

Extracted from:

The Agile Samurai

How Agile Masters Deliver Great Software

This PDF file contains pages extracted from The Agile Samurai, published by the Pragmatic Bookshelf. For more information or to purchase a paperback or PDF copy, please visit <http://www.pragprog.com>.

Note: This extract contains some colored text (particularly in code listing). This is available only in online versions of the books. The printed versions are black and white. Pagination might vary between the online and printer versions; the content is otherwise identical.

Copyright © 2010 The Pragmatic Programmers, LLC.

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

The
Pragmatic
Programmers

The Agile Samurai

How Agile Masters
Deliver
Great Software



Jonathan Rasmusson

Edited by Susannah Davidson Pfalzer

Contents

Acknowledgments	13
It's Good to See You	14
How to Read This Book	15
Fun Bits with Purpose	15
Online Resources	16
I Introducing Agile	17
1 Agile in a Nutshell	18
1.1 Deliver Something of Value Every Week	19
1.2 How Does Agile Planning Work?	22
1.3 Done Means Done	23
1.4 Three Simple Truths	25
2 Meet Your Agile Team	28
2.1 How Are Agile Projects Different?	29
2.2 What Makes an Agile Team Tick	31
2.3 Roles We Typically See	36
2.4 Tips for Forming Your Agile Team	46
II Agile Project Inception	49
3 How to Get Everyone on the Bus	50
3.1 What Kills Most Projects	51
3.2 Ask the Tough Questions	51
3.3 Enter the Inception Deck	53
3.4 How It Works	53
3.5 The Inception Deck in a Nutshell	54

4	Seeing the Big Picture	56
4.1	Ask: Why Are We Here?	57
4.2	Create an Elevator Pitch	59
4.3	Design a Product Box	63
4.4	Create a NOT List	66
4.5	Meet Your Neighbors	67
5	Making It Real	74
5.1	Show Your Solution	75
5.2	Ask What Keeps Us Up at Night	76
5.3	Size It Up	80
5.4	Be Clear on What's Going to Give	83
5.5	Show What It's Going to Take	89
III	Agile Project Planning	95
6	Gathering User Stories	96
6.1	The Problem with Documentation	96
6.2	Enter the User Story	100
6.3	Elements of Good User Stories	101
6.4	How to Host a Story-Gathering Workshop	110
7	Estimation: The Fine Art of Guessing	116
7.1	The Problem with High-Level Estimates	116
7.2	Turning Lemons into Lemonade	118
7.3	How Does It Work?	124
8	Agile Planning: Dealing with Reality	132
8.1	The Problems with Static Plans	133
8.2	Enter the Agile Plan	135
8.3	Be Flexible About Scope	139
8.4	Your First Plan	141
8.5	The Burn-Down Chart	150
8.6	Transitioning a Project to Agile	153
8.7	Putting It into Practice	154
IV	Agile Project Execution	162

9	Iteration Management: Making It Happen	163
9.1	How to Deliver Something of Value Every Week	164
9.2	The Agile Iteration	165
9.3	Help Wanted	166
9.4	Step 1: Analysis and Design: Making the Work Ready .	167
9.5	Step 2: Development: Do the Work	173
9.6	Step 3: Test: Check the Work	174
9.7	Kanban	176
10	Creating an Agile Communication Plan	181
10.1	Four Things to Do During Any Iteration	182
10.2	The Story-Planning Meeting	182
10.3	The Showcase	184
10.4	Plan the Next Iteration	184
10.5	How to Host a Mini-Retrospective	186
10.6	How Not to Host a Daily Stand-Up	188
10.7	Do Whatever Works for You	189
11	Setting Up a Visual Workspace	193
11.1	Uh-oh...Here Come the Heavies!	193
11.2	How to Create a Visual Workspace	197
11.3	Show Your Intent	199
11.4	Create and Share a Common Domain Language	200
11.5	Watch Those Bugs	201
V	Creating Agile Software	204
12	Unit Testing: Knowing It Works	205
12.1	Welcome to Vegas, Baby!	206
12.2	Enter the Unit Test	208
13	Refactoring: Paying Down Your Technical Debt	215
13.1	Turn on a Dime	216
13.2	Technical Debt	217
13.3	Make Payments Through Refactoring	219
14	Test-Driven Development	228
14.1	Write Your Tests First	229
14.2	Use the Tests to Deal with Complexity	233

15 Continuous Integration: Making It Production-Ready	238
15.1 Showtime	238
15.2 A Culture of Production Readiness	241
15.3 What Is Continuous Integration?	242
15.4 How Does It Work?	243
15.5 Establish a Check-in Process	244
15.6 Create an Automated Build	246
15.7 Work in Small Chunks	248
15.8 Where Do I Go from Here?	250
VI Appendixes	253
A Agile Principles	254
A.1 The Agile Manifesto	254
A.2 Twelve Agile Principles	254
B Resources	256
C Bibliography	257
Index	259

The Pragmatic Bookshelf

The Pragmatic Bookshelf features books written by developers for developers. The titles continue the well-known Pragmatic Programmer style and continue to garner awards and rave reviews. As development gets more and more difficult, the Pragmatic Programmers will be there with more titles and products to help you stay on top of your game.

Visit Us Online

The Agile Samurai

<http://pragprog.com/titles/jtrap>

Source code from this book, errata, and other resources. Come give us feedback, too!

Register for Updates

<http://pragprog.com/updates>

Be notified when updates and new books become available.

Join the Community

<http://pragprog.com/community>

Read our weblogs, join our online discussions, participate in our mailing list, interact with our wiki, and benefit from the experience of other Pragmatic Programmers.

New and Noteworthy

<http://pragprog.com/news>

Check out the latest pragmatic developments, new titles and other offerings.

Buy the Book

If you liked this eBook, perhaps you'd like to have a paper copy of the book. It's available for purchase at our store: pragprog.com/titles/jtrap.

Contact Us

Online Orders:	www.pragprog.com/catalog
Customer Service:	support@pragprog.com
Non-English Versions:	translations@pragprog.com
Pragmatic Teaching:	academic@pragprog.com
Author Proposals:	proposals@pragprog.com
Contact us:	1-800-699-PROG (+1 919 847 3884)