## 1. What Is an Agile Methodology?

1.1 Rapid Business Change: The Ultimate Driver 1
1.2 What Must Agile Methodologies be Able to Do? 2
1.3 Agility: What Is It and How Do We Achieve It? 2
1.4 Evolving Software: Obstacles and Possibilities 5
1.5 The Quality Agenda 6
1.6 Do We Really Need All This Mountain of Documentation? 9
1.7 The Human Factor 10
1.8 Some Agile Methodologies 11
   1.8.1 Dynamic Systems Development Method 12
   1.8.2 Feature-Driven Design 13
   1.8.3 Crystal 14
   1.8.4 Agile Modeling 14
   1.8.5 SCRUM 15
   1.8.6 Summary Table 15
1.9 Review 16

## Exercise

## Conundrum

## References

## 2. Extreme Programming Outlined

2.1 Some Guiding Principles 19
2.2 The Five Values 20
   2.2.1 Communication 20
   2.2.2 Feedback 22
   2.2.3 Simplicity 24
   2.2.4 Courage 24
   2.2.5 Respect 25
2.3 The 12 Basic Practices of XP 25
   2.3.1 Test-First Programming 25
   2.3.2 Pair Programming 26
   2.3.3 On-Site Customer 27
   2.3.4 The Planning Game 28
3. Foundations: People and Teams Working Together

3.1 Software Engineering in Teams 41
3.2 Personalities and Team Success 42
3.3 Observations of Team Behavior in XP Projects 46
3.4 Setting Up a Team 50
3.5 Developing Team Skills 52
3.6 Training Together 54
3.7 Finding and Keeping a Client for a University-Based Project or a Small Business Start-Up 54
3.8 The Organizational Framework 56

3.9 Planning 60
   3.9.1 PERT (Program Evaluation and Review Technique) 61
   3.9.2 Gantt Charts 62

3.10 Dealing with Problems 65
   3.10.1 Basic Strategies 65
   3.10.2 When Things Go Really Wrong 66

3.11 Risk Analysis 68
3.12 Review 69

Exercises 69
Conundrum 70
References 70

4. Starting an XP Project

4.1 Project Beginnings 73
   4.1.1 Researching the Business Background 74
   4.1.2 Exploring the Outline System Description 76
4.2 The First Meetings with the Client 79
4.3 Business Analysis and Problem Discovery 80
4.4 The Initial Stages of Building a Requirements Document 82
4.5 Techniques for Requirements Elicitation 84
4.6 Putting Your Knowledge Together 85
4.7 Getting Technical 85
4.8 Developing the Requirements Documents 88
4.9 Specifying and Measuring the Quality Attributes of the System 91
   4.9.1 Identifying Attributes 92
   4.9.2 Specifying the Acceptable Level of an Attribute 94
   4.9.3 User Characteristics and User Interface Characteristics 95
4.10 The Formal Requirements Document and System Metaphor 96
   4.10.1 Commentary 106
4.11 Contract Negotiation 108
4.12 Case Study: The Impact of Organizational Politics 114
4.13 Review 116
Conundrum 116
References 117

5. Identifying Stories and Preparing to Build 119
5.1 Looking at the User Stories 119
5.2 Collections of Stories 128
   5.2.1 Pharmacovigilance 129
   5.2.2 Stamps System 131
   5.2.3 DELTAH (Developing European Leadership Through Action-Learning in Healthcare) 131
5.3 User Interfaces 139
5.4 Communicating Clearly with the Customer and Building Confidence 141
5.5 Demonstrating the Non-Functional Requirements 143
   5.5.1 Non-Functional Requirements 143
5.6 Estimating Resources 144
   5.6.1 Software Cost Estimation 145
   5.6.2 Object Point Analysis 146
   5.6.3 COSMIC FFP 147
5.7 Review 149
Exercises 149
Conundrum 150
References 151

6. Bringing the System Together as a Coherent Concept 153
6.1 What is the Problem? 153
6.2 A Simple Common Metaphor 156
6.3 Architectures and Patterns 159
7. Designing the System Tests 181

7.1 Preparing to Build Functional Test Sets 181
  7.1.1 Tests and Testing 181
  7.1.2 Testing from a Model 183
  7.1.3 Developing the Model 187
7.2 Testing with the Data in Mind 191
7.3 The Full Functional System Testing Strategy 192
7.4 The Thinking Behind the System Test Process 193
  7.4.1 An Algorithm for Determining the Transition Cover 198
7.5 Design for Test 201
  7.5.1 Design for Test Principle 1: Controllability 202
  7.5.2 Design for Test Principle 2: Observability 202
7.6 Test Documentation 203
7.7 Non-Functional Testing 205
  7.7.1 Reliability 206
  7.7.2 Usability 206
  7.7.3 Efficiency 207
  7.7.4 Portability 207
7.8 Testing Internet Applications and Web Sites 207
7.9 Review 209
Exercise 210
Conundrum 213
References 213

8. Units and Their Tests 215

8.1 Basic Considerations 215
8.2 Identifying the Units 216
8.3 Unit Testing 219
8.4 More Complex Units 222
  8.4.1 Case Example: The AddElement Function in JHotDraw 223
8.5 Automating Unit Tests 232
  8.5.1 Writing Unit Tests in JUnit 233
  8.5.2 Managing Tests 235
x Contents

Exercises 267
Conundrum 267
Reference 267

11. Reflecting on the Process 269

11.1 Skills and Lessons Learned 269
11.2 The XP Experience 270
11.3 Personal and Team Assessment 270
11.4 Review 271
Exercises 271
11.5 Conundrums: Discussion 271
11.6 A Final Word 277

12. Lifestyle Matters 281

12.1 Keeping Fit 282
    12.1.1 Correct Sitting Position 283
    12.1.2 Combating RSI 284
12.2 General Well-Being 285
12.3 Mental Preparation 285
12.4 Diet 286
    12.4.1 Diet and Brain Function 286
    12.4.2 Summary of Dietary Information 287
12.5 Music and Work 288
12.6 Review 289
References 290

Appendix 291

Bibliography 305

Index 309