
CONTENTS

| | |
|--------------------------|------|
| Foreword | xv |
| Preface | xvii |
| Acknowledgments | xxi |
| About the Author | xxv |
| Chapter I | |
| Introduction to Agile | I |
| History of Agile | 3 |
| Snowbird | 10 |
| After Snowbird | 13 |
| Agile Overview | 14 |
| The Iron Cross | 15 |
| Charts on the Wall | 15 |
| The First Thing You Know | 18 |
| The Meeting | 18 |
| The Analysis Phase | 19 |
| The Design Phase | 20 |
| The Implementation Phase | 21 |
| The Death March Phase | 22 |
| Hyperbole? | 23 |

| | |
|--|-----------|
| A Better Way | 23 |
| Iteration Zero | 24 |
| Agile Produces Data | 25 |
| Hope versus Management | 27 |
| Managing the Iron Cross | 27 |
| Business Value Order | 31 |
| Here Endeth the Overview | 31 |
| Circle of Life | 31 |
| Conclusion | 35 |
| Chapter 2 The Reasons for Agile | 37 |
| Professionalism | 38 |
| Software Is Everywhere | 39 |
| We Rule the World | 41 |
| The Disaster | 42 |
| Reasonable Expectations | 43 |
| We Will Not Ship Shyt! | 43 |
| Continuous Technical Readiness | 45 |
| Stable Productivity | 46 |
| Inexpensive Adaptability | 49 |
| Continuous Improvement | 50 |
| Fearless Competence | 50 |
| QA Should Find Nothing | 52 |
| Test Automation | 52 |
| We Cover for Each Other | 54 |
| Honest Estimates | 54 |
| You Need to Say “No” | 55 |
| Continuous Aggressive Learning | 55 |
| Mentoring | 56 |
| The Bill of Rights | 56 |
| Customer Bill of Rights | 56 |
| Developer Bill of Rights | 57 |
| Customers | 57 |
| Developers | 59 |
| Conclusion | 61 |

| | | |
|------------------|--|-----------|
| Chapter 3 | Business Practices | 63 |
| | Planning | 64 |
| | Trivariate Analysis | 65 |
| | Stories and Points | 66 |
| | ATM Stories | 67 |
| | Stories | 74 |
| | Story Estimation | 76 |
| | Managing the Iteration | 78 |
| | The Demo | 80 |
| | Velocity | 81 |
| | Small Releases | 82 |
| | A Brief History of Source Code Control | 83 |
| | Tapes | 85 |
| | Disks and SCCS | 85 |
| | Subversion | 86 |
| | Git and Tests | 87 |
| | Acceptance Tests | 88 |
| | Tools and Methodologies | 89 |
| | Behavior-Driven Development | 90 |
| | The Practice | 90 |
| | Whole Team | 93 |
| | Co-Location | 94 |
| | Conclusion | 96 |
| Chapter 4 | Team Practices | 97 |
| | Metaphor | 98 |
| | Domain-Driven Design | 99 |
| | Sustainable Pace | 100 |
| | Overtime | 102 |
| | Marathon | 103 |
| | Dedication | 103 |
| | Sleep | 104 |
| | Collective Ownership | 104 |
| | The X Files | 106 |
| | Continuous Integration | 107 |
| | Then Came Continuous Build | 108 |
| | The Continuous Build Discipline | 109 |

| | | |
|------------------|----------------------------|------------|
| | Standup Meetings | 110 |
| | Pigs and Chickens? | 111 |
| | Shout-out | 111 |
| | Conclusion | 111 |
| Chapter 5 | Technical Practices | 113 |
| | Test-Driven Development | 114 |
| | Double-Entry Bookkeeping | 114 |
| | The Three Rules of TDD | 116 |
| | Debugging | 117 |
| | Documentation | 117 |
| | Fun | 118 |
| | Completeness | 119 |
| | Design | 121 |
| | Courage | 121 |
| | Refactoring | 123 |
| | Red/Green/Refactor | 124 |
| | Bigger Refactorings | 125 |
| | Simple Design | 125 |
| | Design Weight | 127 |
| | Pair Programming | 127 |
| | What Is Pairing? | 128 |
| | Why Pair? | 129 |
| | Pairing as Code Review | 129 |
| | What about the Cost? | 130 |
| | Just Two? | 130 |
| | Management | 130 |
| | Conclusion | 131 |
| Chapter 6 | Becoming Agile | 133 |
| | Agile Values | 134 |
| | Courage | 134 |
| | Communication | 134 |
| | Feedback | 135 |
| | Simplicity | 135 |
| | The Menagerie | 136 |

| | |
|---|-----|
| Transformation | 137 |
| The Subterfuge | 138 |
| The Lion Cubs | 138 |
| Weeping | 139 |
| Moral | 139 |
| Faking It | 139 |
| Success in Smaller Organizations | 140 |
| Individual Success and Migration | 141 |
| Creating Agile Organizations | 141 |
| Coaching | 142 |
| Scrum Masters | 143 |
| Certification | 143 |
| Real Certification | 144 |
| Agile in the Large | 144 |
| Agile Tools | 148 |
| Software Tools | 148 |
| What Makes for an Effective Tool? | 149 |
| Physical Agile Tools | 151 |
| The Pressure to Automate | 152 |
| ALMs for the Not-Poor | 153 |
| Coaching—An Alternative View | 155 |
| The Many Paths to Agile | 155 |
| From Process Expert to Agile Expert | 156 |
| The Need for Agile Coaching | 157 |
| Putting the Coach into Agile Coach | 158 |
| Going Beyond the ICP-ACC | 158 |
| Coaching Tools | 159 |
| Professional Coaching Skills Are Not Enough | 159 |
| Coaching in a Multiteam Environment | 160 |
| Agile in the Large | 161 |
| Using Agile and Coaching to Become Agile | 161 |
| Growing Your Agile Adoption | 162 |
| Going Big by Focusing on the Small | 164 |
| The Future of Agile Coaching | 165 |
| Conclusion (Bob Again) | 165 |

| | | |
|------------------|---|------------|
| Chapter 7 | Craftsmanship | 167 |
| | The Agile Hangover | 169 |
| | Expectation Mismatch | 170 |
| | Moving Apart | 172 |
| | Software Craftsmanship | 173 |
| | Ideology versus Methodology | 174 |
| | Does Software Craftsmanship Have Practices? | 175 |
| | Focus on the Value, Not the Practice | 176 |
| | Discussing Practices | 177 |
| | Craftsmanship Impact on Individuals | 178 |
| | Craftsmanship Impact on Our Industry | 179 |
| | Craftsmanship Impact on Companies | 180 |
| | Craftsmanship and Agile | 181 |
| | Conclusion | 182 |
| Chapter 8 | Conclusion | 183 |
| Afterword | | 185 |
| Index | | 191 |