



Robert C. Martin Series

The Software Craftsman

Professionalism, Pragmatism, Pride

Foreword by Robert C. Martin

Sandro Mancuso

The Software Craftsman

Software Craftsman, The: Professionalism, Pragmatism, Pride

Table of Contents

Cover

Half Title

Title Page

Copyright Page

Contents

Foreword by Robert C. Martin

Preface

Acknowledgments

About the Author

Part I: Ideology and Attitude

Chapter 1 Software Development in the Twenty-First Century

Seniority

A New Reality

Chapter 2 Agile

Process-Oriented Agile Disciplines

Technical-Oriented Disciplines

What Is It to Be Agile?

Agile Manifesto

The Agile Transformation Era

The Agile Hangover

Agile versus Software Craftsmanship

Summary

Table of Contents

Chapter 3 Software Craftsmanship

- A Better Metaphor
- What Does Wikipedia Say?
- A More Personal Definition
- A Shorter Definition
- Beyond Definitions
- Craft, Trade, Engineering, Science, or Art
- Software Craftsmanship History
- Summary

Chapter 4 The Software Craftsmanship Attitude

- Who Owns Your Career?
- Keeping Ourselves Up to Date
- Know Who to Follow
- Practice, Practice, Practice
- Socialize
- Deliberate Discovery
- Work-Life Balance
- Summary

Chapter 5 Heroes, Goodwill, and Professionalism

- Learning How to Say No
- Providing Options
- Enlightened Managers
- Summary

Chapter 6 Working Software

- Working Software Is Not Enough
- Looking After Our Garden
- The Invisible Threat
- The Wrong Notion of Time
- Legacy Code

Table of Contents

Summary

Chapter 7 Technical Practices

The Right Thing versus the Thing Right

Context

Extreme Programming History

Practices and Values

Accountability

Pragmatism

Summary

Chapter 8 The Long Road

A Tale from a Brazilian Teenager

Focus and Determination

Job as Investment

Autonomy, Mastery, and Purpose

Career Inside Companies

Summary

Part II: A Full Transformation

Chapter 9 Recruitment

An Ordinary Job Description

Too Busy to Interview

No Job Descriptions

Recommendations

Community Involvement

Defining Effective Filtering Criteria

Proactive Recruitment

Summary

Chapter 10 Interviewing Software Craftsmen

A Business Negotiation

Identifying Productive Partnerships

Table of Contents

Good Interviews

Taking a Punt

Hiring for an Existing Team versus Hiring for a New Team

Pre-Interview Coding Exercises

Everyone Should Know How to Interview

Developers Must Interview Developers

Summary

Chapter 11 Interview Anti-Patterns

Dont Be a Smart-Ass Interviewer

Dont Use Brainteasers

Dont Ask Questions to Which You Dont Know the Answers

Dont Try to Make the Candidate Look Like a Fool

Dont Block the Internet

Dont Code on a Piece of Paper

Dont Use Algorithms

Dont Conduct Phone Interviews

Summary

Chapter 12 The Cost of Low Morale

The Agile Hangover: Low Morale

The Cost of Employing 9-to-5 Developers

Constrained by Lack of Motivation

Injecting Passion

Summary

Chapter 13 Culture of Learning

Wrong Motivation

Creating a Culture of Learning

What if Others Dont Want to Join In?

Summary

Chapter 14 Driving Technical Changes

Table of Contents

Identifying Skepticism Patterns

Be Prepared

Where Do We Start?

Fear and Incompetence

How Do I Convince My Manager?

How Do I Convince My Team to Do TDD?

Facing the Skeptics

Should We Really Care about All That?

Summary

Chapter 15 Pragmatic Craftsmanship

Quality Is Always Expected

Busting the Expensive and Time-Consuming Quality Myth

Refactoring

The One Way of Developing Software

Helping the Business

Software Projects Are Not about Us

Great versus Mediocre

Four Rules of Simple Design

Craftsmanship and Pragmatism

Summary

Chapter 16 A Career as a Software Craftsman

Being a Craftsman

Career Progression

Roads and Milestones

Job Diversity

The Mission

Appendix: Craftsmanship Myths and Further Explanations

Index

Table of Contents