Index

• Numerics •

12 Agile Principles, The
  collaboration, 28
  communication, 28, 243
  conversation, 28
  conveying information, 28
  cost management, 220
  customer satisfaction, 27–30
  delivering working software, 28
  focus, 21
  four groups, 28
  list of, 27–28
  measuring progress, 28
  motivation, 28
  origin, 20
  project management, 28
  quality, 28
  quality management, 251–252
  risk management, 264
  scope management, 188
  securing goods and services, 195–196
  self-organizing teams, 28
  simplicity, 28, 59
  sustainable development, 28, 33
  team dynamics, 227–228
  teamwork, 28
  time management, 207

efficiency, 57
emphasis on value, 59
to executives, 56–57
improving value, 58
increasing ROI opportunity, 57
to management, 58–59
product development, 57–58
agile artifacts
  for cost management, 226
  scope management roles, 195–197
  for time management, 219
agile benefits
  collaboration and ownership, 311–312
  customer satisfaction, 310
  performance visibility, 314
  product quality, 309–310
  project control, 314–315
  project predictability, 315
  reducing risk, 315–316
  relevance of metrics, 313
  team morale, 310–311
agile certifications
  CSD (Certified Scrum Developer), 75
  CSM (Certified Scrum Master), 75
  CSPO (Certified Scrum Project Owner), 75
  obtaining, 287–288
  PMI-ACP (Agile Certified Practitioner), 75
  Agile Certified Practitioner (PMI-ACP), 75
  agile champion, choosing, 284
agile communication. See also
  conversations; questions
  The Agile Manifesto, 244
  Agile Principles, 243
  channels, 246–248
  face-to-face, 259–260
  fidelity, 78
  high-tech, 84–85
  historical versus agile, 244
  importance of, 22, 24, 28, 102, 242–243
  low-tech, 82–84
  methods, 245–248
  in self-management, 232
  showing versus telling, 248
  status and progress reporting, 248–250

• A •
adaptation. See inspection and adaptation
affinity estimating, 137–138. See also
  estimates
  agile
cargo cult, 300
double work, 249, 300
faux, 300
teams in, 112
The Agile Alliance, 20, 328
agile appeal
to customers, 57–58
to development teams, 59
agile communication (continued)
  task boards, 249
  types of, 245
agile environment, establishing, 238–240, 287–288
agile frameworks
  adaptation, 64
  basis, 64
  described, 63
  empirical approach, 64
  flexibility and stability, 48–50
  inspection, 64
  iterations, 66
  lean approach, 67–69
  reducing nonproductive tasks, 51–52
  scrum, 70–74
  transparency, 64
  XP (extreme programming), 69–70
Agile Leadership Network, 329
agile litmus test, 41–42
The Agile Manifesto, 13
  collaboration, 195
  communication, 244
  contract negotiation, 25–26, 195
  customer collaboration, 25–26
  focus, 21
  following plans, 26–27
  importance of conversation, 22
  individuals and interactions, 21–23, 47
  inspection and adaptation, 73, 104
  origin, 20
  processes and tools, 21–23, 47
  responding to change, 26–27
  software versus documentation, 23–25
  text of, 20
  Value 1, 21–23
  Value 2, 23–25
  Value 3, 25–26
  Value 4, 26–27
agile mentor
  choosing, 284–285
  role of, 95
agile movement impact
  on attitude toward change, 39
  on business and IT groups, 39
  on knowledge workers, 39
  on processes, 39
The Agile Principles
  collaboration, 28
  communication, 28, 243
conversation, 28
conveying information, 28
cost management, 220
customer satisfaction, 27–30
delivering working software, 28
focus, 21
four groups, 28
list of, 27–28
measuring progress, 28
motivation, 28
origin, 20
project management, 28
quality, 28
quality management, 251–252
risk management, 264
scope management, 188
securing goods and services, 195–196
self-organizing teams, 28
simplicity, 28, 59
sustainable development, 28, 33
team dynamics, 227–228
teamwork, 28
time management, 207
agile processes, training in, 287–288
agile project management
  avoiding pitfalls, 299–303
  awareness, 290–291
  barely specific, 25
  benefits, 43–46, 292
  changes to come, 41
  dealing with deadlines, 44
  defined, 2, 9
  estimates, 295
  excitement, 290–291
  gathering feedback, 297
  gold-plating, 25
  history, 13–14
  implementation strategy, 289–290
  improving, 297
  inspection and adaptation, 289–290
  iterative cycle, 45–46
  making mistakes, 297
  maturing, 297–298
  moving to, 281–282
  pilot project, 291–293
  product backlog, 295
  product roadmap, 295
  product strategy, 295
  running sprints, 295–296
Satir’s Curve, 293
scaling virally, 299
success metrics, 293–294
training, 294
transformation team, 290
agile projects. See also pilot project; projects accountability, 230
adaptation, 15, 17, 73, 177–178
assigned deadlines, 210
benefits, 17
budget considerations, 210
cost of failure, 270
deciding risk model, 266
determining lengths of, 210
empirical control method, 15
functionality completed, 210
inspection, 15, 17, 73, 177–178
iterations, 15
minimizing scope creep, 17
self-funding, 223–224
starting, 49
success rates, 17, 264
transparency, 15
versus waterfalls, 15–16
agile resources. See also websites
The Agile Alliance, 328
Agile Leadership Network, 329
InfoQ online community, 329
Lean Essays, 330
online cheat sheet, 327
Platinum Edge, 330
PMI community, 328
The Scrum Alliance, 328
Scrum Development Yahoo! Group, 329
What Is Extreme Programming?, 330
agile roles
agile mentor, 95
development team, 89–90
establishing, 87–88
product owner, 90–92
scrum master, 92–93
in sprints, 159–161
stakeholders, 94–95
agile transition, supporting, 288
application development, money spent on, 11
artifacts
for cost management, 226
scope management roles, 195–197
for time management, 219
automated testing. See also testing
functional, 263
integration, 263
lack of, 301
performance, 264
performing, 163
regression, 263
role in quality management, 262–264
smoke, 264
static testing, 163, 264
system testing, 163
unit testing, 163, 263
user acceptance, 263

• B •
Beck, Kent, 69
behavior, increasing maturity level, 105
benefits of agile. See agile benefits
Bid it low and watch it grow, 201
Booch, Grady, 14
budget, creating, 222. See also
cost management
bugs, origin of terminology, 253
burndown charts. See also progress
in development sprint, 181
downloading, 156
examples, 156–157
Expected, 156–157, 250
Failing Fast, 157, 250
Less Complicated, 157, 250
Lying, 157, 250
More Complicated, 250
More complicated item, 156–157
Not Participating, 157, 250
profiles, 250
using, 83, 154–156, 249

• C •
certifications, 75
Certified Scrum Developer (CSD), 75
Certified Scrum Master (CSM), 75
Certified Scrum Project Owner (CSPO), 75
changes. See also scope changes
accommodating, 50
improving adaptation to, 57–58
managing, 49
responding to, 26–27
client feedback, collecting, 49
Cockburn, Alistair, 78, 245

collaboration
importance of, 102
websites, 85

collaboration and ownership, increasing, 311–312

colocation, impact on communication, 78–79

commitment
challenges, 280–281
concerns, 281
establishing, 96–97
getting, 279–280
individual, 279
making transition, 280–281
organizational, 278
roadblocks, 280
significance, 277

communication. See also conversations; questions
The Agile Manifesto, 244
Agile Principles, 243
channels, 246–248
face-to-face, 259–260
fidelity, 78
high-tech, 84–85
historical versus agile, 244
importance of, 22, 24, 28, 102, 242–243
low-tech, 82–84
methods, 245–248
in self-management, 232
showing versus telling, 248
status and progress reporting, 248–250
task boards, 249
types of, 245
consensus
deciding with, 103
gaining, 94

contract negotiation, 25–26
contracts for services
agile approaches, 201–202
closing, 205
creating, 200–202
description of work, 201–202
reviewing, 202
revising, 202
signing, 204

cost management
agile artifacts for, 226
creating budget, 222
as expression of project time, 221
historical versus agile, 220–221
increases, 221
self-funding projects, 223–224
velocity for long-range costs, 224–226
cost structures for services
fixed-price projects, 199–200
fixed-time projects, 200
not-to-exceed projects, 200
time-and-materials projects, 200
costs. See also project costs
Agile Principles, 220
defined, 220
determining, 226
lowering, 224–226
in self-management, 232
courage, establishing, 99–100
cross-functional teams, working with, 237–238
cross-functionality, benefits, 100–101
CSD (Certified Scrum Developer), 75
CSM (Certified Scrum Master), 75
CSPO (Certified Scrum Project Owner), 75
Cunningham, Ward, 69
customer collaboration, 25–26
valuing, 195
customer representative. See product owner
customer satisfaction, 27–30, 310
enacting principles, 29
strategies, 30
customers
appeal of agile to, 57–58
defining, 29
dissatisfaction of, 30
identifying in user stories, 130–131
participation, 58
preparing for product release, 184
daily scrum. See also scrum
  in development sprint, 181
duration, 152–153
effectiveness, 153
enabling team coordination, 152
guidelines, 152–153
holding, 111
planning, 151–154
purpose, 153–154
risk management, 273
in Roadmap to Value, 192
scheduling, 153
deadlines
  assigning, 210
  dealing with, 44
determining high-level, 126
decomposition, 127, 133–134
dedicated teams, working with, 235–237.
  See also scrum teams; team dynamics
definitions of done, 267
delivery, speeding up, 53
deployment, defined, 2. See also
development
  Derby, Esther, 176
  design, defined, 2
desktop sharing, 85
development
  defined, 2
  sustainability of, 28, 33, 162–163
development sprint. See also sprints
  burndown chart, 181
daily activities, 181
daily scrum, 181
daily scrum, 181
end-of-day reporting, 182
product backlog, 181
versus release sprint, 180–182
sprint backlog, 181
sprint planning, 181
sprint retrospective, 182
sprint review, 182
development teams. See also teamwork
  appeal of agile to, 59
  choosing, 282–283
  focus of, 54
focusing in daily scrum, 153
ideal size of, 216
limiting size of, 240
members of, 89–90
nurturing, 47
role in quality management, 259
role in scrum framework, 73
tasks during sprints, 159–161
working with product owner, 49
distractions. See also focus
  management, 81
  multiple projects, 80
  multitasking, 80
  outside influences, 81
  over-supervising, 81
  removing, 80–81, 214
documentation
  evaluating, 24–25
  five whys, 25
  minimizing, 52
  technical versus user, 180
done, definitions of, 267
double work agile, 249, 300
efficiency, realizing, 57
elaboration, process of, 162
e-mail, nonproductivity of, 52
empirical control method, 15
environment. See physical environment
estimates
  defined, 2
developing, 295
  refining, 134–136
estimating by affinity, 137–138
estimation poker, playing, 134–136
EVM (Earned Value Management), 249
executives, appeal of agile to, 56–57
extreme programming (XP)
  focus of, 69
  versus lean approach, 75
  practices, 70–71
  principles, 70
  roles, 69
  versus scrum, 75
• **F** •

failing fast, concept of, 271
failure of projects
costs of, 269–270
detecting faster, 55–56, 270–271
features
arranging, 121–122
estimating, 123–125
grouping by themes, 122
ordering, 123–125
in product roadmap, 120
providing information about, 134
usage statistic, 112
feedback
cycle of, 171–172
gathering, 297
Fibonacci sizing sequence
in affinity estimating, 137–138
using, 124, 133
the fist of five, 94
focus, establishing, 97–98, See also distractions
functional testing, 263
funding and spending cycles, 41

• **G–H** •
gold-plating, 25, 173
goods and services. See procurement management

• **I** •

index cards, writing user stories on, 128, 132
individuals and interactions, 21–23
InfoQ online community, 329
information, conveying, 28
information radiator, using, 83
innovation, impact of stability on, 50
inspection and adaptation, 15, 17
benefits, 297
planning, 113
in quality management, 261–262
in self-management, 104
in sprint retrospective, 177–178
using daily, 73
instant messaging, 85
integration, defined, 2
integration testing, 263
INVEST approach
  Estimable, 135
  Independent, 135
  Negotiable, 135
  Small, 135
  Testable, 135
  Valuable, 135
iterations. See also development sprint;
release sprint
agile roles in, 159–161
components, 142–143
concentrating on, 49
cyclical nature, 72
defined, 142
determining status of, 156
diagram, 72
duration, 142, 148–149
inspection and adaptation, 73
length, 142
planning, 49, 111
priorities in, 49
in projects, 50
as recurring processes, 72
running, 295–296
starting, 50
tasks of product owner, 160
testing within, 254–256
tracking progress, 248
in waterfalls, 66
working in, 15, 45

• **J** •

Jeffries, Ron, 69, 330
JIT (just-in-time) planning, 58, 109
  benefits, 68
  inspect and adapt, 113
  kanban boards, 67–68
Jones, Daniel T., 68

• **K** •

kanban board
  role in controlling production, 67–68
  versus task board, 159
  using, 83–84
## Index

### L

- Larsen, Diana, 176

### M

- lean approach
  - coining terminology, 68
  - versus extreme programming, 75
  - origins in manufacturing, 67
  - principles, 68–69
  - versus scrum, 75
  - simplicity, 1, 67

### N

- Nonaka, Ikujiro, 13

### O

- online collaboration tools, using, 242
- openness, establishing, 98
- organization clout, importance of, 165
- organizational commitment, 278

### P

- pair programming, role in quality management, 258
- participation, importance of, 103–104
- PDSA (Plan-Do-Study-Act) approach, 13–14
- peer review
  - conducting, 163–164
  - role in quality management, 258
- performance testing, 264
- performance visibility, improving, 314
- personas, including in user stories, 131
- philosophy. See team philosophy
- physical environment
  - burndown charts, 83
  - choosing tools, 85–86
  - collaboration websites, 85
  - collocating teams, 78–79
  - conversations, 82
  - desktop sharing, 85
  - inappropriateness, 302
  - information radiator, 83
  - instant messaging, 85
  - kanban board, 83–84
  - movability in, 81–82
  - project room, 79–80
  - removing distractions, 80–81
  - representing units of work, 83
- scrum room, 79–80
- setting up dedicated area, 79–80
- sticky notes, 83
- video conferencing, 85
- webcams, 85
- white board, 83–84
- pilot project, identifying, 291–293. See also agile projects; projects
pitfalls. See also roadblocks
avoiding, 299–303
cargo cult agile, 300
diluting until dead, 303
discipline slips, 302
double work agile, 300
faux agile, 300
lack of automated testing, 301
lack of support for learning, 303
lack of training, 300
lack of transition support, 301
physical environment, 302
poor team selection, 302
product owner, 301
Plan-Do-Study-Act (PDSA) approach, 13–14
planning. See also release plan; Roadmap to Value; sprint planning
in agile, 110
iterations, 111
just-in-time, 109
necessity of, 112
Roadmap to Value, 110
sprints, 111
Platinum Edge websites, 330
Platinum Principles
resisting formality, 37–38
teamwork, 38
visualizing versus writing, 38–39
PMI (Project Management Institute), 75, 328
PMI community website, 328
PMI-ACP (Agile Certified Practitioner), 75
poker cards, estimation, 135
Poppendieck, Mary and Tom, 68, 330
presentations, preparing for, 52
priority, calculating relative, 124–125
problems, preventing via questions, 303–305
process documentation. See documentation
process waste, minimizing, 58–59
processes and tools, 21–23
procurement
defined, 195
in self-management, 231
procurement management, 197. See also scope management
companies, 203
contracts for services, 200–202
costs of services, 199–202
determining needs, 198–199
expressing benefits, 203
goods and services, 199
organizational considerations, 202–204
organizations, 203
selecting vendors, 198–199
product backlog
adding requirements to, 194
completing, 139–141
considering for scope changes, 194
defined, 140
developing, 295
in development sprint, 181
estimate, 140
example, 140
keeping up to date, 141, 194
making up, 121
reviewing, 249
risk management, 272
story points, 140
for time management, 219
product deployment. See also deployment
customer testing, 184
legal considerations, 183
manufacturing, 183
marketing campaigns, 182
marketing support, 184
preparing marketplace for, 183–184
preparing organization for, 182–183
sales, 182
support, 183
support channels, 184
product development, appeal of agile in, 57–58
product features
arranging, 121–122
estimating, 123–125
grouping by themes, 122
ordering, 123–125
in product roadmap, 120
providing information about, 134
usage statistic, 112
product objective, developing, 114–115
product owner
characteristics, 91
ineffectiveness, 301
management of user stories, 129
responsibilities, 91, 114
role in quality management, 259
role in scrum framework, 73
role in sprint review meetings, 174
role of, 29, 47, 49, 90–92
tasks during sprints, 160
working with development team, 49
product owner review, conducting, 164
product quality, improving, 309–310. See also quality management
product requirements. See also user stories
breaking down, 133–134
decomposing, 133–134
defined, 2
determining, 131–132
estimating, 123–124
prioritizing, 125
refining, 133–134
scoring, 123–124
product roadmap. See also Roadmap to Value
arranging product features, 121–122
backlog, 121
calculating relative priority, 124–125
categories on white board, 122
creating, 111, 118–119, 295
decomposing requirements, 120
epic user stories, 120
features, 120
Fibonacci sizing sequence, 124
high-level time frames, 126
identifying requirements, 119–121
prioritizing requirements, 125
requirement value and effort, 123–124
requirements on white board, 122
risk management, 272
in Roadmap to Value, 190
saving, 126
versus task boards, 158
tasks, 120
themes, 120
for time management, 219
user stories, 120
product schedule document, evaluating, 24
product scope, defined, 187
product strategy, developing, 295
product technical specifications,
evaluating, 24
product vision. See also vision statement
defining, 113–114
drafting statement, 115–117
identifying, 111
risk management, 272
statement, 113–114
product waste, minimizing, 58–59
productive work, maximizing, 51–52
productivity, increasing, 97
products
evaluating, 23
focusing on, 23
progress. See also burndown charts; sprint backlog; task progress
measuring, 28
reporting on, 248–250
tracking, 154–159
project communication plan, evaluating, 24
project control
improving, 55
increasing, 314–315
project costs, relationship to duration, 320. See also costs
project facilitator. See scrum master
project failure, detecting faster, 55–56
project management, 28
advantages, 34–35
defined, 1
facilitating, 36–37
historical versus agile, 44–45
origins, 10–11
principles of, 34–37
role of, 34
wasteful practices, 12–13
Project Management Institute (PMI), 75, 328
project owner, considering, 283–284
project predictability, improving, 315
project scope, defined, 187
project stakeholders
choosing, 285
identifying, 130
participation, 58
role in writing user stories, 132
role of, 94–95
project status, reporting, 39–40, 248–250
project teams. See also scrum
teams; teamwork
improving, 54–55
members of, 88, 93
product owner, 29, 47
scrum master, 47
supporting, 54
project timelines, estimating, 211–213
projects. See also agile projects;
pilot project
defined, 9
failure rate, 11
finishing, 51
managing with dislocated teams, 241–243
statistics, 11
success rate, 11
tracking progress of, 248

• Q •

QA (quality assurance), 254
quality
raising, 53
in self-management, 233
quality management, 28. See also
product quality
acceptance criteria, 259–260
Agile Principles, 251–252
automated testing, 262–264
collective code ownership, 258
continuous integration, 258
development team, 259
development techniques, 257–259
face-to-face communication, 259–260
feedback, 253
good design, 256–257
historical versus agile, 252–253
inspection and adaptation, 261–262
pair programming, 258
peer reviews, 258
principles of, 31–32
proactive, 256–261
product owner, 259
QA (quality assurance), 254
sprint retrospective, 262
sprint reviews, 261–262
strategies, 32
sustainable development, 260–261
TDD (test-driven development), 258
technical excellence, 256–257
testing within sprints, 254–256
user stories, 259–260
questions, asking to prevent
problems, 303–305. See also
communication; conversations

• R •

regression testing, 263
relative priority, calculating, 124–125
release, defined, 138
release date, determining, 141
release goal, establishing, 141
release plan. See also planning
creating, 111, 141–142
example, 142
product backlog, 139–141
risk management, 272
in Roadmap to Value, 139, 190
for time management, 219
release sprint. See also sprints
backlog, 182–183
contents, 180
versus development sprint, 180–182
requirement levels
Epic User Stories, 134
Features, 133
Theme, 133
User Stories, 134
requirements. See also user stories
breaking down, 133–134
decomposing, 133–134
determining, 131–132
defined, 2
estimating, 123–124
prioritizing, 125
refining, 133–134
scoring, 123–124
requirements document, evaluating, 24
resources. See also websites
The Agile Alliance, 328
Agile Leadership Network, 329
InfoQ online community, 329
Lean Essays, 330
online cheat sheet, 327
Platinum Edge, 330
PMI community, 328
The Scrum Alliance, 328
Scrum Development Yahoo! Group, 329
What Is Extreme Programming?, 330
respect, establishing, 98–99
retrospectives. See sprint retrospective
return on investment (ROI), defined, 115
risk management
Agile Principles, 264
daily scrum, 273
definition of done, 267
failing fast projects, 269–271
historical versus agile, 265
product backlog, 272
product roadmap, 272
product vision, 272
release plan, 272
ROI models, 268
self-funding projects, 268–269
sprint backlog, 272
sprint planning, 272
sprint retrospective, 272
sprint review, 273
task board, 273
risks
defined, 264
identifying, 271–273
prioritizing, 271–273
reducing, 315–316
responding to, 271–273
in self-management, 233
roadblocks. See also pitfalls
avoiding, 213
daily scrum, 111, 151, 192
examples, 165
identifying, 164–166
organizational impediments, 165
preventing, 215
self-funding projects, 268–269
task board, 273
seven stages, 111, 190–192
sprint planning, 192
sprint retrospective, 111, 192
sprint review, 111, 169–170, 192
vision, 190
ROI (return on investment)
agile project, 322–323
defined, 115
models, 268
traditional project, 321–322
ROI opportunity, increasing, 57
roles
agile mentor, 95
development team, 89–90
establishing, 87–88
product owner, 90–92
scrum master, 92–93
in sprints, 159–161
stakeholders, 94–95
Roos, Daniel, 68
Royce, Winston, 10, 14, 64, 66

S
Satir’s Curve, 293
Schwaber, Ken, 14, 217
scope
defined, 2
in self-management, 231
scope bloat, explained, 12
scope changes. See also changes
assessing requirements, 193
estimating effort, 193
introducing, 192–193
managing, 193–195
managing from time perspective, 215–216
prioritizing requirements, 194
product backlog, 194
scope creep, minimizing, 17, 188
scope management. See also
procurement management
agile artifacts for, 195
The Agile Principles related to, 188
development teams, 190
historical versus agile, 189
product backlog, 196
product owners, 190
product roadmap, 196
release plan, 196
sprint backlog, 196
scope management (continued)
  stages of Roadmap to Value, 190–192
  vision statement, 196
scrum, defined, 88. See also daily scrum
Scrum Alliance, 14
The Scrum Alliance, 328
Scrum Development Yahoo! Group, 329
scrum framework, 46
  artifacts, 73–74
  development team, 73
  diagram, 72
  events, 74
  versus extreme programming, 75
  versus lean approach, 75
  meetings, 74
  product backlog, 74
  product increment, 74
  product owner, 73
  roles, 73
  scrum master, 73
  sprint backlog, 74
  sprints, 70, 72–73
  working with agile mentors, 73
  working with stakeholders, 73
scrum master
  characteristics, 93
  choosing, 283
  communication skills, 93
  negotiation skills, 93
  organization clout, 165
  role in estimation, 134
  role of, 47, 92–93
  role scrum framework, 73
  as servant-leader, 234–236
  tasks during sprints, 160–161
scrum teams. See also dedicated
teams; multiple scrum teams;
project teams; teams
  collocated versus dislocated, 241–242
  collocating, 78–79
  communication, 242–243
  defined, 88
  developing focus, 97
  distributing, 241
  members of, 88, 90
  tasks during sprints, 159–161
self-funding projects, risk management, 268–269
self-management
  benefits, 103–104
  communication, 232
  cost, 232
  procurement, 231
  quality, 233
  result of, 312
  risk, 233
  scope, 231
  team dynamics, 232
  time, 231
self-organization, benefits, 102–103
servant-leader
  characteristics, 234–235
  scrum master as, 234–236
Sherwart, Walter, 13–14
shippable functionality
  activities, 161
  creating, 161–166
  defined, 161
  developing, 162–163
  elaboration, 162
  relating to user stories, 161
  verifying, 163–164
Shu Ha Ri technique, 298
size-limited teams, benefits, 104–105
smoke testing, 264
software features, use of, 12
software projects. See also
  working software
  definition of done, 23
  failure, 11
  success, 11
software versus documentation, 23–25
Spears, Larry, 234
Spira, Jonathan, 97
“spiral development method,” 14
sprint backlog. See also progress
  burndown chart, 154–156
  contents, 249
  creating, 143–144
  creating tasks for, 147–149
  creating with spreadsheet, 156
  in development sprint, 181
  evaluating, 148
  example, 145
  items in release sprint, 180
preparing for product deployment, 182–183
risk management, 272
for time management, 219
tracking progress, 154–156
updating, 167, 249
sprint goals, committing to, 102
sprint planning. See also planning
risk management, 272
in Roadmap to Value, 192
sprint planning meeting, 146–149
choosing user stories, 146–147
duration, 144, 146
enabling swarming, 147
listing sprints, 148
setting goals, 146–147
timeboxing, 144
sprint retrospective
agenda, 176–177
in development sprint, 182
discussion items, 176
duration of meeting, 175–176
goal, 174
holding, 111, 174–175
inspection and adaptation, 177–178
meeting, 175–177
planning for, 175
purpose, 262
questions covered, 176
results, 174
risk management, 273
in Roadmap to Value, 192
sprint review meeting
collecting feedback, 173–174
cycle of feedback, 171–172
gold-plating, 173
guidelines, 172
importance of, 173
role of product owner, 174
to sprint length ratio, 172
user stories, 174
sprint reviews
defined, 169, 248
in development sprint, 182
holding, 111, 169–174
indicating status of done, 170–171
inviting others to, 249
preparing to demonstrate, 170–171
purpose, 261–262
risk management, 273
in Roadmap to Value, 192
sprints. See also development sprint;
release sprint
agile roles in, 159–161
components, 142–143
concentrating on, 49
cyclical nature, 72
defined, 142
determining status of, 156
diagram, 72
duration, 142, 148–149
inspection and adaptation, 73
length, 142
planning, 49, 111
priorities in, 49
in projects, 50
as recurring processes, 72
running, 295–296
starting, 50
tasks of product owner, 160
testing within, 254–256
tracking progress, 248
in waterfalls, 66
working in, 15, 45
stakeholders
choosing, 285
identifying, 130
participation, 58
role in writing user stories, 132
role of, 94–95
Standish Group study, 11
static testing, performing, 163, 264
status, reporting on, 248–250
status quo, problem with, 11–13
status report, evaluating, 24
sticky notes, using, 83
story point, defined, 133. See also
user stories
strategy, developing, 295
success metrics, identifying, 293–294
sustainable development, role in quality
management, 260–261
Sutherland, Jeff, 14, 241
swarming, enabling, 147
system testing, performing, 163
• T •

Takeuchi, Hirotaka, 13
task boards
Accept column, 158
To Do column, 158
Done column, 158
example, 158–159
versus kanban board, 159
versus product roadmap, 158
In Progress column, 158
risk management, 273
using, 158–159, 249
task progress, reporting on, 167. See also progress
tasks
creating for sprint backlog, 147–149
duration, 144
identifying, 102
in product roadmap, 120
task-switching, result of, 235–236
TDD (test-driven development), 258
team agreement, creating, 104
team coordination, enabling, 152
team dynamics, 235–236. See also dedicated teams
Agile Principles, 227–228
cross-functional teams, 237–238
historical versus agile, 229
in self-management, 232
self-managing, 229–233
self-organizing, 229–233
team members
agile champion, 284
agile mentor, 284–285
characteristics, 89
choosing, 282–285
development team, 282–283
project owner, 283–284
project stakeholders, 285
Roadmap to Value, 286
scrum master, 283
turnover, 326
team morale, raising, 310–311
team performance, improving, 53–55
team philosophy
changing, 100–105
cross-functionality, 100–101
mature behavior, 105
self-management, 103–104
self-organization, 102–103
size-limited teams, 104–105
team structures, customizing, 312
teams. See also scrum teams
in agile, 112
dislocated, 241–243
selecting, 302
self-organizing, 28
size-limited, 104–105
teamwork, 28. See also development teams;
project teams
Platinum Principle, 38
practices, 33
principles of, 32–34
strategies, 34
sustainable development, 33
technical documentation, 180
test-driven development (TDD), 258
testing. See also automated testing
defined, 2
preparing for, 53
within sprints, 254–256
time, considering in self-management, 231
time frames, determining high-level, 126
time management
agile artifacts for, 219
Agile Principles, 207
benefit, 208
determining project length, 210
historical versus agile, 208
product backlog, 219
product roadmap, 219
release plan, 219
sprint backlog, 219
strategic schedules, 209
tactical schedules, 209
using multiple scrum teams, 216–218
velocity, 209–210
Index

345

time to market, 319–320
time zones, considering, 242
Toyota, origin of, 67
training, lack of, 300
training in agile methods, 294
transition support, lack of, 301
Twelve Agile Principles. See also 12 Agile Principles, The

• U •

unit testing, performing, 163, 263
units of work, representing, 83
user acceptance testing, 263
user documentation, 180
user stories. See also product requirements; story point
benefit, 129
creating, 129–132
customers, 130–131
decomposing into tasks, 147–148
defined, 210
ensuring quality in, 135
estimating, 134–136
example, 128–129
gathering and managing, 129
INVEST approach, 135
parts of, 128
personas, 131
product requirements, 131–132
project stakeholders, 130
relating to shippable functionality, 161
role in quality management, 259
using index cards, 128, 132
using low-tech tools, 128
validation steps, 128
verification, 164

• V •

values. See also Roadmap to Value
commitment, 96–97
courage, 99–100
establishing, 95–100
focus, 97–98
openness, 98
respect, 98–99
velocity
adjusting, 210–215
benefits, 210
calculating, 211
consistency, 214–215
defined, 313
determining long-range costs, 224–226
estimating project timelines, 211–213
guidelines, 212
increasing, 213–214
measuring, 209
monitoring, 210–215
role in time management, 209–210
as trending tool, 211
using, 212–213
variations, 213
vendors
avoiding low-balling, 201
defining product scope, 201
selecting, 198–199
working with, 204–205
verifying
automated testing, 163
peer review, 163–164
product owner review, 164
videoconferencing, 85, 242
vision in Roadmap to Value, 190
vision statement. See also product vision
drafting, 115–117
finalizing, 118
revising, 117–118
validating, 117–118
visualizing versus writing, 38–39
von Moltke, Helmuth, 109

• W •

Wake, Bill, 135
waterfall
versus agile project, 15–16
coining, 10
defined, 1
waterfall approach
versus agile methodologies, 56
iterations, 66
origins, 64–65
shortfalls of, 48
waterfall projects
cost of failure, 269–270
failure detection, 55–56
linear methodology, 45
testing phase, 45
webcams, 85
websites. See also agile resources
burndown chart, 156
CSD (Certified Scrum Developer), 75
CSM (Certified Scrum Master), 75
CSPO (Certified Scrum Project Owner), 75
customer dissatisfaction, 30
documentation form, 25
estimation poker cards, 135
marshmallow challenge, 44
PMI-ACP (Agile Certified Practitioner), 75
project management table, 36
sprint backlog template, 156
XP123.com, 135
What Is Extreme Programming?, 330
white board
categories on, 122
requirements on white board, 122
using, 83–84
Womack, James P., 68
work speed. See velocity
working software, delivering, 28. See also
software projects
Wujec, Tom, 44

• X–Y–Z •

XP (extreme programming)
focus of, 69
versus lean approach, 75
practices, 70–71
principles, 70
roles, 69
versus scrum, 75
XP practices
coding standards, 71
collective code ownership, 71
continuous integration, 71
design improvement, 71
pair programming, 71
planning game, 71
refactoring, 71
simple design, 71
small releases, 71
sustainable pace, 71
system metaphor, 71
TTD (test-driven development), 71
whole team, 71
XP123.com, 135