Lean Software Development

An Agile Toolkit

The Agile Software Development Series

Cockburn • Highsmith
Series Editors

- Adapting agile practices to your development organization
- Uncovering and eradicating waste throughout the software development lifecycle
- Practical techniques for every development manager, project manager, and technical leader

Forewords by
Jim Highsmith
and Ken Schwaber

Mary Poppendieck
Tom Poppendieck

Lean Software Development

Lean Software Development: An Agile Toolkit: An Agile Toolkit

Table of Contents

Contents
Foreword by Jim Highsmith
Foreword by Ken Schwaber
Preface
Introduction
Chapter 1 Eliminate Waste
- The Origins of Lean Thinking
- Tool 1: Seeing Waste .
  - Partially Done Work
  - Extra Processes
  - Extra Features
  - Task Switching
  - Waiting
  - Motion
  - Defects
  - Management Activities
- Tool 2: Value Stream Mapping
  - Map Your Value Stream
  - An Agile Value Stream Map
- Try This

Chapter 2 Amplify Learning
- The Nature of Software Development
  - Perspectives on Quality

Table of Contents

- Tool 3: Feedback
  - Software Development Feedback Loops
- Tool 4: Iterations
  - Iteration Planning
  - Team Commitment
  - Convergence
  - Negotiable Scope
- Tool 5: Synchronization
  - Synch and Stabilize
  - Spanning Application
  - Matrix
- Tool 6: Set-Based Development
  - Set-Based Versus Point-Based
  - Set-Based Software Development
- Try This

Chapter 3 Decide as Late as Possible

- Concurrent Development
  - Concurrent Software Development
  - Cost Escalation
- Tool 7: Options Thinking
  - Delaying Decisions
  - Options
  - Microsoft Strategy, circa 1988
  - Options Thinking in Software Development
- Tool 8: The Last Responsible Moment
- Tool 9: Making Decisions
  - Depth-First Versus Breadth-First Problem Solving
  - Intuitive Decision Making
  - The Marines

Table of Contents

Simple Rules

Simple Rules for Software Development

Try This

Chapter 4 Deliver as Fast as Possible

Why Deliver Fast?

Tool 10: Pull Systems

Manufacturing Schedules

Software Development Schedules

Software Pull Systems

Information Radiators

Tool 11: Queuing Theory

Reducing Cycle Time

How Queues Work

Tool 12: Cost of Delay

Product Model

Application Model

Tradeoff Decisions

Try This

Chapter 5 Empower the Team

Beyond Scientific Management

CMM

CMMI

Tool 13: Self-Determination

The NUMMI Mystery

A Management Improvement Process

Tool 14: Motivation

Magic at 3M

Purpose

The Building Blocks of Motivation

Table of Contents

Long Days and Late Nights

Tool 15: Leadership

Leadership

Respected Leaders

Master Developers

The Fuzzy Front End

Where Do Master Developers Come From?

Project Management

Tool 16: Expertise

Nucor

Xerox

Communities of Expertise

Standards

Try This

Chapter 6 Build Integrity In

Integrity

Perceived Integrity

Conceptual Integrity

The Key to Integrity

Tool 17: Perceived Integrity

Model-Driven Design

Maintaining Perceived Integrity

Tool 18: Conceptual Integrity

Software Architecture Basics

Emerging Integrity

Tool 19: Refactoring

Keeping Architecture Healthy

Maintaining Conceptual Integrity

Isnt Refactoring Rework?

Table of Contents

- Tool 20: Testing
  - Communication
  - Feedback
  - Scaffolding
  - As-Built
  - Maintenance
- Try This

Chapter 7 See the Whole

- Systems Thinking
- Tool 21: Measurements
  - Local Optimization
  - Why Do We Suboptimize?
  - Measuring Performance
  - Information Measurements
- Tool 22: Contracts
  - Can There Be Trust Between Firms?
  - But Software Is Different
  - The Purpose of Contracts
  - Fixed-Price Contracts
  - Time-and-Materials Contracts
  - Multistage Contracts
  - Target-Cost Contracts
  - Target-Schedule Contracts
  - Shared-Benefit Contracts
  - The Key: Optional Scope
- Try This

Chapter 8 Instructions and Warranty

- CautionUse Only as Directed.
- Instructions

Table of Contents

Sphere of Influence

Large Company

Small Company

Special Work Environments

Troubleshooting Guide

Warranty

Bibliography

Index