

# Table of Contents

<b>Preface</b> .....	<b>ix</b>
<b>1. The Foundations of Product Thinking</b> .....	<b>1</b>
Case Study	2
The First Attempt, with No Scenarios	2
Meh Results	3
What Went Wrong?	4
A Second Attempt with Scenarios	4
Scenarios as Stories That Inspire	5
Scenarios That Capture User Interviews	6
Scenarios That Highlight Product Gaps	6
Scenarios That Highlight a Frictionful Experience	7
Scenarios That Validate the Feature You're Planning to Build	7
Scenarios as Tests	8
Searching for the Key Answers	8
So, What's the Scenario?	8
A Motivation	9
A Persona	11
A Simulation	14
How to Use Scenarios?	17
Chapter Summary	18
Exercises	18
Answers	19

# Part I. Develop

<b>2. Guiding Users Through Your Product.....</b>	<b>23</b>
Case Study Introduction	24
Scenarios in the User Journey	26
The Discovery Scenario	26
Product Discovery Mapping	27
Leverage Users' Knowledge	29
Offer Multiple Routes to Discovery	30
Gracefully Reveal Complexity	32
Multipersona Design	33
Turn Unknown Unknowns into Known Unknowns	34
The Understanding Scenario	35
Picking Understandable Names	35
Classic Naming Advice Revisited	35
Giving Redundant Explanations	39
The Usage Scenario	41
Optimizing the Whole User Journey	42
The Limits of Signifiers	43
Chapter Summary	44
Exercises	44
Answers	45
<b>3. Errors and Warnings.....</b>	<b>47</b>
The Value of Diagnostics	48
Scenarios for Diagnostics	49
Categorizing Error Scenarios	50
Categorizing Errors in Practice	51
Warning and Error Messages	53
Case Study Introduction	53
Provide Context	54
Make Error and Warning Messages Actionable	57
Raise Errors at the Interface	58
Upfront Validations	59
Repackage Errors	60
Raise Programmable Errors	61
Raise Specific Errors	61
Group Errors According to Scenario Category	62
Keep Information Around for Diagnostics	62
Diagnose Early	65
Do Static Validations	65
Validate Upfront	66

Let Them Test	66
Request User Confirmations	67
Chapter Summary	68
Exercises	69
Answers	70

## Part II. Deliver

<b>4. Experiencing Your Own Product.....</b>	<b>73</b>
Tests as Dogfooding	74
What Kinds of Tests Should You Write?	75
Scenario Tests	78
Functional Tests	80
End-to-End Tests	80
User Acceptance Testing	81
Documentation-Driven Development	82
Documentation Shouldn't Be Load Bearing	83
Scenarios for Documentation Readers	83
Writing Documentation as Dogfooding	89
Friction Logging	90
Writing Friction Logs	91
Receiving Friction Logs	92
Friction Logging Culture	93
Samples	94
Chapter Summary	95
Exercises	95
Answers	96
<b>5. Keeping Up with Users.....</b>	<b>99</b>
A New Job Description—Digital Twin Caretaker	101
Designing for Change	102
Case Study Introduction	103
The Technological Underpinnings of Change	104
Getting User Feedback	108
Beta Versions Lower the Blast Radius of Failures	108
Feedback Widgets Help People Raise Their Voices	109
Champions Programs Offer Depth	109
Surveys Offer Breadth	110
The User Support Flywheel	110
Product Metrics	116
Case Study Introduction	117

Adoption Metrics	119
Value Metrics	121
Key Performance Indicators	123
Tactical and Strategic Metrics	124
Chapter Summary	125
Exercises	126
Answers	126

## Part III. Discover

<b>6. Understanding Your Target Audience.....</b>	<b>131</b>
Real Users, Not Straw Men	132
Case Study Introduction	134
Stand Back, Wield Science	134
Customer Discovery	135
Getting Interviews	135
Customer Discovery Interviews	136
Sales Calls	141
Customer Discovery Surveys	142
Networking with Interviewees	143
Customer Interviews Rules of Thumb	143
Crafting and Communicating a Target Audience	144
Choosing a Target Audience	144
Gaining Alignment with Personas	145
Audiences for the App Center	147
Nonpersonas	148
Selecting Features Based on a Target Audience	149
Keeping Your Focus	150
Multipersona Products	151
When Personas Are in Conflict	151
Understanding the Value of a Customer	153
Prioritizing Among Competing Personas	153
Chapter Summary	154
Exercises	154
Answers	155
<b>7. Discovering Your Product Through Simulation.....</b>	<b>157</b>
From Vision to Requirements	157
Case Study Introduction	160
AI Assistant Product Brief	161
Finish Your Product Vision with North Star Scenarios	162

Scenario-Driven Discovery	162
Brainstorm Scenarios as a Team	163
Select and Refine the North Star Scenarios	164
Convert the North Star Scenarios into Requirements	167
High-Level Requirements	168
The Product Requirements Document	169
Use Case Compendium	169
Organize Your Use Case Compendium	171
Prioritize the Requirements for Your First Milestone	173
What Really Matters for Prioritization?	173
Define the Target North Star Scenarios for Your First Milestone	178
Prioritize the Use Case Compendium	178
Build Detailed User Flows for Your First Milestone	181
Validating User Flows	184
Translate User Flows into Jobs to Be Done	185
Feeding Back into the Requirements	185
Chapter Summary	186
Exercises	187
Answers	188

---

## Part IV. Define

<b>8. Interaction Design</b>	<b>191</b>
The Role of Bias and Ideology in Software Design	193
Debate: Flexible Versus Opinionated	193
Debate: Optimism Versus Pessimism	194
Debate: Open Source Versus Proprietary	194
The Prerequisites of Good Design	195
Case Study Introduction	195
Call Attention to the Correct Usage of Your Product and Away from	
Incorrect Usage	197
Choose Safe, Predictable Defaults, or None at All	198
Optimize Your Path of Least Resistance	200
Give Affordances to the Right Persona in the Right Scenario	201
Perform Validations	203
Time the Affordances You Ship Wisely	204
If It's Worth Building, It's Worth Validating	205
Be Neither an Optimist nor a Pessimist	205
Apply the Rule of Three	206
Build It in Stages	207
If Necessary, Start with an Experimental Version	209

161	Narrow Versus Extensible Features	Scenario-Driven Discovery	209
162	Chapter Summary	Brainstorm Scenarios as a Team	211
164	Exercises	Select and Refine the North Star Scenarios	212
167	Answers	Convert the North Star Scenarios into Requirements	212
168	Chapter Summary	High-Level Requirements	
99	<b>9. Product Architecture.....</b>	The Product Requirements Document	<b>215</b>
169	The Foundations of Product Architecture	Use Case Compendium	216
171	Zoom Out and Look at the Big Picture	Organize Your Use Case Compendium	217
173	Avoid the Streetlight Effect	Prioritize the Requirements for Your First Mile	217
175	Communicate the User Impact of a Proposal	What Really Matters for Product Success	218
178	Use Technologies That Bridge the System-Product Gap	Define the Bridge	219
178	Case Study Introduction	Prioritize the Use Case Compendium	220
181	Reliable User Experiences	Build Detailed User Flows for Your First Milestone	220
184	Latency	Validating User Flows	221
185	Availability	Translate User Flows into Jobs to Be Done	223
187	Data Consistency	Feeding Back into the Requirements	224
186	Latency, Availability, and Data Consistency Trade-Offs	Chapter Summary	227
187	Scalability	Exercises	229
188	Scale Simulations	Answers	229
	Communicating Nonfunctional Requirements to Users		233
	Chapter Summary		235
	Exercises	Define	236
	Answers		236
191	Wrap-Up	Interaction Design	238
193	<b>Index.....</b>	The Role of Bias and Ideology in Software Design	<b>239</b>
194		Debate: Optimism Versus Pessimism	
194		Debate: Open Source Versus Proprietary	
195		The Prerequisites of Good Design	
195		Case Study Introduction	
		Call Attention to the Correct Usage of Your Product and Analyze the	
197		Incorrect Usage	
198		Choose Safe, Predictable Details as Opposed to Novelty	
200		Optimize Your Path of Least Resistance	
201		Give Affordances to the Right Persons in the Right Scenario	
203		Perform Validations	
204		Time the Affordances You Ship Wisely	
205		If It Ain't Broke, Don't Fix It: The Value of Inertia	
205		Be Neither an Optimist nor a Pessimist	
206		Apply the Rule of Three	
207		Build It in Stages	