International Edition

Software Engineering

Fourth Edition

Shari Lawrence Pfleeger • Joanne M. Atlee

PEARSON

Contents

Preface 13

About the Authors 23

1 Why Software Engineering? **27**
1.1 What Is Software Engineering? 28
1.2 How Successful Have We Been? 31
1.3 What Is Good Software? 35
1.4 Who Does Software Engineering? 40
1.5 A Systems Approach 42
1.6 An Engineering Approach 47
1.7 Members of the Development Team 51
1.8 How Has Software Engineering Changed? 53
1.9 Information Systems Example 61
1.10 Real-Time Example 63
1.11 What This Chapter Means for You 65
1.12 What This Chapter Means for Your Development Team 66
1.13 What This Chapter Means for Researchers 66
1.14 Term Project 66
1.15 Key References 68
1.16 Exercises 69

2 Modeling the Process and Life Cycle **71**
2.1 The Meaning of Process 71
2.2 Software Process Models 74
2.3 Tools and Techniques for Process Modeling 89
2.4 Practical Process Modeling 96
2.5 Information Systems Example 99
2.6 Real-Time Example 101
2.7 What This Chapter Means for You 102
2.8 What This Chapter Means for Your Development Team 102
2.9 What This Chapter Means for Researchers 103
2.10 Term Project 103
2.11 Key References 103
2.12 Exercises 107

3 Planning and Managing the Project **108**
3.1 Tracking Progress 108
3.2 Project Personnel 121
3.3 Effort Estimation 130
3.4 Risk Management 145
3.5 The Project Plan 149
3.6 Process Models and Project Management 151
3.7 Information Systems Example 159
3.8 Real-Time Example 161
3.9 What This Chapter Means for You 162
3.10 What This Chapter Means for Your Development Team 162
3.11 What This Chapter Means for Researchers 163
3.12 Term Project 163
3.13 Key References 163
3.14 Exercises 164

4 Capturing the Requirements **167**
4.1 The Requirements Process 168
4.2 Requirements Elicitation 170
4.3 Types of Requirements 174
4.4 Characteristics of Requirements 181
4.5 Modeling Notations 182
4.6 Requirements and Specification Languages 213
4.7 Prototyping Requirements 217
4.8 Requirements Documentation 219
4.9 Validation and Verification 224
4.10 Measuring Requirements 230
4.11 Choosing a Specification Technique 232
4.12 Information Systems Example 235
4.13 Real-Time Example 237
4.14 What This Chapter Means for You 239
4.15 What This Chapter Means for Your Development Team 240
4.16 What This Chapter Means for Researchers 240
4.17 Term Project 241
4.18 Key References 244
4.19 Exercises 245

5 Designing the Architecture **249**
5.1 The Design Process 249
5.2 Modeling Architectures 257
5.3 Decomposition and Views 257
5.4 Architectural Styles and Strategies 261
5.5 Achieving Quality Attributes 271
5.6 Collaborative Design 282
5.7 Architecture Evaluation and Refinement 284
5.8 Documenting Software Architectures 299
5.9 Architecture Design Review 302

5.10 Software Product Lines 305
5.11 Information Systems Example 310
5.12 Real-Time Example 312
5.13 What This Chapter Means for You 313
5.14 What This Chapter Means for Your Development Team 314
5.15 What This Chapter Means for Researchers 314
5.16 Term Project 315
5.17 Key References 316
5.18 Exercises 316

6 Designing the Modules 319
6.1 Design Methodology 320
6.2 Design Principles 321
6.3 OO Design 336
6.4 Representing OO Designs in the UML 347
6.5 OO Design Patterns 364
6.6 Other Design Considerations 374
6.7 OO Measurement 379
6.8 Design Documentation 389
6.9 Information Systems Example 392
6.10 Real-Time Example 393
6.11 What This Chapter Means for You 393
6.12 What This Chapter Means for Your Development Team 395
6.13 What This Chapter Means for Researchers 395
6.14 Term Project 395
6.15 Key References 396
6.16 Exercises 396

7 Writing the Programs 399
7.1 Programming Standards and Procedures 399
7.2 Programming Guidelines 402
7.3 Documentation 413
7.4 The Programming Process 417
7.5 Information Systems Example 421
7.6 Real-Time Example 422
7.7 What This Chapter Means for You 423
7.8 What This Chapter Means for Your Development Team 424
7.9 What This Chapter Means for Researchers 424
7.10 Term Project 424
7.11 Key References 425
7.12 Exercises 425

8 Testing the Programs 427
8.1 Software Faults and Failures 427
8.2 Testing Issues 433
8.3 Unit Testing 438
8.4 Integration Testing 452

8.5 Testing Object-Oriented Systems 459
8.6 Test Planning 462
8.7 Automated Testing Tools 464
8.8 When to Stop Testing 467
8.9 Information Systems Example 472
8.10 Real-Time Example 473
8.11 What This Chapter Means for You 474
8.12 What This Chapter Means for Your Development Team 474
8.13 What This Chapter Means for Researchers 475
8.14 Term Project 475
8.15 Key References 475
8.16 Exercises 476

9 Testing the System 479
9.1 Principles of System Testing 479
9.2 Function Testing 492
9.3 Performance Testing 498
9.4 Reliability, Availability, and Maintainability 500
9.5 Acceptance Testing 509
9.6 Installation Testing 512
9.7 Automated System Testing 512
9.8 Test Documentation 513
9.9 Testing Safety-Critical Systems 524
9.10 Information Systems Example 535
9.11 Real-Time Example 537
9.12 What This Chapter Means for You 539
9.13 What This Chapter Means for Your Development Team 539
9.14 What This Chapter Means for Researchers 539
9.15 Term Project 540
9.16 Key References 540
9.17 Exercises 541

10 Delivering the System 545
10.1 Training 545
10.2 Documentation 550
10.3 Information Systems Example 557
10.4 Real-Time Example 558
10.5 What This Chapter Means for You 558
10.6 What This Chapter Means for Your Development Team 558
10.7 What This Chapter Means for Researchers 559
10.8 Term Project 559
10.9 Key References 559
10.10 Exercises 559

11 Maintaining the System **561**
11.1 The Changing System 561
11.2 The Nature of Maintenance 569
11.3 Maintenance Problems 572
11.4 Measuring Maintenance Characteristics 580
11.5 Maintenance Techniques and Tools 586
11.6 Software Rejuvenation 594
11.7 Information Systems Example 600
11.8 Real-Time Example 601
11.9 What This Chapter Means for You 602
11.10 What This Chapter Means for Your Development Team 602
11.11 What This Chapter Means for Researchers 603
11.12 Term Project 603
11.13 Key References 603
11.14 Exercises 604

12 Evaluating Products, Processes, and Resources **606**
12.1 Approaches to Evaluation 606
12.2 Selecting an Evaluation Technique 611
12.3 Assessment vs. Prediction 615
12.4 Evaluating Products 621
12.5 Evaluating Processes 636
12.6 Evaluating Resources 652
12.7 Information Systems Example 657
12.8 Real-Time Example 658
12.9 What This Chapter Means for You 658
12.10 What This Chapter Means for Your Development Team 659
12.11 What This Chapter Means for Researchers 659
12.12 Term Project 659
12.13 Key References 660
12.14 Exercises 660

13 Improving Predictions, Products, Processes, and Resources **662**
13.1 Improving Predictions 662
13.2 Improving Products 671
13.3 Improving Processes 675
13.4 Improving Resources 683
13.5 General Improvement Guidelines 686
13.6 Information Systems Example 687
13.7 Real-Time Example 688
13.8 What This Chapter Means for You 688
13.9 What This Chapter Means for Your Development Team 688
13.10 What This Chapter Means for Researchers 689
13.11 Term Project 689

13.12 Key References 689
13.13 Exercises 690

14 The Future of Software Engineering **691**
14.1 How Have We Done? 691
14.2 Technology Transfer 694
14.3 Decision-Making in Software Engineering 702
14.4 The Professionalization of Software Engineering: Licensing, Certification, and Ethics 715
14.5 Term Project 732
14.6 Key References 732
14.7 Exercises 734

Annotated Bibliography 735

Index 771