn:9787040279474

《国外优秀信息科学与技术系列教学用书·软件工程:理论与实践(第4版·影印版)》主要内容: This introduction to software engineering and practice addresses both procedural and object-oriented development. The book applies concepts consistently to two common examples — a typical information system and a real-time system. It combines theory with real, practical applications by providing an abundance of case studies and examples from the current literature. This revision has been thoroughly updated to reflect significant changes in software engineering, including modeling and agile methods.

作者介绍:

Shaft Lawrence Pfleeger (Ph.D., Information Technology and Engineering, George Mason University; M.S., Planning, Pennsylvania State University; M.A., Mathematics, Pennsylvania State University; B.A., Mathematics, Harpur College) is a senior information scientist at the RAND Corporation. Her current research focuses on policy and decision-making issues that help organizations and government agencies understand whether and how information technology supports their missions and

goals. Her work at RAND has involved assisting clients in creating software measurement programs, supporting government agencies in defining information assurance policies, and supporting decisions about cyber security and homeland security.

Joanne M. Atlee (Ph.D. and M.S., Computer Science, University of Maryland; B.S., Computer Science and Physics, College of William and Mary; P.Eng.) is an Associate Professor in the School of Computer Science at the University of Waterloo. Her research focuses on software modeling, documentation, and analysis. She is best known for her work on model checking software requirements specifications. Other research interests include model-based software engineering, modular software development, feature interactions, and cost-benefit analysis of formal software development techniques. Atlee serves on the editorial boards for IEEE Transactions on Software Engineering, Software and Systems Modeling, and the Requirements Engineering Journal and is Vice Chair of the International Federation for Information Processing (IFIP) Working Group 2.9, an international group of researchers working on advances in software requirements engineering. She is Program Co-Chair for the 31st International Conference on Software Engineering (ICSE'09).

目录: Preface About the Authors 1 Why Software Engineering? 1.1 What Is Software Engineering? 1.2 How Successful Have We Been? 1.3 What Is Good Software? 1.4 Who Does Software Engineering? 1.5 A Systems Approach 1.6 An Engineering Approach 1.7 Members of the Development Team 1.8 How Has Software Engineering Changed? 1.9 Information Systems Example 1.10 Real-Time Example 1.11 What This Chapter Means for You 1.12 What This Chapter Means for Your Development Team 1.13 What This Chapter Means for Researchers 1.14 Term Project 1.15 Key References 1.16 Exercises 2 Modeling the Process and Life Cycle 2.1 The Meaning of Process 2.2 Software Process Models 2.3 Tools and Techniques for Process Modeling 2.4 Practical Process Modeling 2.5 Information Systems Example 2.6 Real-Time Example 2.7 What This Chapter Means for You 2.8 What This Chapter Means for Your Development Team 2.9 What This Chapter Means for Researchers 2.10 Term Project 2.11 Key References 2.12 Exercises 3 Planning and Managing the Project 3.1 Tracking Progress 3.2 Project Personnel 3.3 Effort Estimation 3.4 Risk Management 3.5 The Project Plan 3.6 Process Models and Project Management 3.7 Information Systems Example 3.8 Real-Time Example 3.9 What This Chapter Means for You 3.10 What This Chapter Means for Your Development Team 3.11 What This Chapter Means for Researchers 3.12 Term Project 3.13 Key References 3.14 Exercises 4 Capturing the Requirements 4.1 The Requirements Process 4.2 Requirements Elicitation 4.3 Types of Requirements 4.4 Characteristics of Requirements 4.5 Modeling Notations 4.6 Requirements and Specification Languages 4.7 Prototyping Requirements 4.8 Requirements Documentation 4.9 Validation and Verification 4.10 Measuring Requirements 4.11 Choosing a Specification Technique 4.12 Information Systems Example 4.13 Real-Time Example 4.14 What This Chapter Means for You 4.15 What This Chapter Means for Your Development Team 4.16 What This Chapter Means for Researchers 4.17 Term Project 4.18 Key References 4.19 Exercises 5 Designing the Architecture 5.1 The Design Process 5.2 Modeling Architectures 5.3 Decomposition and Views 5.4 Architectural Styles and Strategies 5.5 Achieving Quality Attributes 5.6 Collaborative Design 5.7 Architecture Evaluation and Refinement 5.8 Documenting Software Architectures 5.9 Architecture Design Review 5.10 Software Product Lines 5.11 Information Systems Example 5.12 Real-Time Example 5.13 What This Chapter Means for You 5.14 What This Chapter Means for Your Development Team 5.15 What This Chapter Means for Researchers 5.16 Term Project 5.17 Key References 5.18 Exercises 6 Designing the Modules 6.1 Design Methodology 6.2 Design Principles 6.3 OO Design 6.4 Representing OO Designs in the UML 6.5 OO Design

Patterns 6.6 Other Design Considerations 6.7 OO Measurement 6.8 Design Documentation 6.9 Information Systems Example 6.10 Real-Time Example 6.11 What This Chapter Means for You 6.12 What This Chapter Means for Your Development Team 6.13 What This Chapter Means for Researchers 6.14 Term Project 6.15 Key References 6.16 Exercises7 Writing the Programs 7.1 Programming Standards and Procedures 7.2 Programming Guidelines 7.3 Documentation 7.4 The Programming Process 7.5 Information Systems Example 7.6 Real-Time Example 7.7 What This Chapter Means for You 7.8 What This Chapter Means for Your Development Team 7.9 What This Chapter Means for Researchers 7.10 Term Project 7.11 Key References 7.12 Exercises8 Testing the Programs 8.1 Software Faults and Failures 8.2 Testing Issues 8.3 Unit Testing 8.4 Integration Testing 8.5 Testing Object-Oriented Systems 8.6 Test Planning 8.7 Automated Testing Tools 8.8 When to Stop Testing 8.9 Information Systems Example 8.10 Real-Time Example 8.11 What This Chapter Means for You 8.12 What This Chapter Means for Your Development Team 8.13 What This Chapter Means for Researchers 8.14 Term Project 8.15 Key References 8.16 Exercises9 Testing the System 9.1 Principles of System Testing 9.2 Function Testing 9.3 Performance Testing 9.4 Reliability, Availability, and Maintainability 9.5 Acceptance Testing 9.6 Installation Testing 9.7 Automated System Testing 9.8 Test Documentation 9.9 Testing Safety-Critical Systems 9.10 Information Systems Example 9.11 Real-Time Example 9.12 What This Chapter Means for You 9.13 What This Chapter Means for Your Development Team 9.14 What This Chapter Means for Researchers 9.15 Term Project 9.16 Key References 9.17 Exercises10 Delivering the System 10.1 Training 10.2 Documentation 10.3 Information Systems Example 10.4 Real-Time Example 10.5 What This Chapter Means for You 10.6 What This Chapter Means for Your Development Team 10.7 What This Chapter Means for Researchers 10.8 Term Project 10.9 Key References 10.10 Exercises11 Maintaining the System 11.1 The Changing System 11.2 The Nature of Maintenance 11.3 Maintenance Problems 11.4 Measuring Maintenance Characteristics 11.5 Maintenance Techniques and Tools 11.6 Software Rejuvenation 11.7 Information Systems Example 11.8 Real-Time Example 11.9 What This Chapter Means for You 11.10 What This Chapter Means for Your Development Team 11.11 What This Chapter Means for Researchers 11.12 Term Project 11.13 Key References 11.14 Exercises12 Evaluating Products, Processes, and Resources 12.1 Approaches to Evaluation 12.2 Selecting an Evaluation Technique 12.3 Assessment vs. Prediction 12.4 Evaluating Products 12.5 Evaluating Processes 12.6 Evaluating Resources 12.7 Information Systems Example 12.8 Real-Time Example 12.9 What This Chapter Means for You 12.10 What This Chapter Means for Your Development Team 12.11 What This Chapter Means for Researchers 12.12 Term Project 12.13 Key References 12.14 Exercises13 Improving Predictions, Products, Processes, and Resources 13.1 Improving Predictions 13.2 Improving Products 13.3 Improving Processes 13.4 Improving Resources 13.5 General Improvement Guidelines 13.6 Information Systems Example 13.7 Real-Time Example 13.8 What This Chapter Means for You 13.9 What This Chapter Means for Your Development Team 13.10 What This Chapter Means for Researchers 13.11 Term Project 13.12 Key References 13.13 Exercises14 The Future of Software Engineering 14.1 How Have We Done? 14.2 Technology Transfer 14.3 Decision-Making in Software Engineering 14.4 The Professionalization of Software Engineering: Licensing, Certification, and Ethics 14.5 Term Project 14.6 Key References 14.7 ExercisesAnnotated BibliographyIndex
• • • • • (收起)

[软件工程_下载链接1](#)

标签

计算机

评论

[软件工程_下载链接1](#)

书评

[软件工程_下载链接1](#)