Beyond Requirements
A
gile software development centers on four values, which are identified in the Agile Alliance’s Manifesto*:  
1. Individuals and interactions over processes and tools  
2. Working software over comprehensive documentation  
3. Customer collaboration over contract negotiation  
4. Responding to change over following a plan  

The development of Agile software requires innovation and responsiveness, based on generating and sharing knowledge within a development team and with the customer. Agile software developers draw on the strengths of customers, users, and developers to find just enough process to balance quality and agility. 

The books in The Agile Software Development Series focus on sharing the experiences of such Agile developers. Individual books address individual techniques (such as Use Cases), group techniques (such as collaborative decision making), and proven solutions to different problems from a variety of organizational cultures. The result is a core of Agile best practices that will enrich your experiences and improve your work.

* © 2001, Authors of the Agile Manifesto
Beyond Requirements

Analysis with an Agile Mindset

Kent J. McDonald

Illustrations by Jeff Rains
Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and the publisher was aware of a trademark claim, the designations have been printed with initial capital letters or in all capitals.

The author and publisher have taken care in the preparation of this book, but make no expressed or implied warranty of any kind and assume no responsibility for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information or programs contained herein.

For information about buying this title in bulk quantities, or for special sales opportunities (which may include electronic versions; custom cover designs; and content particular to your business, training goals, marketing focus, or branding interests), please contact our corporate sales department at corpsales@pearsoned.com or (800) 382-3419.

For government sales inquiries, please contact governmentsales@pearsoned.com.

For questions about sales outside the U.S., please contact international@pearsoned.com.

Visit us on the Web: informit.com/aw

Library of Congress Cataloging-in-Publication Data
McDonald, Kent J.
   Beyond requirements : analysis with an agile mindset / Kent J. McDonald ; illustrations by Jeff Rains.
   pages cm
   Includes bibliographical references and index.
1. Decision making. 2. Requirements engineering. 3. Business requirements analysis. I. Title.
T57.95.M384 2016
658.4'0354—dc23
2015022866

Copyright © 2016 Pearson Education, Inc.

Illustrations by Jeff Rains

All rights reserved. Printed in the United States of America. This publication is protected by copyright, and permission must be obtained from the publisher prior to any prohibited reproduction, storage in a retrieval system, or transmission in any form or by any means, electronic, mechanical, photocopying, recording, or likewise. To obtain permission to use material from this work, please submit a written request to Pearson Education, Inc., Permissions Department, 200 Old Tappan Road, Old Tappan, New Jersey 07675, or you may fax your request to (201) 236-3290.

ISBN-10: 0-321-83455-0
Text printed in the United States on recycled paper at RR Donnelley in Crawfordsville, Indiana.
First printing, September 2015
To all of those who asked, “Is the book done yet?”
Yes, yes it is.
This page intentionally left blank
# Contents

Preface ................................................................. xv
Acknowledgments ................................................... xxv
About the Author .................................................. xxvii

## Part I: Ideas ....................................................... 1

### Chapter 1: Guiding Principles ................................. 3
- Introduction ..................................................... 3
- Deliver Value .................................................. 4
- Collaborate ..................................................... 5
- Iterate .......................................................... 7
- Simplify ......................................................... 8
- Consider Context ............................................. 9
- Decide Wisely .................................................. 10
- Reflect and Adapt ............................................. 11
- Conclusion ..................................................... 12
- If You Remember Nothing Else ......................... 12

### Chapter 2: Helpful Concepts ................................. 15
- Introduction ..................................................... 15
- Needs and Solutions ......................................... 15
- Outcome and Output ........................................ 19
- Discovery and Delivery ..................................... 20
- If You Remember Nothing Else ......................... 23

### Chapter 3: Influence of Lean Startup ..................... 25
- Introduction ..................................................... 25
- Customer Development ..................................... 25
- Build-Measure-Learn ........................................ 29
- Metrics .......................................................... 31
  - Good Metrics ............................................... 32
### CONTENTS

- Things to Consider with Metrics ........................................... 34
- Creating Your Metrics ...................................................... 36
- If You Remember Nothing Else ........................................... 38

#### Chapter 4: Decision Making ............................................ 39

- Introduction ................................................................. 39
- A Structure for Decision Making ......................................... 39
  - Determine the Decision Maker ........................................ 39
  - Select a Decision Mechanism ......................................... 41
  - Determine What Information Is Needed ............................ 42
  - Make a Timely Decision ................................................ 43
  - Build Support with Peers/Stakeholders ............................ 45
  - Communicate the Decision ............................................. 45
  - Enact the Decision ..................................................... 46
- Real Options ................................................................. 46
- Cognitive Biases ............................................................. 48
- Elicitation ................................................................. 49
- Analysis ................................................................. 51
- Decision Making .......................................................... 52
- If You Remember Nothing Else ........................................... 53

#### Chapter 5: Deliver Value .................................................. 55

- Introduction ................................................................. 55
- Feature Injection ........................................................... 55
  - Identify the Value ....................................................... 56
  - Inject the Features ..................................................... 59
  - Spot the Examples ...................................................... 61
- Minimum Viable Product .................................................. 63
- Minimum Marketable Features .......................................... 65
- If You Remember Nothing Else ........................................... 67

#### Chapter 6: Analysis with an Agile Mindset ......................... 69

- Introduction ................................................................. 69
- What Is the Need? ......................................................... 71
- What Are Some Possible Solutions? ................................. 71
- What Should We Do Next? .............................................. 72
- What Are the Details of This Part (i.e., Telling the Story)? .... 73
- If You Remember Nothing Else ........................................... 73
## Part II: Case Studies ........................................ 75

**Chapter 7: Case Study: Conference Submission System** ....................... 77  
  Introduction ......................................................... 77  
  The Need ............................................................... 77  
  The Possible Solution(s) ............................................. 78  
  The Deliveries of Value ............................................... 79  
  Define-Build-Test ................................................... 81  
  The Incident of the Themes .......................................... 84  
  Agile2014 ............................................................... 90  
  Lessons Learned .................................................... 92

**Chapter 8: Case Study: Commission System** ................................. 95  
  Introduction ......................................................... 95  
  The Need ............................................................... 96  
  The Possible Solution(s) ............................................. 96  
  The Deliveries of Value ............................................... 97  
  Lessons Learned .................................................... 98

**Chapter 9: Case Study: Data Warehouse** .................................. 101  
  Introduction ......................................................... 101  
  The Need ............................................................... 101  
  The Possible Solution(s) ............................................. 102  
  The Deliveries of Value ............................................... 103  
  Lessons Learned .................................................... 110

**Chapter 10: Case Study: Student Information System** .............. 111  
  Introduction ......................................................... 111  
  The Need ............................................................... 111  
  The Possible Solution(s) ............................................. 114  
  Lessons Learned .................................................... 118

## Part III: Techniques ................................................. 121

**Chapter 11: Understanding Stakeholders** .............................. 123  
  Introduction ......................................................... 123  
  Stakeholder Analysis ............................................... 123  
  User Analysis ......................................................... 124  
  Stakeholder Map ..................................................... 124  
  What It Is ............................................................. 124
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>An Example</td>
<td>125</td>
</tr>
<tr>
<td>When to Use It</td>
<td>126</td>
</tr>
<tr>
<td>Why Use It</td>
<td>126</td>
</tr>
<tr>
<td>How to Use It</td>
<td>126</td>
</tr>
<tr>
<td>Caveats and Considerations</td>
<td>129</td>
</tr>
<tr>
<td>Additional Resources</td>
<td>129</td>
</tr>
<tr>
<td>Commitment Scale</td>
<td>129</td>
</tr>
<tr>
<td>What It Is</td>
<td>129</td>
</tr>
<tr>
<td>An Example</td>
<td>129</td>
</tr>
<tr>
<td>When to Use It</td>
<td>130</td>
</tr>
<tr>
<td>Why Use It</td>
<td>130</td>
</tr>
<tr>
<td>How to Use It</td>
<td>131</td>
</tr>
<tr>
<td>Caveats and Considerations</td>
<td>132</td>
</tr>
<tr>
<td>Additional Resource</td>
<td>132</td>
</tr>
<tr>
<td>User Modeling</td>
<td>133</td>
</tr>
<tr>
<td>What It Is</td>
<td>133</td>
</tr>
<tr>
<td>An Example</td>
<td>133</td>
</tr>
<tr>
<td>When to Use It</td>
<td>135</td>
</tr>
<tr>
<td>Why Use It</td>
<td>135</td>
</tr>
<tr>
<td>How to Use It</td>
<td>136</td>
</tr>
<tr>
<td>Caveats and Considerations</td>
<td>137</td>
</tr>
<tr>
<td>Additional Resources</td>
<td>137</td>
</tr>
<tr>
<td>Persona</td>
<td>138</td>
</tr>
<tr>
<td>What It Is</td>
<td>138</td>
</tr>
<tr>
<td>An Example</td>
<td>138</td>
</tr>
<tr>
<td>When to Use It</td>
<td>139</td>
</tr>
<tr>
<td>Why Use It</td>
<td>139</td>
</tr>
<tr>
<td>How to Use It</td>
<td>139</td>
</tr>
<tr>
<td>Caveats and Considerations</td>
<td>140</td>
</tr>
<tr>
<td>Additional Resources</td>
<td>140</td>
</tr>
<tr>
<td>Chapter 12: Understanding Context</td>
<td>141</td>
</tr>
<tr>
<td>Introduction</td>
<td>141</td>
</tr>
<tr>
<td>Purpose-Based Alignment Model</td>
<td>142</td>
</tr>
<tr>
<td>What It Is</td>
<td>142</td>
</tr>
<tr>
<td>The Quadrants Explained</td>
<td>143</td>
</tr>
<tr>
<td>An Example</td>
<td>144</td>
</tr>
<tr>
<td>When to Use It</td>
<td>145</td>
</tr>
<tr>
<td>Why Use It</td>
<td>145</td>
</tr>
</tbody>
</table>
Contents

How to Use It ............................................. 145
Caveats and Considerations ................................ 146
Additional Resource .................................. 147
Six Questions ........................................... 147
What It Is ................................................ 147
An Example ............................................ 148
When to Use It ......................................... 148
Why Use It ............................................. 148
How to Use It .......................................... 149
Caveats and Considerations .............................. 149
Additional Resource .................................. 150
Context Leadership Model ............................... 150
What It Is ................................................ 150
An Example ............................................ 154
When to Use It ......................................... 154
Why Use It ............................................. 155
How to Use It .......................................... 155
Caveats and Considerations .............................. 156
Additional Resource .................................. 157

Chapter 13: Understanding the Need ...................... 159
Introduction ............................................. 159
Decision Filters .......................................... 160
What It Is ................................................ 160
An Example ............................................ 160
When to Use It ......................................... 161
Why Use It ............................................. 161
How to Use It .......................................... 161
Caveats and Considerations .............................. 163
Additional Resources .................................. 163
Project Opportunity Assessment ......................... 163
What It Is ................................................ 163
An Example ............................................ 164
When to Use It ......................................... 165
Why Use It ............................................. 166
How to Use It .......................................... 166
Caveats and Considerations .............................. 166
Additional Resource .................................. 167
CONTENTS

Problem Statement ............................................... 167
What It Is ......................................................... 167
An Example ......................................................... 167
When to Use It .................................................... 168
Why Use It ........................................................ 168
How to Use It ...................................................... 168
Caveats and Considerations ................................. 168
Additional Resource ........................................... 169

Chapter 14: Understanding the Solution(s) ............... 171
Introduction ....................................................... 171
Impact Mapping ................................................... 173
What It Is ........................................................ 173
An Example ......................................................... 173
When to Use It ..................................................... 174
Why Use It ........................................................ 176
How to Use It ...................................................... 176
Caveats and Considerations ................................. 177
Additional Resources .......................................... 177
Story Mapping ..................................................... 177
What It Is ........................................................ 177
An Example ......................................................... 178
When to Use It ..................................................... 178
Why Use It ........................................................ 178
How to Use It ...................................................... 180
Caveats and Considerations ................................. 182
Additional Resources .......................................... 182
Collaborative Modeling ......................................... 182
What It Is ........................................................ 182
An Example ......................................................... 183
When to Use It ..................................................... 184
Why Use It ........................................................ 186
How to Use It ...................................................... 186
Caveats and Considerations ................................. 188
Additional Resources .......................................... 188
Acceptance Criteria ............................................. 188
What It Is ........................................................ 188
An Example ......................................................... 189
When to Use It ..................................................... 190
Definition of Done .............................................. 211
What It Is ......................................................... 211
An Example ...................................................... 211
When to Use It ................................................... 211
Why Use It ....................................................... 211
How to Use It .................................................... 212
Caveats and Considerations ................................. 212
Additional Resources ........................................... 213
System Documentation ........................................ 213
What It Is ......................................................... 213
An Example ...................................................... 214
When to Use It ................................................... 214
Why Use It ....................................................... 214
How to Use It .................................................... 215
Caveats and Considerations ................................. 215
Additional Resources ........................................... 217

Part IV: Resources ................................................. 219
Glossary ............................................................. 221
References .......................................................... 245
Index ................................................................. 249
Preface

What This Book Is About

I wrote Beyond Requirements to paint a picture of analysis in IT projects and how it can be applied with an agile mindset to make those projects more effective. For the purposes of this book I think of analysis as the activities involved with

- Understanding stakeholders
- Understanding context
- Understanding the need
- Understanding the solution(s)
- Organizing and persisting solution information

Performing these activities with an agile mindset, which I explain in Chapter 1, best positions teams to satisfy stakeholder needs. As a result, I assume that people are approaching work with an agile mindset (which is up to each individual to adopt) and that they are using agile techniques. Most of the techniques I describe can also be used in other environments, of course, but they’re most effective when combined with agile approaches.

Who Is This Book For?

If you find yourself performing analysis on a project in order to make sure the project is delivering the right thing, this book is for you. You may identify yourself as a business analyst (or derivation of that title), product owner, product manager, project manager, tester, or developer.

I chose to target those performing analysis activities or possessing analysis skills rather than analysts as a role, or even analysts as a profession. While it is true that the people who are most endowed with the analysis skill set are those who generally fill an analyst role, I didn’t want the advice in this book to get hung up on discussions such as, “The analyst does this, the developer does that, the tester does this other thing.” I’d much rather focus on describing why and when techniques are most appropriate and leave it up to you and your team to
determine who is the best person to do various activities. In many cases, multiple people on your team will end up doing analysis in order to take advantage of strong technical and business knowledge.

The business analyst role exists primarily because in the past several organizations used a prescriptive, phase-based approach to software development. In this approach, there was a time period in the project when the main work was eliciting and documenting requirements. Since it made sense to structure the software development organization according to how project work was done, all the people doing work in the analysis phase were lumped together and called business analysts. But gathering and documenting requirements didn’t generate much respect for the people doing it. Members of the analysis community longingly eyed the success project managers had enjoyed by proclaiming project management a profession, and they chose to do the same.

A lot of good things have come from the “professionalization” of business analysis, including more consideration of, training on, and attention to analysis skills. However, the benefits are somewhat diluted by the effort required to justify a separate profession for people who elicit, document, and manage requirements, and the overspecialization that may result. That effort would be better spent figuring out how analysis can be used to make projects more successful.

That doesn’t change the fact that you have a business analyst title and you have spent a considerable amount of your career honing your business analysis skills. Where does that leave you? Looking at analysis as an activity more than a role, title, or profession means that you can use your in-depth knowledge of analysis techniques to help your teams solve the right problems in the right way and help out with other activities on the project whenever possible.

To What Context Does This Book Apply?

This book focuses on the analysis that occurs on IT projects. An IT project is any project that results in solutions, often involving software, that support internal business processes, automate manual processes, or streamline current processes. Examples include building a system to support the session submission process for a conference, implementing a system to calculate and deliver commissions, reporting and data warehousing solutions, or implementing a solution to track student information at a nonprofit school.

I chose this focus for a few reasons. First, activities labeled as business analysis and the role of business analyst seem to be more prevalent in IT projects than in activities focused on product development. Second, most of the existing literature in the analysis space seems to assume a product development context,
and the context of the IT departments of an organization strikes me as underserved. Third, and probably most important, it's where most of my experience lies, so focusing on that topic gives me the opportunity to write from actual experience.

As I describe how analysis with an agile mindset works on IT projects, I won't delve too much into how to do tried-and-true analysis techniques. There are already enough resources that do a fantastic job of explaining those techniques, and it dilutes the focus of this book. Instead, I'll focus on why those techniques are helpful and when they are best used. I do introduce a few techniques from other skill sets not commonly known in analysis circles, and in those cases I provide a more detailed description of how to perform that technique. In all cases, I provide my favorite references for more information about those techniques.

The word project has acquired a certain stigma in the agile community. Those who apply that stigma feel as though the use of the word project implies some of the downsides of the way that projects are managed in a waterfall setting.

The term project often suggests the following:

- The temporary nature of projects is applied to the teams that work on them. People are brought to the work instead of the work being brought to the team.
- It takes a while to get an effort going due to the extensive chartering and planning that come with trying to predict the future 6 to 12 months out.
- Even though projects are intended to be temporary (or maybe because of that), they are rarely stopped once they get started. Sponsors and teams get enamored with projects and become more reluctant to end a project the longer it goes on.
- The project funding approach may encourage grouping multiple small changes together in order to justify expenditure, increasing the time before the changes are delivered to waiting stakeholders.

While these problems certainly exist, merely using the word project does not ensure that they will happen. I reasoned that most people are familiar with the idea of the project, and it would be more useful to explain that these patterns are antipatterns and it's possible for projects to work differently than to use a new term for an existing concept and deal with all of the confusion that could cause. As Deanna, one of my editors, suggested, I should just "own it" when it comes to using the word project.
What Problem Is This Book Trying to Solve?

Analysis is often portrayed as eliciting and documenting requirements, frequently in terms that sound a lot like asking people what they want and writing it down. Deep philosophical discussions about analysis often center on the best way to capture requirements: “Should I use a use case, or should I use a user story?” Requirements are important, but they are a means to an end, not the end in and of themselves. As I described previously, analysis is about understanding your stakeholders and their needs, identifying the best solution for satisfying those needs in your particular context, and then building a shared understanding of that solution. Requirements play a part in that work, especially around describing the need, but they are certainly not the end product.

One fundamental problem this book is trying to solve is how to determine whether your IT project is doing the right thing and how analysis can help you do that. It’s about changing the purpose of analysis from requirements gathering and capture to problem solving and building shared understanding. Along with that comes a substantial change in how your team views requirements and designs. They are no longer deliverables that get tossed over the wall to the people performing the next step in the process. Now both requirements and designs are tools that teams can use to build a shared understanding of the solution they seek to deliver in order to reach a desired outcome.

A second fundamental problem this book attempts to solve is to demonstrate how to do analysis in an agile setting. As many teams first adopt agile approaches, they struggle with finding the right balance between identifying a viable solution and describing that solution in too much detail too early. This book aims to show you how to perform analysis in an iterative fashion so that you can take advantage of the learning that occurs during development, testing, and deployment. While doing so, it also demonstrates that many analysis techniques are applicable in an agile setting with changes to when and to what extent you perform those techniques. I sought to solve this problem because many teams that adopt agile think analysis is no longer necessary, and as a result they end up creating solutions that don’t solve the identified problem, or don’t solve any problem at all.

How the Book Is Organized

This book is organized into three main parts to make it a bit easier to consume. The first part, “Ideas,” covers the agile mindset and some key principles that underlie the agile mindset and effective analysis. The second part, “Case Studies,” features four case studies that show how to practically apply the ideas in a variety of situations. The third part, “Techniques,” takes a deeper view of some techniques that are very helpful for using analysis in an agile setting.
Part I: Ideas

The first section takes a look at some key ideas that I consider essential for effectively performing analysis in an agile setting. These include the concepts that describe an agile mindset, and some helpful concepts from outside traditional analysis thinking that supplement typical analysis techniques. Finally, I build on those ideas to place analysis techniques in context.

Chapter 1: Guiding Principles

As I help teams adopt agile and tighten up their analysis approach, I find that adopting the appropriate mindset is more important than mastering a specific set of techniques. With the proper mindset and a great deal of self-discipline a team can be successful with minimal process. Without the proper mindset, teams find that they must continuously add process to aid the collaboration that comes naturally to those who have the right mindset.

What is the proper mindset? There are a variety of perspectives on that. The original definition of the agile mindset is encapsulated by the “Manifesto for Agile Software Development” and the corresponding principles. Others have expanded on those original ideas to describe the agile mindset, and I have done the same, placing emphasis on aspects that encourage building the right thing. I describe my perspective on the agile mindset through seven guiding principles:

- Deliver **value**
- Collaborate
- Iterate
- Simplify
- Consider context
- Decide wisely
- Reflect and adapt

Chapter 2: Helpful Concepts

I use this chapter to introduce some ideas that form the conceptual basis for the following chapters. The ideas discussed include

- Needs and solutions
- Outcome and **output**
- Discovery and **delivery**
Chapter 3: Influence of Lean Startup
This chapter explores some concepts of Lean Startup and describes how these concepts can be applied effectively to the context of IT projects. Those concepts include

- Customer development
- Build-Measure-Learn
- Metrics

Chapter 4: Decision Making
This chapter discusses decision making in more detail, specifically a structure for decision making, the idea of Real Options, and the cognitive biases I find can get in the way of effective decision making.

Chapter 5: Deliver Value
In this chapter I discuss some key concepts surrounding value delivery, including Feature Injection, minimum viable product, and minimum marketable feature.

Chapter 6: Analysis with an Agile Mindset
While I’m not necessarily advocating a new “analysis process,” I wanted to provide a general description of how analysis flows alongside the lifecycle of a project. This chapter positions the techniques from Chapters 11 through 15 in their usual location in the project lifecycle.

I don’t spend a great deal of time talking about this flow specifically because it is not the same on every project, but going through the whole flow once helps put the techniques into the proper perspective and helps to explain why certain techniques make more sense in some contexts than in others.

Part II: Case Studies
In this part of the book, I share four stories intended to describe analysis in a real-world setting. These stories illustrate the ways a variety of IT projects used the ideas described in Chapters 1 through 6 and the techniques described in later chapters. While I cannot cover every possible situation, I hope this mix of case studies provides fairly broad coverage of the various environments in which you may find yourself. In addition, they should furnish ideas for using the same techniques in different situations and adjusting your approach based on your current context.
Chapter 7: Case Study: Conference Submission System
This is the story of developing and maintaining the submission system for the Agile2013 and Agile2014 conferences. This was a fairly straightforward project, but it provides the opportunity to position several analysis techniques in their proper context.

Chapter 8: Case Study: Commission System
This case describes what happened when a health insurance company undertook a project to replace multiple commission systems. The case explores some good techniques for projects involving off-the-shelf software and the tendency to gold plate.

Chapter 9: Case Study: Data Warehouse
This case tells the story of a project to incorporate a new source of data into an existing data warehouse. This story explores analysis in a business intelligence project, another environment that can benefit from an agile mindset.

Chapter 10: Case Study: Student Information System
This case explores analysis in a nonprofit setting and focuses on the decisions that need to be made when a project is initially being considered.

Part III: Techniques
In this section I describe a series of techniques that can be helpful in many different settings using my technique brief format. That format covers the following aspects of a technique:

- What it is
- An example
- When to use it
- Why use it
- How to use it
- Caveats and considerations
- Additional resources

Chapter 11: Understanding Stakeholders
This chapter describes some techniques that are helpful for understanding the people you work with. The first two techniques are useful for understanding
the people whose needs you are trying to satisfy—better known as stakeholder analysis. The other two techniques in this chapter will help you better understand the people who are actually going to use the solution you deliver; let’s call this user analysis. The techniques I cover include

- Stakeholder map
- Commitment scale
- User modeling
- Persona

Chapter 12: Understanding Context
Understanding context means familiarizing yourself with the nature of the business and sharing that information with the rest of the team. You want to put the project in the perspective of the overall organization and determine what the project is intended to do. If the project does not support something explicitly related to the organization’s strategy or ongoing operations, don’t do it.

This chapter introduces several techniques for understanding the organization as a whole and using that information to guide decisions about your projects. The techniques described in this chapter are often called strategy analysis (formerly enterprise analysis) in the analyst community.

- The Purpose-Based Alignment Model
- Six questions
- The Context Leadership Model

Chapter 13: Understanding the Need
A key and often overlooked aspect of IT projects is figuring out the real need that must be satisfied, determining if it is worth satisfying, and sharing that understanding with the entire team. If those activities were done more frequently, the story told about IT projects would undoubtedly be much brighter.

In this chapter, I introduce a set of techniques that I have found very helpful for performing those activities:

- Decision filters
- Project opportunity assessment
- Problem statement
Chapter 14: Understanding the Solution(s)
Once we understand the need we’re trying to satisfy and we’ve determined that it’s worth satisfying, we should investigate possible solutions. The plural form is intentional. Project teams often limit themselves by focusing on one possible solution too soon instead of leaving their options open. In many cases there are multiple options.

In this chapter I identify a variety of techniques for exploring multiple solutions and describing the solutions that seem best, all in a way that is meaningful for everyone working on the project:

- Impact mapping
- Story mapping
- Collaborative modeling
- Acceptance criteria
- Examples

Chapter 15: Organizing and Persisting Solution Information
This chapter describes techniques that help teams visualize progress and the aspect of the solution they are working on, as well as a way to persist key information about the solution for future reference. The techniques described in this chapter include

- Discovery board
- Definition of ready
- Delivery board
- Definition of done
- System documentation

Part IV: Resources
In this final part of the book, I provide a couple of resource sections that summarize key definitions and reference sources collected from the rest of the book.

Glossary
It’s always a good practice to establish a common language for your projects. Since I am trying to be very specific about how I refer to certain concepts, and
in the interest of eating my own dog food, I decided to establish a glossary for *Beyond Requirements*. This should help me be consistent in my use of certain words, or at least give you a chance to catch me if I am inconsistent. Words in the glossary appear in bold the first time they are mentioned in the text.

References
Throughout the book I reference several great sources of additional information about the topics I discuss. This section compiles all the references into a single list. Take some time to check out the references listed here; there’s some great stuff.

In addition to the resources included in the book, beyondrequirements.com features additional thoughts on analysis with an agile mindset, new technique briefs, and updates to the material in the book.
Acknowledgments

This is not the first book I have written, but it is the first I took on by myself, or at least that’s what I thought the case was when I started. It turns out that while I’ll be listed as the only author, this book would not have been possible without the help of several people.

There are two people who played the biggest part in how the book looks and reads. Jeff Rains created all the hand-drawn graphics in the book. It was important that the graphics reinforce the idea of having a conversation at a whiteboard. Jeff’s great work allowed me to get that message across while allowing you to be able to read the graphics. Deanna Burghart provided the first line of defense that prevented me from doing horrendous things to the English language. I have worked with Deanna for several years as she edited my pieces for ProjectConnections.com. I knew when I started working on this book a . . . um . . . couple of years ago that I wanted her editorial help. She, as always, did a great job helping me sound like me.

I have been fortunate in my professional life to work and interact with brilliant people who look at things in a slightly different way and who do not hesitate to share their perspective with me. Several of those people played a part in this book, but it’s important that I thank three especially. It is truly an honor and a privilege to be able to fall back on these three to discuss ideas and ways to describe them. Gojko Adzic’s extensive review notes were an immense help during the editing stage and helped me see things from a different and better perspective. Todd Little reviewed most of the book during the final editing stages and, as always, provided practical and insightful advice to help me crystallize my revisions. Chris Matts, long a primary source of cutting-edge, yet eminently practical thought in the space of analysis, generously discussed several ideas for this book and was a key source of many of the more important ones. My understanding of the nuances of analysis and IT project work is due largely to being fortunate enough to know these three practitioners.

I was fortunate to receive feedback from a wide range of professionals. Special thanks go to Robert Bogetti, Sarah Edrie, James Kovacs, Chris Sterling, and Heather Hassebroek for reading and commenting on the entire draft. Their comments were very helpful in shaping and refining my initial thoughts into something that I hope is a bit more coherent. Thanks also to Diane Zajac-Woodie, Deb McCormick, Brandon Carlson, Mary Gorman, Julie Urban,
Pollyanna Pixton, Matt Heusser, Tina Joseph, and Ellen Gottesdiener, who all gave feedback on aspects of the draft.

Finally, thanks to Chris Guzikowski, acquisitions editor at Addison-Wesley, who had the patience to stick with me through the drawn-out writing process, and Jeffrey Davidson, who didn’t let an opportunity go by without nagging me about finishing the book. Jeffrey, I’m not sure if Chris put you up to that or not, but I suspect he’s glad you did, regardless.
About the Author

Kent J. McDonald uncovers better ways of delivering value by doing it and helping others do it. His years of experience include work in business analysis, strategic planning, project management, and product development in a variety of industries, including financial services, health insurance, performance marketing, human services, nonprofit, and automotive. He is active in the business analysis and agile software development communities helping people share stories about what does and does not work.

Kent has a Bachelor of Science degree in industrial engineering from Iowa State University and an MBA from Kent State University.

Kent is also a coauthor of *Stand Back and Deliver: Accelerating Business Agility*. 
This page intentionally left blank
Chapter 8

Case Study: Commission System

Introduction
McMillan Insurance is a midsize health insurance company located in a midsize city in the middle of the United States. McMillan has grown through acquisition, and until recently one of its practices was to let each company keep its own identity when dealing with anyone outside the walls of headquarters. This included the relationships with independent agents and the resulting commission structures. This meant that Arthur, the manager of the commissions area, had to deal with a slew of different very unique commission rules down to the individual agent level, and the resulting hodgepodge of commission “systems” required to administer those different commission plans. McMillan has finished its acquisition binge and now realizes that some commonality needs to be introduced in many areas, including commissions.

Arthur was charged with making the commissions area more efficient, so his first instinct was to find a new commission system that would allow him to administer all the various commission plans in one place, while still maintaining all the unique commission structures. He sat down with a couple of more experienced members of his staff, and they started scouring the Internet for possible products. A quick search revealed several options. (Of course, this should have been obvious just from the seven different software applications McMillan had inherited from the acquired companies, only one of which was built in-house.)

It was at this point that they reached out to IT for some help figuring out what to do. Arthur was a little hesitant to do that at first because he was concerned that IT would want to build something in-house. He was pleasantly surprised when Heather, a business analyst from IT on the team, suggested that instead of immediately going out and looking for specific products they should step back
and think about what need they were trying to satisfy. Heather and Arthur sat down to discuss the current situation and what Arthur hoped to accomplish.

## The Need

As a result of their conversation, Arthur and Heather identified the following objectives:

- Reduce the time it takes to produce commission payments from one week to two days.
- Reduce the time required to set up a new commission plan from six weeks to one week (needed every time a new product is created).
- Reduce the time required to set up a new agent from one day to one hour.

They then discussed the characteristics of a desirable solution. As they were talking, Heather used the Purpose-Based Alignment Model (Chapter 12) to identify commissions as a **parity activity**, and Arthur realized that trying to have unique commission rules for every agent was, in effect, overinvesting in commissions. Data from the existing commission payments indicated that the unique rules did not have a direct impact on what the agents sold, so they were probably not worth the effort that Arthur’s area spent in creating and administering them. Arthur made a note to talk to the sales managers about reducing the complexity of the commission rules.

At this point a team was formed that included Arthur and some of the more experienced members of his staff as well as Heather and a few others from IT. Arthur and Heather described the objectives they had put together and then worked with the team to create decision filters for the project, to make sure everyone was on the same page.

Here are the decision filters they came up with:

- Will this reduce the cycle time for commission payments?
- Will this help us set up a commission plan faster?
- Will this help us set up a new agent faster?

## The Possible Solution(s)

Once the team had a good understanding of what they were trying to accomplish, they decided they needed to identify options for realizing those objectives, starting with reducing the time required for commission payments. They used
impact mapping (Chapter 14) to help them identify options. Several options came up, including simplifying the commission rules and consolidating the multiple commission systems into one. The team also identified multiple options for dealing with the existing systems:

- Build something in-house.
- Revise the existing conglomerate.
- Purchase something.
- Outsource all commissions activity.
- Do nothing.

The team decided that the best route was to start with simplifying the rules for commissions in one of the acquired companies to see if there was any impact on sales. At the same time, they started the search for software to replace all of the existing commission systems. Table 8.1 lists the characteristics that served as criteria for the search.

The team included the optional characteristics as a way of seeing if any commonly used applications used complex rule logic, in case they found data to support the need for unique commission rules.

### The Deliveries of Value

The team split the work into a series of rounds. (They chose that term instead of releases, because not every round involved deploying software.) They weren’t
sure how many rounds they would have at the beginning, but they knew they would be organized along the lines shown in Table 8.2.

The team figured that after the first couple of rounds they would simplify rules and move the units to the new commission system at the same time. They staggered the first few so that they could isolate the changes and get a sense of what impact those changes had on sales.

Lessons Learned

The effort is still going on at the time of this writing, but the team has already learned several lessons:

**Not all problems require a technical solution.** The team found that simplifying the commission rules helped reduce the amount of time required to process commissions a great deal and confirmed their suspicions that unique rules did not have a large impact on sales agent behavior. Even so, the team decided it would be good to consolidate all the processing on a single system.

**You may not realize how good you have it on your side of the fence.** As the team started their search for a new commission system, they decided to include the five purchased systems they were already using to administer parts of their commissions. They found that as a result of simplifying commission rules, one of the systems they already had fit the bill nicely for what they were trying to do. They had to upgrade that commission system several versions, but once they

---

**Table 8.2 Rounds of Work**

<table>
<thead>
<tr>
<th>Round</th>
<th>Contents</th>
</tr>
</thead>
</table>
| 1     | • Simplify the commission rules for Southern Comfort Insurance (SCI).  
       | • Identify a commission system to purchase. |
| 2     | • Implement a commission system in-house.  
       | • Use the commission system for McMillan agents (who already had straightforward commission rules).  
       | • Simplify the commission rules for Western Amalgamated Insurance (WAI). |
| 3     | • Use the new commission system for SCI.  
       | • Phase out the existing commission system for SCI.  
       | • Simplify the commission rules for Eastern Agrarian Insurance (EAI). |
| 4–N   | • Roll out the commission system to the remaining units.  
       | • Simplify the commission rules for the remaining units.  
       | • Phase out the existing commission systems. |
did, they found that their work mainly consisted of creating new interfaces for any data they didn’t have in that system already.

Commercial off-the-shelf (COTS) systems often contain good industry practices. When the team picked the commission system, they found they could use that unit’s commission process for all the other units as well. That process was one suggested by the developers of the existing commission system. Switching to that process for all the units provided even more improvement in overall commissions processing and eased the transition effort since the team didn’t have to come up with new processes for each unit.

Don’t forget change management. Just because the team didn’t have to come up with new processes didn’t make the change completely turnkey. The commissions team did not have much trouble with the change, since over half of the team was involved on the project to switch commission systems, but they had a bit of change management to do with the agents. When they found out that commission structures were changing, most of the agents complained. Loudly. The team found that the best way to help the agents adapt to the change was to give them examples of their own commissions under both the old and the new structures. Most of the agents found that their commissions would stay consistent, or even increase. The only agents whose commissions decreased were those few who had studied the old plans enough to use loopholes to maximize their revenue. These agents were among the highest compensated but were only middle of the pack in terms of actual sales.

Don’t overlook interdependencies with other efforts. The team originally thought they would have to do a lot of work to interface with a new set of systems for each unit they brought onto the new commission system. Shortly into the project, the team caught wind that the accounting and new business systems were also undergoing projects to make things more uniform. The commissions team got together with the other two teams and synced their rollout plans so they affected the same units in the same order, though not necessarily at the same time. That meant that the commissions team did not have to build new interfaces for every additional unit; they just had to revise the ones they had already built.
This page intentionally left blank
Index

A
Acceptance criteria
“On Acceptance Criteria for User Stories,” 192
“Acceptance Criteria vs. Scenarios,” 192
additional resources, 192
appropriate use of, 190
caveats and considerations, 191–192
definition, 188–189, 221
element, 189–190
Growing Agile: A Coach’s Guide to Agile Requirements, 192
guidelines for expressing business rules. See RuleSpeak
mind map of, 189
potential criteria, 190
process description, 190–191
purpose of, 188, 190
RuleSpeak, 191, 192
in system documentation, 216
“On Acceptance Criteria for User Stories,” 192
“Acceptance Criteria vs. Scenarios,” 192
Acceptance test driven development. See Examples (Agile technique)
Actionable metrics
appropriate use of, 36
definition, 221
purpose of, 34
Adaptation, characteristic of initiatives, 11–12
“Adventures in Scaling Agile,” 177
Adzic, Gojko
examples (Agile technique), 62–63, 196
focusing on the desired outcome, 4
impact mapping, 174, 175–176, 177
Specification by Example, 62
Agile Alliance, 213
The Agile Culture: Leading through Trust and Ownership, 150
Agile mindset. See Analysis, with an Agile mindset
“Agile Models Distilled: Potential Artifacts for Agile Modeling,” 188
Agile Project Leadership Network (APLN), 12
Agreed upon, characteristic of objectives, 18
Ambler, Scott, 140, 217
Analysis
cognitive bias. See Cognitive bias, analysis
doing discovery and delivery, 20–23
doing needs and solutions, 15–19
doing outcomes and outputs, 19–20
scope of, xv
Analysis, with an Agile mindset
decision filters, identifying needs, 71
delivery boards, 73
diagram of, 70
information radiators, 73
needs, identifying, 71
possible solutions, identifying, 71–72
release backlog, 72–73
release planning, 72–73
story mapping, 72
user stories, 72–73
visualization boards, 73
Analyst. See Business analyst
Anchoring effect
cognitive bias, 51
definition, 221
APLN (Agile Project Leadership Network), 12
Appropriate practices
vs. best practices, 9–10
definition, 221
Arbitrary decision mechanism
definition, 222
description, 41
Ariely, Dan, 11, 48
Automated testing, 93
Availability heuristic
definition, 222
description, 51
B
BABOK v3, definition, 222
BACCM (Business Analysis Core Concept Model)
core concepts, 16–17
definition, 222
Backbone, definition, 222
Backlog items, in system documentation, 216
Backlogs
failure to identity complete solutions, 186
as wish lists, 186
Bandwagon effect. See also Groupthink
definition, 222
description, 49
Barely sufficient approach
definition, 222
description, 8–9
Baseline, attribute of objectives, 18
BDD (behavior-driven development). See Examples (Agile technique)
Behavior-Driven Development: Using Examples in Conversation to Illustrate Behavior—A Work in Progress, 197
Berndtsson, Johan, 174
Best practices, definition, 9. See also Appropriate practices
“Best Practices for Agile/Lean Documentation,” 217
Bezos, Jeff, 241
Blank, Steve, 26, 230
Books and publications
“On Acceptance Criteria for User Stories,” 192
“Acceptance Criteria vs. Scenarios,” 192
“Adventures in Scaling Agile,” 177
The Agile Culture: Leading through Trust and Ownership, 150
“Agile Models Distilled: Potential Artifacts for Agile Modeling,” 188
Behavior-Driven Development: Using Examples in Conversation to Illustrate Behavior—A Work in Progress, 197
“Best Practices for Agile/Lean Documentation,” 217
Bridging the Communication Gap: Specification by Example and Agile Acceptance Testing, 196
Commitment, 46
Commitment: Novel about Managing Project Risk, 204
Competitive Engineering, 18
“Comprehensive Documentation Has Its Place,” 217
“Customer Guide,” 197
“Decision Filters,” 163
“Definition of Done,” 213
“Definition of Ready,” 206
Discover to Deliver, 21
The Entrepreneur’s Guide to Customer Development, 26
The Four Steps to the Epiphany, 26
“Getting the Most out of Impact Mapping,” 174
Growing Agile: A Coach’s Guide to Agile Requirements, 192
How to Measure Anything, 32
“How Visualization Boards Can Benefit Your Team,” 204, 210
“Impact Mapping,” 177
Impact Mapping: Making a Big Impact with Software Products, 177
“Inclusive Modeling: User Centered Approaches for Agile Software Development,” 188
Inspired: How to Create Products Customers Love, 163, 167
“An Interview with the Authors of ‘Stand Back and Deliver: Accelerating Business Agility,’” 163
Lean Analytics, 32
Manage Your Project Portfolio, 69
“Personas, Profiles, Actors, & Roles: 
Modeling Users to Target Successful Product Design,” 137, 140
Predictably Irrational, 48
Rath & Strong’s Six Sigma Pocket Guide, 132
The Software Requirements Memory Jogger: A Pocket Guide to 
Help Software and Business Teams Develop and Manage 
Requirements, 169
Specification by Example: How Successful Teams Deliver the Right 
Software, 196
“Stakeholder Analysis,” 129
Stand Back and Deliver: Accelerating 
Business Agility, 10, 147, 150, 157, 163
Thinking, Fast and Slow, 48
User Stories Applied: For Agile 
Software Development, 137
User Story Mapping: Discover the 
Whole Story, Build the Right 
Product, 182
“Using a Definition of Ready,” 206
Box, George E. P., 61
Break the Model approach definition, 222
Feature Injection, 62
Bridging the Communication Gap: 
Specification by Example and Agile Acceptance Testing, 196
Budgeting
conference submission system case study, 90–91, 94
vs. estimating, 94
Build-Measure-Learn loop
definition, 223
description, 30–31
in the lean startup process, 29–31
leap-of-faith assumptions, 31
Business analysis, definition, 223
Business Analysis Core Concept Model 
(BACCM)
core concepts, 16–17
definition, 222
Business analysts
cognitive bias, 49–50
definition, 223
Business case
definition, 223
presenting required information, 43
Business domain model, definition, 223
Business goals. See Goals
Business objectives. See Objectives
Business rule catalog, in system documentation, 213
Business rules, guidelines for expressing.
See RuleSpeak
Business value
case studies, 57–58
definition, 56–57, 223
Feature Injection, 56–57
in Feature Injection, 56
Business value model
definition, 223
Feature Injection, 57–58
C
Cagan, Marty, 163, 167
Campbell-Pretty, Em, 177
Carlson, Brandon
adding themes, 84–86
budgeting Agile 2014, 90–92
define-build-test, 81–84
identifying solutions, 78–81
Case studies
financial services company, 57
Mercury space program, 47
minimum viable products, 64–65
new payroll system, 60
nurse line service, 44
organizing conferences, 58
Case studies, commission system change management, 99
commercial off-the-shelf (COTS) systems, 99
deliveries of value, 97–98
identifying solutions, 96–97
interdependencies, 99
lessons learned, 98–99
needs assessment, 96
non-technical solutions, 98
rounds of work, 97–98
Case studies, conference submission system
acceptance criteria, 189–190
adding themes, 84–90
Case studies, conference submission system (continued)
automated testing, 93
budgeting, 90–91
budgeting vs. estimating, 94
cards for, 169
conveying requirements, 82
define-build-test, 81–84
deliveries of value, 79–92
differentiating activities, 78
distributed teams, 93–94
documentation, 91–92
documented examples (Agile technique), 193–194
documentation, example, 214
identifying solutions, 78
dependent user roles and activities, 79
lessons learned, 92–94
letting approach dictate tools, 93
needs assessment, 77–78
project opportunity assessment, 165
public commenting, 87–88
reporting, 88
story map, 79, 88–89
story mapping, example, 179
stubbed identity service, 82
system documentation, example, system documentation, example
team trust and transparency, effect on documentation, 92
when even Scrum is overkill, 92

Case studies, data warehouse decision filters, 104
deliveries of value, 103–109
identifying solutions, 102–103
lessons learned, 110
needs assessment, 101–102
performance metrics, 109

Case studies, student information system
cost-benefit analysis, 119
elements, 142, 144
functional requirements, 117–118
identifying solutions, 114–118
lessons learned, 118–119
needs assessment, 111–114
Purpose-Based Alignment Model, 142, 144
RFPs, 114–118, 119
six questions, 147–148
solutions looking for problems, 119
Causal metrics, 35–36, 36

Cauwenbergh, Pascal Van, 58–59
Change in the BACCM, 16
definition, 224
Change management, case study, 99
Claxton-Huang, Jane, 66
Clustering illusion
cognitive bias, 51
definition, 224
Cockburn, Alistair, 12, 222
Cognitive bias affecting analysts, 49–50
affecting stakeholders, 49. See also Bandwagon effect; Curse of knowledge; Herd instinct; Response bias
definition, 224
overview, 48–50
Predictably Irrational, 48
Thinking, Fast and Slow, 48
Cognitive bias, analysis anchoring effect, 51
availability heuristic, 51
clustering illusion, 51
deformation professionelle, 51
focusing effect, 51
frequency illusion, 51
observation selection bias, 51
recency bias, 51
Semmelweis reflex, 51
sharpshooter fallacy, 51
survivorship bias, 51
Cognitive bias, decision making false consensus effect, 52
fist of five, 53
group attribution error, 52
irrational escalation, 52
loss aversion, 53
mitigating, 53
sunk cost bias, 52–53
throwing good money after bad, 52
Cognitive bias, elicitation affecting analysts, 49–50
bandwagon effect, 49
biases affecting stakeholders, 49
confirmation bias, 50
the curse of knowledge, 49
framing effect, 50
herd instinct, 49
mirror imaging, 50
mitigating, 49, 50
observer-expectancy effect, 50
response bias, 49
Cohn, Mike, 137
Collaboration
characteristic of initiatives, 5–7
vs. consensus, 6
definition, 224
examples, 6
teams vs. workgroups, 6
Collaborative modeling
additional resources, 188
“Agile Models Distilled: Potential Artifacts for Agile Modeling,” 188
appropriate use of, 184–186
caveats and considerations, 188
text diagrams, 183
data dictionaries, 183
definition, 182–183, 224
element, 183–184
functional decomposition, 183
glossaries, 183
“Inclusive Modeling: User Centered Approaches for Agile Software Development,” 188
logical data models, 183
organization charts, 183
process description, 186–187
process flow, 183
purpose of, 186
report mockups, 183
state transition diagrams, 183, 184
techniques, 183
value stream maps, 183
wireframes, 183
Commercial off-the-shelf (COTS) systems, case study, 99
Commission system. See Case studies, commission system
Commit to, transform, or kill, definition, 224
Commitment, 46
Commitment: Novel about Managing Project Risk, 204
Commitment scale. See also Stakeholder engagement
appropriate use of, 130
caveats and considerations, 132
definition, 129, 224
type, 129–130
process description, 131–132
purpose of, 124, 130
Rath & Strong’s Six Sigma Pocket Guide, 132
Commitments vs. options, 46–47. See also Real Options
Communicating a decision, 45–46
Company building, definition, 26, 224
Company creation, definition, 26, 225
Competitive Engineering, 18
Complexity attributes, Context Leadership Model, 151
Complexity risks, mitigating, 156
“Comprehensive Documentation Has Its Place,” 217
Conference organizing, case study, 58
Conference submission system. See Case studies, conference submission system
Confirmation bias. See also Observer-expectancy effect
definition, 225
description, 50
Consensus, decision mechanism
vs. collaboration, 6
definition, 225
description, 41
Constraint, attribute of objectives, 18
Context
in the BACCM, 16
characteristic of initiatives, 9–10
definition, 225
Context diagrams
collaborative modeling, 183
definition, 225
Context Leadership Model. See also Purpose-Based Alignment Model;
Six questions
advantages of a two-by-two matrix, 9
appropriate use of, 154
caveats and considerations, 156–157
complexity attributes, 151
complexity risks, mitigating, 156
definition, 150, 225
type, 151–154
process description, 154–157
purpose of, 155
Context Leadership Model (continued)

Stand Back and Deliver: Accelerating Business Agility, 157
uncertainty attributes, 152
uncertainty risks, mitigating, 156
Cooper, Alan, 138, 140
Cooper, Brant, 26
Core concept, definition, 225–226
Correlated metrics, 35–36, 36
COTS (commercial off-the-shelf) systems, case study, 99
Croll, Alistair, 32, 34, 37
The curse of knowledge
cognitive bias, 49
definition, 226
Customer, definition, 226
Customer development
definition, 26, 226
in IT projects, 26
in the lean startup process, 25–29
Customer discovery
definition, 26, 226
in IT projects, 28
process description, 27
“Customer Guide,” 197
Customer validation, definition, 26, 226
Customer-problem-solution hypothesis,
definition, 226

D
Dalton, Nigel, 43
Data dictionaries
collaborative modeling, 183
definition, 227
Data warehouse. See Case studies, data warehouse
Deadlines, determining, 47
Decider
vs. decision leader, 40
definition, 227
determining, 39–41
Decider decides with discussion
definition, 227
description, 41
Decider decides without discussion
definition, 227
description, 42
Deciding wisely, characteristic of initiatives, 10–11
Decision filters
additional resources, 163
appropriate use of, 161
case study, 104
caveats and considerations, 163
“Decision Filters,” 163
definition, 160, 227
example, 160
identifying needs, 71
“An Interview with the Authors of ‘Stand Back and Deliver: Accelerating Business Agility,’” 163
process description, 161–162
purpose of, 161
role in delivering value, 4
Stand Back and Deliver: Accelerating Business Agility, 163
“Decision Filters,” 163
Decision leader
vs. decider, 40
definition, 227
Decision maker, determining, 39–41
Decision making. See also Cognitive bias
building support, 45
communicating the decision, 45–46
determining a deadline, 47
determining required information, 42
determining the decision maker, 39–41
enacting the decision, 46
options vs. commitments, 46–47
process structure, 39
Real Options, 46–48
timely decisions, 43–44
Decision mechanisms
arbitrary, 41
consensus, 41
decider decides with discussion, 41
decider decides without discussion, 42
delgation, 41–42
majority vote, 42
negotiation, 42
spontaneous agreement, 42. See also Groupthink
Deep Thought Academy. See Case studies, student information system
Define-build-test, case study, 81–84
Definition of done
additional resources, 213
appropriate use of, 211
INDEX 255

definition, 211, 227
“Definition of Done,” 213
example, 211
process description, 212
purpose of, 211–212
“Definition of Done,” 213
Definition of ready
additional resources, 206
appropriate use of, 205
caveats and considerations, 206
definition, 204, 227
“Definition of Ready,” 206
example, 204
process description, 205
purpose of, 205
“Using a Definition of Ready,” 206
“Definition of Ready,” 206
Déformation professionnelle
cognitive bias, 51
definition, 227
Delegation decision mechanism, 41–42
Delivering value
case studies, 79–92, 97–98, 103–109
characteristic of initiatives, 4–5
minimum viable product (MVP), 63–64
Delivering value, Feature Injection. See also MMF (minimum marketable feature); MVP (minimum viable product)
Break the Model approach, 62
business value, 56–57
business value model, 57–58
elements, as specifications, 61–63
identifying value, 56–59
Increase Revenue, Avoid Costs,
Improve Service (IRACIS), 56–57
injecting the features, 59–61
Key Example pattern, 62
overview, 55–56
role of value points, 59
stakeholder expectations, 60–61
story mapping, 60
Delivery
analyzing, 20–23
definition, 21, 228
Delivery boards. See also Discovery boards
additional resources, 210
analysis with an Agile mindset, 73
appropriate use of, 208
caveats and considerations, 210
Commitment: Novel about Managing Project Risk, 204
creating, 209
definition, 206–207, 228
example, 207–208
process description, 209
purpose of, 208
using, 209
Deming, W. Edwards, 30
Deming Cycle. See PDSA (Plan-Do-Study-Act) cycle
Deming Wheel. See PDSA (Plan-Do-Study-Act) cycle
Denne, Mark, 65–67
Denning, Steve, 57
Design
definition, 228
vs. requirements, 22–23
Design thinking
definition, 228
in the development process, 22–23
Differentiating activities
case study, 78
definition, 228
identifying, 147
in the Purpose-Based Alignment Model, 143, 146
Discover to Deliver, 21
Discovery
analyzing, 20–23
definition, 21, 228
Discovery boards. See also Delivery boards
additional resources, 204
appropriate use of, 201
caveats and considerations, 203–204
Commitment: Novel about Managing Project Risk, 204
creating, 202
definition, 200, 228
example, 200–201
“How Visualization Boards Can Benefit Your Team,” 204, 210
process description, 202–203
purpose of, 201–202
using, 202–203
Distributed teams, 93–94
Documentation. See System documentation
Domain, definition, 228
Domingues, Ingrid, 174
Done, definition of. See Definition of done

E
Elicitation
cognitive bias. See Cognitive bias, elicitation
definition, 228
Elssamadisy, Amr, 163
Email, conveying requirements with, 82
Enacting a decision, 46
Enterprise, definition, 229
The Entrepreneur's Guide to Customer Development, 26
Examples (Agile technique)
additional resources, 196–197
appropriate use of, 194–195
Behavior-Driven Development: Using Examples in Conversation to Illustrate Behavior—A Work in Progress, 197
Bridging the Communication Gap: Specification by Example and Agile Acceptance Testing, 196
caveats and considerations, 196
“Customer Guide,” 197
in a decision table, 193
definition, 192–193, 229
example, 193–194
formats, 192
Framework for Integrated Test (Fit) format, 192, 194–195
Gherkin format, 192, 195, 197
Key Example pattern, 62
process description, 195–196
purpose of, 195
Specification by Example, 62
Specification by Example: How Successful Teams Deliver the Right Software, 196
as specifications, 61–63
system documentation, 216
in system documentation, 216
Examples (used in this book)
acceptance criteria, 189–190
collaboration, 6
Facilitate, definition, 229
False consensus effect in decision making, 52
definition, 229
Feature, definition, 66, 229
Feature files (example), 83–84
Feature Injection
Break the Model approach, 62
business value, 56–57
business value model, 57–58
case study, 60
definition, 229
examples, as specifications, 61–63
identifying value, 56–59
Increase Revenue, Avoid Costs, Improve Service (IRACIS), 56–57
INDEX

injecting the features, 59–61
Key Example pattern, 62
overview, 55–56
role of value points, 59
stakeholder expectations, 60–61
story mapping, 60
Fichtner, Abby, 25–26
Financial services company, case study, 57
Fist of five
 in decision making, 53
definition, 229–230
Fit. See Framework for Integrated Test
Fitzpatrick, Rob, 28
Focusing effect
definition, 230
description, 51
The Four Steps to the Epiphany, 26
Framework for Integrated Test, definition, 230
Framework for Integrated Test (Fit) format
appropriate use of, 194–195
of examples (Agile technique), 192
Framing effect
cognitive bias, 50
definition, 230
Frequency illusion
cognitive bias, 51
definition, 230
Functional decomposition
collaborative modeling, 183
definition, 230
Functional requirements, case study, 117–118, 119
Functionalist, definition, 230

G
Gamestorming, 129
Geary, Chris, 46, 204, 210
“Get out of the building” technique
definition, 230
description, 28
“Getting the Most out of Impact Mapping,” 174
Gherkin format
definition, 231
eexamples (Agile technique), 192, 195
online description of, 197
Gilb, Tom, 18
Glenn, John, 47
Glossaries, in collaborative modeling, 183
Glossaries, in system documentation
definition, 213, 231
for intermediate communication, 9
Glossary, of terms in this book, 221–243
Goals
definition, 17, 231
role in delivering value, 4
Good practices. See Appropriate practices
Gorman, Mary, 21
Gottesdiener, Ellen, 21, 169
Group attribution error
in decision making, 52
definition, 231
Groupthink. See also Spontaneous agreement
bandwagon effect, 49
definition, 231
herd instinct, 49
Growing Agile: A Coach’s Guide to Agile Requirements, 192
Guidelines for expressing business rules. See RuleSpeak

H
Herd instinct. See also Groupthink
definition, 231
in elicitation, 49
High influence/high interest stakeholders, 128
High influence/low interest stakeholders, 128
Holst, Darrin
adding themes, 84–86
budgeting Agile 2014, 90–92
define-build-test, 81–84
identifying solutions, 78–81
How to Measure Anything, 32
“How Visualization Boards Can Benefit Your Team,” 204, 210
Hubbard, Douglas, 32

I
Impact mapping
additional resources, 177
“Adventures in Scaling Agile,” 177
align context, 175–176
appropriate use of, 173–176
Impact mapping (continued)
caveats and considerations, 177
definition, 173, 231
discover context, 175
element, 173–174
experiment context, 175
“Getting the Most out of Impact Mapping,” 174
“Impact Mapping,” 177
Impact Mapping: Making a Big Impact with Software Products, 177
iterate context, 175
key questions, 173
process description, 176–177
purpose of, 176
useful contexts, 174–175
“Impact Mapping,” 177
Impact Mapping: Making a Big Impact with Software Products, 177
“Inclusive Modeling: User Centered Approaches for Agile Software Development,” 188
Increase Revenue, Avoid Costs, Improve Service (IRACIS), 56–57
Information radiators, definition, 232. See also Delivery boards; Discovery boards
Initiatives
definition, 232
desirable characteristics of, 3
Injecting features. See Feature Injection
Inspired: How to Create Products Customers Love, 163, 167
Interdependencies, case study, 99
“An Interview with the Authors of ‘Stand Back and Deliver: Accelerating Business Agility,’” 163
Inventory turn, definition, 232
INVEST (independent, negotiable, valuable, estimable, small, testable), 81
IRACIS (Increase Revenue, Avoid Costs, Improve Service), 56–57
Irrational escalation
in decision making, 52
definition, 232
IT (Information Technology), definition, 232
IT project, definition, 233
Iteration
characteristic of initiatives, 7–8
definition, 232
K
Kahneman, Daniel, 11, 48
Keogh, Liz, 188, 192, 197
Key Example pattern, 62
Kraft, Chris, 47
L
Lacey, Mitch, 213
Lagging metrics, 35, 36
Laing, Samantha, 192
Leadership style, determining. See Context Leadership Model
Leading indicator, definition, 233
Leading metrics, 35, 36
Lean Analytics, 32
Lean startup
Build-Measure-Learn loop, 29–31
customer development, 25–29
metrics, 31–38
Lean TECHniques, 78
Leap-of-faith assumptions, 31
Lessons learned, case studies
commission system case study, 98–99
conference submission system case study, 92–94
data warehouse case study, 110
student information system case study, 118–119
Linders, Ben, 206
Little, Todd
Context Leadership Model, 157
decision filters, 163
Purpose-Based Alignment Model, 147
six questions, 150
software prioritization, 91
Logical data models
collaborative modeling, 183
definition, 233
Loss aversion
in decision making, 53
definition, 233
Low influence/high interest stakeholders, 128
Low influence/low interest stakeholders, 128
INDEX 259

M
Maassen, Olav, 46, 204, 210
Majority vote decision mechanism
definition, 233
description, 42
Mamoli, Sandy, 192
Manage Your Project Portfolio, 69
Marketable, definition, 66
Matts, Chris
Break the Model approach, 62
business value, 56
business value model, 58
Commitment, 46
discovery boards, 204, 210
Feature Injection, 55–56
Real Options, 46, 237
visualization board, 242
Maximizing work not done, 8–9
McDonald, Kent
decision filters, 163
discovery boards, 204, 210
Purpose-Based Alignment Model, 147
system documentation, 217
McMillan Insurance. See Case studies, commission system; Case studies, data warehouse
Measurable, characteristic of objectives, 18
Mercury space program, case study, 47
Metadata, in system documentation, 214
Method, attribute of objectives, 18
Methodology, definition, 233
Metrics
correlated vs. causal, 35–36, 36
creating, 36–37
desirable characteristics, 32–33
example, 33
exploratory vs. reporting, 34–35, 36
leading vs. lagging, 35, 36
in the lean startup process, 31–38
One Metric That Matters (OMTM), 37
qualitative vs. quantitative, 33, 36
for specific situations, 36–37
vanity vs. actionable, 33, 36
MindTools, 129
Minimum, definition, 66
Mirror imaging
definition, 234
in elicitation, 50
MMF (minimum marketable feature)
definition, 233
description, 65–67
feature, definition, 66
marketable, definition, 66
minimum, definition, 66
vs. minimum viable product, 66–67
Modeling
BACCM (Business Analysis Core Concept Model), 16–17, 222
Models. See also Collaborative modeling; Context Leadership Model; Purpose-Based Alignment Model; User modeling
BACCM (Business Analysis Core Concept Model), 16–17
business domain model, 223
business value model, 57–58, 223
in system documentation, 216
wrong vs. useful, 61
The Mom Test
definition, 234
description, 29
MVP (minimum viable product)
case study, 64–65
definition, 234
description, 63
vs. minimum marketable features, 66–67
purpose of, 64
N
Name, attribute of objectives, 18
Needs
in the BACCM, 16
definition, 234
origins of, 19
separating from solutions, 19
Needs assessment
with an Agile mindset, 71
case studies, 77–78, 96, 101–102, 111–114
case study, 77–78
overview, 15–19
Needs assessment techniques. See also specific techniques
decision filters, 160–163
problem statement, 167–169
project opportunity assessment, 163–167
Negotiation decision mechanism
- definition, 234
- description, 42

Nickolaisen, Niel
- Context Leadership Model, 157
decision filters, 160, 163
Purpose-Based Alignment Model, 142, 147
six questions, 150, 239

Non-technical solutions, case study, 98
Nurse line service, case study, 44

O

Objectives
- attributes for, 18
- characteristics of, 18
definition, 17, 234
role in delivering value, 4

Observation selection bias
in analysis, 51
definition, 234

Observer-expectancy effect. See also Confirmation bias
definition, 234
in elicitation, 50

OMTM (One Metric That Matters)
definition, 235
description, 37

Options vs. commitments, 46–47. See also Real Options

Organization (legal entity), definition, 235
Organizational charts
collaborative modeling, 183
definition, 235
determining a decision maker, 40–41

Outcomes
analyzing, 19–20
definition, 235

Outputs
analyzing, 19–20
definition, 235

P

Parity activities
- case study, 96
definition, 235
vs. purpose, 146
Purpose-Based Alignment Model, 143, 146

Partner activities, 143
Patton, Jeff
- personas, 139, 140
- story mapping, 178, 180–181, 182
- user modeling, 137
Payroll system, case study, 60
PDSA (Plan-Do-Study-Act) cycle
definition, 236
description, 30

Performance metrics, case study, 109
Permissions, in system documentation, 214

Personas
- additional resources, 140
- appropriate use of, 139
caveats and considerations, 140
definition, 138, 235
example, 138
The Inmates Are Running the Asylum: Why High-Tech Products Drive Us Crazy and How to Restore the Sanity, 140
“Personas: An Agile Introduction,” 140
process description, 139–140
purpose of, 124, 139
useful characteristics, 139
“Personas, Profiles, Actors, & Roles: Modeling Users to Target Successful Product Design,” 137, 140

Pivot, definition, 236
Pixton, Pollyanna
- Context Leadership Model, 157
decision filters, 163
Purpose-Based Alignment Model, 147
six questions, 150

Plan-Do-Study-Act (PDSA) cycle
definition, 236
description, 30

Pols, Andy, 58

Post-mortems. See Lessons learned
Predictably Irrational, 48

Problem statement
appropriate use of, 168
caveats and considerations, 169
components of, 167
definition, 167, 236
example, 167–168
process description, 168
Index

purpose of, 168

The Software Requirements Memory Jogger: A Pocket Guide to Help Software and Business Teams Develop and Manage Requirements, 169

Problem-solution fit, definition, 236

Process flow
  collaborative modeling, 183
  definition, 236
  in system documentation, 214

Product, definition, 236

Program, definition, 237

Project documentation
  definition, 237
  description, 216–217

Project opportunity assessment
  appropriate use of, 165
  caveats and considerations, 166–167
  definition, 163, 237
  example, 164–165

Inspired: How to Create Products Customers Love, 163, 167
  process description, 166
  purpose of, 166
  questions for, 164

Projects
  definition, 237
  stigma associated with, xvii

Public commenting, case study, 87–88

Pull systems, 56

Purpose vs. parity, 146

Purpose-Based Alignment Model. See also Context Leadership Model;
  Six questions
  advantages of a two-by-two matrix, 9
  appropriate use of, 145
  case studies, 96, 113, 119–120
  case study, 142, 144
  caveats and considerations, 146–147
  definition, 237
  description, 142–144
  process description, 145–146
  purpose vs. parity, 146

Stand Back and Deliver: Accelerating Business Agility, 147

Purpose-Based Alignment Model, quadrants
  differentiating activities, 143, 146, 147
  examples, 142, 144

parity activities, 143, 146

partner activities, 143

“Who cares” activities, 144

Q

Qualitative metrics, 33, 36

Quantitative metrics, 33, 36

R

Rath & Strong Management Consultants, 132

Rath & Strong’s Six Sigma Pocket Guide, 132

Ready, definition of. See Definition of ready

Real Options
  definition, 237
  description, 46–48

Realistic, characteristic of objectives, 18

Recency bias
  in analysis, 51
  definition, 237

Reflection, characteristic of initiatives, 11–12

Rehearsal. See Iteration

Release, definition, 237

Release backlog
  during analysis, 72–73
  definition, 238

Release planning
  during analysis, 72–73
  definition, 238

Report mockups
collaborative modeling, 183
  definition, 238

Reporting, case study, 88

Reporting metrics, 34–35, 36

Requirements
  case study, 117–118, 119
  conveying, 82
  definition, 238
  vs. design, 22–23
  expressing, 89–90

Growing Agile: A Coach’s Guide to Agile Requirements, 192

The Software Requirements Memory Jogger: A Pocket Guide to Help Software and Business Teams Develop and Manage Requirements, 169
Resources. See Books and publications
Response bias
  definition, 238
  effects on stakeholders, 49
Retrospectives
  characteristic of initiatives, 11–12
  definition, 238
  description, 11–12
  examples of. See Lessons learned, case studies
RFP (requests for proposal), case study, 114–118, 119
Ries, Eric
  *The Lean Startup*, 25, 29
  leap-of-faith assumptions, 31
  on metrics, 36
  minimum viable product, 63–64
Risk, definition, 238
Risk management, 204
Ross, Ron
  acceptance criteria, 191, 192
  RuleSpeak, 191, 192, 238
Rothman, Johanna
  commit to, transform, or kill, 224
  *Manage Your Project Portfolio*, 69
  on portfolio projects, 69
Royce, Winston, 20–21
RuleSpeak
  acceptance criteria, 191, 192
  definition, 238
S
Scope
  definition, 238
  role in delivering value, 4
Scrum.org, 213
Semmelweis, Ignaz, 239
Semmelweis reflex
  in analysis, 51
  definition, 239
Service, definition, 239
Shared vision techniques. See also specific techniques
  decision filters, 160–163
  problem statement, 167–169
  project opportunity assessment, 163–167
Sharpshooter fallacy
  definition, 241
  description, 51
Shewhart, Walter, 30, 236
Simplification
  barely sufficient approach, 8–9
  characteristic of initiatives, 8–9
  maximizing work not done, 8–9
  perfect as the enemy of the good, 8–9
Six questions. See also Context
  Leadership Model; Purpose-Based Alignment Model
  *The Agile Culture: Leading through Trust and Ownership*, 150
  appropriate use of, 148
  case study, 148
  caveats and considerations, 149–150
  definition, 147–148, 239
  desired answers, 149
  example, 148
  process description, 149
  purpose of, 148–149
SMART (specific, measurable, agreed upon, realistic, time framed), 17–18
SME (subject matter expert)
  definition, 241
  in the development process, 12
Solution identification, case studies, 78–81, 96–97, 114–118
Solution identification, refining. See also specific techniques
  definition of done, 211–213
  definition of ready, 204–206
  delivery boards, 206–210
  discovery boards, 200–204
  system documentation, 213–217
Solution identification techniques. See also specific techniques
  acceptance criteria, 188–192
  collaborative modeling, 182–188
  examples (Agile technique), 192–197
  impact mapping, 173–177
  story mapping, 177–182
Solutions
  analyzing, 15–19
  in the BACCM, 16
definition, 239
identifying (case study), 78
identifying, with an Agile mindset, 71–72
origins of, 19
separating from need, 19
Solutions looking for problems, 119
Specific, characteristic of objectives, 18
Specification by example. See Examples (Agile technique)
*Specification by Example: How Successful Teams Deliver the Right Software*, 196
Sponsors
definition, 239
as source of needs, 19
Spontaneous agreement decision mechanism
definition, 239
selecting a decision mechanism, 42
Stakeholder analysis
definition, 240
types of stakeholders, 123–124. See also specific types
“Stakeholder Analysis,” 129
Stakeholder engagement. See also Commitment scale
high influence/high interest, 128
high influence/low interest, 128
low influence/high interest, 128
low influence/low interest, 128
Stakeholder expectations, Feature Injection, 60–61
Stakeholder maps
with actions, 127–128
additional resources, 129
appropriate use of, 126
caveats and considerations, 129
description, 124–125
engagement levels, 127–128
example, 125
primary outcomes, 125
process description, 126–128
purpose of, 126
“Stakeholder Analysis,” 129
uses for, 124
Stakeholders
in the BACCM, 16
building support for decision making, 45
definition, 239
“get out of the building,” 28
talking to, 28
types of, 123. See also specific types
*Stand Back and Deliver: Accelerating Business Agility*, 10, 147, 150, 157, 163
Startup, 240. See also Lean startup
State transition diagrams
collaborative modeling, 183, 184
definition, 240
Story mapping
additional resources, 182
during analysis, 72
appropriate use of, 178
as a backlog visualization tool, 181
case study, 79, 88–89
caveats and considerations, 182
definition, 177, 240
as an elicitation tool, 180–181
equivalent, 178, 179
Feature Injection, 60
process description, 180–181
purpose of, 178
*User Story Mapping: Discover the Whole Story, Build the Right Product*, 182
Storyboards, definition, 240
Strategy, definition, 240
Strategy analysis, definition, 240–241
Stubbed identity service, 82
Student information system. See Case studies, student information system
Subject matter expert (SME)
definition, 241
in the development process, 12
Sunk cost bias, 52–53
Survivorship bias
in analysis, 51
definition, 241
System documentation
acceptance criteria, 216
additional resources, 217
appropriate use of, 214
backlog items, 216
“Best Practices for Agile/Lean Documentation,” 217
business rule catalog, 213
caveats and considerations, 215–217
System documentation (continued)
“Comprehensive Documentation Has Its Place,” 217
contents of, 213–214
contribution to the desired outcome, 5
definition, 213–214, 241
effects of teams, 91–92
effect, 214
examples (Agile technique), 216
glossary, 213
metadata, 214
models, 216
permissions, 214
process description, 215
process flows, 214
project documentation, 216–217
purpose of, 214–215
user interfaces, 214

T
Target, attribute of objectives, 18
Teams
definition, 241
distributed, 93–94
organizational context. See Context Leadership Model; Purpose-Based Alignment Model; Six questions
trust and transparency, effect on
documentation, 92
vs. workgroups, 6
Texas sharpshooter fallacy
definition, 241
description, 51
Themes
case study, 84–90
definition, 241
Thinking, Fast and Slow, 48
Three amigos, definition, 241
Throwing good money after bad, 52
Time framed, characteristic of objectives, 18
Timely decisions, 43–44
Tools, dictated by your approach, 93
Triple constraints
definition, 241
role in delivering value, 4
Tversky, Amos, 48
Two-pizza rule, 241

U
Uncertainty attributes, 152
Uncertainty risks, mitigating, 156
Units, attribute of objectives, 18
User analysis
definition, 242
description, 124
User interfaces, in system documentation, 214
User modeling
additional resources, 137
appropriate use of, 135
brainstorming users, 133–134
caveats and considerations, 137
definition, 133, 242
example, 133–135
organizing and consolidating users, 134
“Personas, Profiles, Actors, & Roles: Modeling Users to Target Successful Product Design,” 137
process description, 135–137
purpose of, 124, 135–136
refining user roles, 135
user roles and descriptions, example, 135
User Stories Applied: For Agile Software Development, 137
User roles
case study, 79
definition, 242
user modeling, 133–135
User stories
during analysis, 72–73
definition, 242
User Stories Applied: For Agile Software Development, 137
User Story Mapping: Discover the Whole Story, Build the Right Product, 182. See also Story mapping “Using a Definition of Ready,” 206

V
Value
in the BACCM, 16
definition, 242
delivering. See Delivering value
identifying, case study, 57
Value points
  definition, 242
  in Feature Injection, 59
  identifying features, 59
Value stream maps
  collaborative modeling, 183
  definition, 242
Van Cauwenberghe, Pascal, 58–59
Vanity metrics
  appropriate use of, 36
  definition, 242
  description, 34–35
Vision, shared. See Shared vision techniques
Visualization boards
  during analysis, 73
  definition, 242
Vlaskovits, Patrick, 26

W
  Waterfall planning technique, 20–21
  “Who cares” activities
    definition, 242
    Purpose-Based Alignment Model, 144
Williams, Walter, 47–48
Wireframes
  collaborative modeling, 183
  definition, 243
Work groups
  definition, 243
  vs. teams, 6

Y
  Yoskovitz, Benjamin, 32, 34, 37