Index

A
Abbreviations, 237–238, 465–466
Abstraction
  in fit criteria, 405
  patterns from, 313–317
  for requirements, 2, 274, 307
  in trawling, 102, 386
Acceptance, usability for, 179–180
Accepted requirements, 437
Access requirements, 491
Accessibility requirements
  fit criteria for, 213
  in usability, 182, 481
Accuracy
  patterns for, 308
  requirements for, 183, 484
Achievable goals, 57–58
Acronyms, 237–238, 465–466
Actions in functional requirements, 217
Active adjacent systems, 80–82
Active stakeholders, 140, 149
Activity diagrams
  for functional requirements, 168
  for scenarios, 143–144
Actors
  defined, 437
  in event-driven use cases, 69–70, 89–90
  in operational requirements, 185
  in product use cases, 152
Adaptability requirements, 490–491
Adapting Volere process, 345–347
Adjacent systems
  active, 80–82
  autonomous, 83–85
  cooperative, 85–86
  in domains of interest, 43
  in event-driven use cases, 72, 79–86
  in function point counting, 519
  interfacing with, 487–488
  legal requirements for, 193
  in operational requirements, 185, 487–488
  and scope, 41
  as stakeholders, 54
  technology in, 131–132
  in trawling, 393
Adjectives, 339
Adjustments in function point counting, 520–521
Adobe Photoshop usability, 180
Adoption, usability for, 181
Adverbs, 339
Agile Manifesto, 4
Agile software development, 4–8
Agility guide
  for blastoff, 38
  for event-driven use cases, 67
  for fit criteria, 203–204
  for functional requirements, 155–156
  for nonfunctional requirements, 172
  for process, 19–20
  for prototyping, 285–286
  for Quality Gateway, 280–281
  for reviewing specifications, 322–323
  for scenarios, 135–136
  for trawling, 93–94
  for writing requirements, 223–225
Air traffic control systems, 116
Airlines
  cargo, 184–185
  check-in agent scenario, 137–144
  flight booking, 112
Alexander, Christopher, 204, 308
Alexander, Ian, 46, 147
Allowable values, requirements for, 183
Alternatives
  blastoff, 65
  functional requirements, 161–162, 167–169
  Quality Gateways, 280–281
  scenarios for, 144–145
Ambiguity
  in functional requirements, 162–164
  reviews for, 339–340
Analysis of prototyping results, 300
Analysts
  apprenticeships for, 102–103
Analysts (cont.)
for scope, 22
for trawling, 94–96
writing by, 26
“And no more” requirements, 189–190
Anonymity for retrospectives, 361
Antagonists in negative scenarios, 147
Anticipated environments
constraints from, 236, 463–464
for new products, 499–500
Appearance requirements, 476
Apprentices, 24, 101–103, 385
ART-SCENE scenario presenter, 146
Artifacts
in apprentices, 103
for deliverables, 346
in domain patterns, 316–317
in functional requirements, 157
in prototyping, 288, 293, 299
retaining, 128
reusing, 319
for stakeholder interviews, 105
Asimov, Isaac, 183
Assembling specifications, 437
Associations, 348–349
Assumptions
in blastoff, 37
defined, 437
relevancy of, 270
in reusing requirements, 305
in risk analysis, 341
in specification template, 239–240, 467–468
of usability, 178
Asynchronous networks, 366
Atomic requirements
in functional requirements, 166
prioritizing, 334
in writing requirements, 243–248
Attributes
in business use cases, 329, 516
in classes, 510–511
in stored data, 518, 520
for user categories, 50–51
Audiences, personas for, 121
Auditing requirements, 189, 197, 493
Authorization, 188
Automated tools
free, 243
for Quality Gateway, 280
for scenarios, 146
Autonomous adjacent systems, 83–85
Availability
assumptions about, 239
requirements for, 183, 188, 484–485
B
Background in specification template, 229–230,
454–455
Bad changes, 358
Baker, Jenny, 306–307
Beck, Kent, 203
Beede, Earl, 200
Benchmarks, 63
Beyer, Hugh, 136, 283, 290
Blame, avoiding, 360
Blastoff, 22–23, 35–37
agility guide for, 38
alternatives in, 65
business terminology in, 377
in case study, 38–39
communication in, 372
constraints in, 37, 60–61, 378
context partitioning in, 377
costs in, 23, 62–63
defined, 438
diagrams for, 369–370, 373, 379
domains of interest in, 378
first-cut risk analysis in, 375–376
follow-up, 381
go/no go decisions in, 23, 37, 64
goals in, 55–60
initial estimates in, 382
naming conventions and definitions in, 37, 61–62
non-events in, 377
objectives, 371, 438
physical arrangements in, 371–372
purpose determination in, 35, 374
reports for, 380
results review, 380–381
risks in, 37, 63–64
scope in, 35, 40–45
stakeholders in, 35–36, 45–55, 376–377
for trawling, 97
work context in, 374
Blogs
as free tools, 243
for nonfunctional requirements, 195
for trawling, 125–126
Boehm, Barry, 7, 508
Boundaries
in scope, 241, 472
use case, 393
Brainstorming
  in trawling, 117–119, 386–387
  videos for, 125
Brand loyalty, 111
Branding standards
  company colors for, 211
  in look and feel requirements, 176
Branson, Richard, 363
Brooks, Fred, 1
Buddy pairing approach, 281
Budgets
  as constraints, 237, 465
  person responsible for, 233
  requirements creep from, 277
Build stage in prototyping, 298–299
Building specifications, 223
Business events
  benefits of, 75–76
  for cost estimates, 62
  defined, 439
  effort for, 434–435, 440
  in event-driven use cases, 73–78
  finding, 76–78, 390–391
  identifying, 326–328
  origin of, 131
  patterns in, 309–313
  in product use cases, 150
  for prototyping, 440
  in scenario templates, 148
  time-triggered, 74
  tracing, 353–356
  in trawling, 390–391, 395
  video, 125
  in work partitioning, 240, 471
  in writing requirements, 244
Business opportunities
  change for, 357
  defined, 439
  in purpose, 230
  in trawling, 393
Business relevancy, 226
Business requirements, 11
Business rules
  in business use case workshops, 115–116
  maintainability requirements for, 186
Business tolerances
  for fit criteria, 207, 271
  in subjective tests, 210
Business use cases, 73, 86–88
  actors in, 89–90
  benefits of, 75–76
  CRUD check for, 330–331
  custodial processes in, 331
data for, 328–330
  events in, 78–79, 326–328
  in function point counting, 511–516
  for innovation, 111
  input, 512–513
  iteration in, 331–332
  low-fidelity prototypes for, 289
  modeling, 328
  output, 514–515
  for product determination, 129
  and scenarios, 136, 139, 148
  scope in, 326
  specification reviews for, 325–332
  in stakeholder interviews, 105
  time-triggered, 515–516
  traceability of, 267, 353–356
  in trawling, 24, 96–98
  workshops, 113–114
    business rules in, 115–116
    outcomes from, 114
    scenarios in, 113–115
    videos for, 125

C
Capabilities
  assumptions about, 239
  document archeology for, 128
  templates for, 11
  of tools, 350
Capacity requirements, 182, 485–486
Cargo airlines, 184–185
Change, 357–358
  in agile software development, 6
  defined, 439
  feedback for, 358–360
  open issues from, 253
  for opportunities, 357
  prototypes for, 414
  in purpose, 230
  requirements creep from, 277
  in trawling, 393
  in world, 358
Character of products, 173
Check-in agents, airline, 137–144
Checklists
  for blastoff decisions, 381
  for exceptions, 146
  for Quality Gateways, 281
  for requirement types, 176
  for risk, 340, 431, 448
  for specification reviews, 323–324
  for specifications, 28
  for stakeholders, 51, 376, 523
Checklists (cont.)
   templates as, 12, 172, 197
   for users, 50
Cheston, G. K., 134
Choices
   for innovation, 112
   scenarios for, 144–145
Clarification questions, 396–397
Class models, 348–349
   for functional requirements, 168–169
   for specification reviews, 342
Classes
   attributes in, 510–511
   for business use cases, 329–330
   in object life history, 296
   for reuse, 318
   in trawling, 104
Clausing, Don, 338
Clients
   names of, 439
   in nonfunctional requirements, 198–199
   in reusing requirements, 304
   in risk analysis, 341
   in specification template, 232–233, 456
   as stakeholders, 47–48
Collaborating systems and applications
   constraints from, 236, 462
   fit criteria for, 214
   in operational requirements, 185
Collaboration in agile software development, 6
Collections of requirements, 308–309
Color
   in branding, 211
   measuring, 206
   in mind maps, 123
Comments
   group, 441
   individual, 442
   project participant, 44
   for retrospectives, 361
Commercial off-the-shelf software
   as constraint, 61, 236, 462–463
   functional requirements for, 167
   in specification template, 253, 497–498
Communication
   in autonomous adjacent systems, 84
   in blastoff, 372
   for deliverables, 346
   prototypes for, 284
Company colors, 211
Comparisons
   Internet for, 111
in linguistic meta-models, 106
Completeness
   patterns for, 308
   reviewing, 406
   testing, 263–265
Compliance requirements, 495–496
Composite requirements, 400
Confidentiality requirements, 187–188
Conflicts
   identifying, 403
   specification reviews for, 337–339
   in writing requirements, 247–248
Connectedness, desire for, 112
Consistency
   objects for, 318
   in terminology, 267–268
Constraints, 10, 234
   assumptions in, 239–240, 467–468
   in blastoff, 37, 60–61, 378
   from environment, 236, 463–464
   facts in, 238, 467
   fit criteria for, 219–220
   formalizing, 400
   mandated, 234–237, 304, 341, 460–465
   naming conventions and definitions,
      237–238, 465–466
   off-the-shelf products, 61, 236, 462–463
   relevancy of, 270
   in reusing requirements, 304
   in risk analysis, 341
   in scenarios, 140
   in specification template, 234–237, 460–465
   in templates, 13
   traceability of, 267
   viability within, 272–273
Constructing products, 8
Consultants
   for security, 190
   as stakeholders, 52
Containing businesses, 46
Content
   model for, 365
   in retrospective reports, 362–363
Content management systems, 293
Context
   in event-driven use cases, 70–72
   of event responses, 310–311
   partitioning, 377
   in patterns, 313
   process in, 20–21
   in scope, 44–45, 240, 469–470
   in stakeholder interviews, 105

D

Design
  constraints in, 235
  and prototyping, 293, 298–299
Design form, reviewing, 437
Detail
  in functional requirements, 160–161, 251
  in management summaries, 351
Development phases, planning, 501
Deviations, exception cases for, 145–146
Diagrams
  blastoff, 369–370, 373, 377, 379
  for business events, 76–78
  context, 76–78, 377, 509–510
  for functional requirements, 168
  mind maps, 122–124
  prototyping, 407, 409, 412
  Quality Gateway, 404–406
  retrospectives, 415–416, 419, 422
  reviewing requirements, 425–426, 430, 433, 436
  for scenarios, 142–144
  state transition, 296
  summary, 368
  trawling, 97, 383–384, 392, 394
  writing requirements, 398
Dictionaries
  in specification template, 237–238, 465–466
  in Volere Requirements Process Model, 437–449
Discipline representation in brainstorming, 118, 386
Discovery, prototypes for, 293
Discussion forums for trawling, 125–126
Documentation
  in agile software development, 5
  in specification template, 256, 504–505
Documents, 438
Domains, 307
  models, 307, 440
  patterns across, 315–317
  patterns for, 314–315
  in reusing requirements, 317–318
Domains of interest
  in blastoff, 378
  in scope, 42–44
Drivers
  in risk analysis, 340–341
  in specification template
    clients, customers, and stakeholders, 232–233, 456–457
    purpose, 229–232, 454–456
users, 233–234, 457–460
  in templates, 13
Drupal system, 293
Dynamic adjacent systems, 80–82
E
Ease of use requirements, 180, 477–479
Easy to learn products, 180, 212
Effects of Quality Gateway, 260–262
Efficiency
  requirements for, 183
  usability for, 179
Effort
  estimating, 62–63
  for events, 434–435, 440
Einstein, Albert, 265
Elasticity of terms, 106
Elephant projects
  in agile software development, 7
  blastoff for, 38
  event-driven use cases for, 67
  fit criteria for, 204
  functional requirements for, 156
  nonfunctional requirements for, 172
  process, 20
  prototyping for, 286
  Quality Gateway for, 261
  scenarios for, 136
  specification reviews for, 323
  trawling for, 94
  writing requirements for, 225
End users as stakeholders, 48
Engineers for prototypes, 291
Entities
  for business use cases, 329
  external, 429
  in object life history, 296
Environment
  constraints from, 236, 461–464
  requirements from, 184–186, 196, 487–489
Error rates, usability for, 179–180
Errors
  in requirements activity, 8
  in software development, 262
Essence
  discovering, 25
  in trawling, 107–109, 386
Estimates
  in blastoff, 382
  cost, 23, 37, 255–256, 272, 504–505
  effort, 434–435
input, 434
Evaluation of brainstorming ideas, 119
Event boundary names, 438
Event business names, 438
Event-driven use cases, 67
   adjacent systems in, 72, 79–86
   agility guide for, 67
   business events in, 73–78
   business use cases in, 78–79, 86–90
   context in, 70–72
   product use cases in, 86–90
   scope in, 69–70, 87
   work in, 67–72
Event/use case models, 440
Events see Business Events
Evolution of requirements, 11
Exceptions
   in business use case workshops, 114
   in functional requirements, 161–162
   scenarios for, 145–146
   specification reviews for, 324
Executive sponsors, 47
Existing documents, 440
Existing procedures, 280, 347. See also Current situation and environment
Expectation management, 334
Expected physical environment, 487
Experts
   domain, 317
   as stakeholders, 45, 52, 54
   <<extend>> constructs, 88
Extensibility requirements, 486
External entities, 429
Externally stored data in function point counting, 518–520
Extreme programming
   essence in, 109
   requirements specifications in, 32
   testing in, 203, 218
   whiteboards for, 19

F
Facilitators for retrospectives, 360–362, 417–418
Facts
   blastoff for, 37
   in reusing requirements, 305
   in risk analysis, 341
   in specification template, 238, 467
Fagan inspections, 323–324
Failures, fit criteria for, 209–210
Family therapy in trawling, 128
Fault tolerance requirements, 183, 485
Feasibility studies, 65
Feasible goals, 57
Feature Points, 508
Features
   in functional requirements, 166–167
   unnecessary, 275–276, 406
Feedback
   change from, 358–360
   interviews for, 387
   prototypes for, 287, 291, 411
Ferdinandi, Patricia, 148
File integrity, fit criteria for, 215
Filtration criteria for retrospectives, 423–424
Financial beneficiaries, 52
Financial constraints, 61
Financial scandals, 194
Finding
   business events, 76–78, 390–391
   functional requirements, 157–160
   nonfunctional requirements, 195–199
First-cut low-fidelity prototypes, 37
First-cut risk analysis, 375–376
First-cut work context, 44–45
Fit criteria, 203
   agility guide for, 203–204
   ambiguous, 339
   for functional requirements, 217–218
   measurement scale for, 206, 271
   for nonfunctional requirements, 208–209
   cultural and political, 216
   legal, 216–217
   look and feel, 211
   maintainability, 215
   operational, 214–215
   performance, 213–214
   product failure, 209–210
   security, 215–216
   subjective tests, 210–211
   usability and humanity, 212–213
   for project purpose, 219
   purpose of, 26, 204–205
   Quality Gateway for, 28, 270–271
   rationale for, 206–207
   reviewing, 405, 440
   for solution constraints, 219–220
   for testability, 453
   testing, 204, 270–271
   in use cases, 218–219
   in writing requirements, 245–246, 401–402
Flesch-Kincaid Grade Level Score, 213
Flesch Reading Ease Score, 212–213
Flows
   in business events, 77–78
   in model, 366–367
Flows (cont.)
  in scope, 45
  in trawling, 97
Focus
  on deliverables, 346–347
  in retrospectives, 360
Follow-up
  blastoff, 381
  for new products, 500
Forces in patterns, 313
Formality
  objects for, 318
  in Quality Gateway, 280
Formalized requirements, 259, 441
Formalized system constraints, 441
Formalizing
  requirements, 400
  system constraints, 400
Free tools, 243
Fun in brainstorming, 118
Function point counting, 507–508
  adjustments in, 520–521
  business use cases in, 511–516
  for cost estimates, 63, 256, 504
  help and resources for, 521–522
  overview, 509
  scope in, 509–510
  stored data in, 510–511, 517–520
Functional beneficiaries, 52
Functional requirements, 9, 155
  agility guide for, 155–156
  alternatives to, 161–162, 167–169
  ambiguity in, 162–164
  conflicts in, 338
  defined, 441
  exceptions in, 161–162
  finding, 157–160
  fit criteria for, 217–218, 401
  grouping, 166–167
  identifying, 399–400
  level of detail in, 160–161, 251
  requirements vs. solutions, 165
  in risk analysis, 341
  scope in, 166, 240–241, 468–473
  in specification template, 240, 249–251, 473–474
  technological, 164–165
  in templates, 13
Functionality, 2, 11
Fundamental processes in business use cases, 331

G
Generalizations
  in linguistic meta-models, 106
  removing, apprenticeships for, 385
Generic processes, 347, 365
Geography, significance of, 111
Glossaries, 237–238, 465–466
Go/no go decisions
  in blastoff, 23, 37, 64
  defined, 441
Goals
  in blastoff, 55–60
  in domain analysis, 317
  purpose tracking, 59–60
  relevancy of, 268
  in specification template, 230–232, 455–456
Gold plating
  defined, 441
  identifying, 275–276, 406
Gordon, Peter, 117
Government as stakeholder, 53–54
Gramm-Leach-Bliley Act, 194
Granularity in functional requirements, 160–161
Group comments, 441
Groups
  for brainstorming, 118
  for functional requirements, 166–167
  requirements by type, 244
  for retrospectives, 361, 417
  special-interest, 54
Guard conditions, 144
Guesswork, problems from, 63
Guidelines for trawling, 395–396
Gutenberg, Johannes, 117

H
Hands-on users in specification template, 457–458
Hardware, safety requirements for, 183
Harmful possibilities, scenarios for, 147–148
Hauser, John, 338
Health Insurance Portability and Accountability Act (HIPAA), 194
Help for function point counting, 521–522
High-fidelity prototypes, 292–294
  building, 410–411
  defined, 441
  testing, 413
High-level requirements, 11, 167
Highsmith, Jim, 381
HIPAA (Health Insurance Portability and Accountability Act), 194
History
object life, 296–297
in writing requirements, 248
Holtzblatt, Karen, 136, 283, 290
Homonyms, 162–164
Horse projects
in agile software development, 7
blastoff for, 38
event-driven use cases for, 67
fit criteria for, 204
functional requirements for, 156
nonfunctional requirements for, 172
process, 20
prototyping for, 285–286
Quality Gateway for, 261
scenarios for, 136
specification reviews for, 323
for trawling, 94
writing requirements for, 224
House of Quality tool, 338
Humanity requirements, 178–182
accessibility, 481
ease of use, 477–479
fit criteria for, 212–213
learning, 479–480
personalization and internationalization, 479
understandability and politeness, 480–481
Icons for prototypes, 292
Ideas
in brainstorming, 118
for solutions, 257, 506
Identified stakeholders, 441
Identifiers for traceability, 267, 353
Identifying
business events, 326–328
composite requirements, 400
customer value ratings, 427–428
dependencies and conflicts, 403
domains of interest, 378
estimation input, 434
filtration criteria, 423
functional requirements, 399–400
gold plating, 275–276, 406
interactions, 428
missing requirements, 427
new and changed requirements, 414
nonfunctional requirements, 401
originators, 279
potential requirements, 399
prototyping opportunity, 428
requirements, 243–244
stakeholders, 376–377, 523
users, 49
Imaginary users, personas for, 119–121
Immunity requirements, 190, 493–494
Implementation environment, constraints
from, 235–236, 461–462
Implementation technology, deciding on, 355
<iostream> constructs, 88
Incremental processes, 30–31
Indirect contributions, 269
Individual comments, 442
Individual product use cases, 473
Individual retrospective reviews, 417
Individuals in agile software development, 5
Industry standard setters, 53
Information
for innovation, 112
in requirements knowledge model, 225–227
Initial estimates
in blastoff, 382
defined, 442
Initiation. See Blastoff
Innovation
from active adjacent systems, 81
trawling for, 110–113
Input
in business events, 77–78
from groups, 442
from individuals, 442
for prototyping, 299
Input business use cases, 512–513
Inquiries in business use cases, 512, 515
Inspections for specification reviews, 323–324
Inspectors as stakeholders, 53
Installed systems for new products, 499
Integrity requirements, 188–189, 492
Intended operating environment, 442
Intended operating environment description, 442
Intended products in stakeholder maps, 46
Intention of nonfunctional requirements, 176
Inter-deliverable associations, 348–349
Interactions
in agile software development, 5
identifying, 428
in storyboards, 295
Interest domains
in blastoff, 378
in scope, 42–44
Interested stakeholders in scenarios, 139, 149
Interfaces  
with adjacent systems, 487–488  
assumptions about, 239  
context, 439  
model for, 366  
for prototypes, 293  
Internal stored data in function point counting, 517–518  
Internationalization, 182, 479  
Internet  
for customer self service, 112  
for product comparisons, 111  
Interviews  
mind maps for, 123  
for retrospectives, 361  
snow cards for, 243  
stakeholders, 104–106  
users, 387–388  
videos for, 124  
Intuitive products, 212  
Inventions  
demand from, 111  
prototypes for, 291  
Isolating work in business use cases, 78  
Issues, 252, 345  
adapting Volere process, 345–347  
change, 357–360  
costs, 255–256, 504–505  
ew new problems, 254, 498–500  
notebooks for, 363  
off-the-shelf solutions, 253, 497–498  
open, 252–253, 496–497  
publishing requirements, 350–353  
retrospectives for, 360–363  
risks, 254–255, 502–503  
solution ideas, 506  
tasks, 254, 500–501  
in templates, 14  
tools, 347–350  
traceability, 353–356  
user documentation and training, 256, 504–505  
IT security requirements, 194  
Italy, customs in, 190–191  
Iteration, 30–31  
in business use cases, 331–332  
low-fidelity prototypes for, 288  
for rabbit projects, 19  

Jones, Capers  
on change control, 360  
on cost of repairing errors, 262  
Feature Points by, 508  
on function points, 520  
on prototypes, 283, 286  
on requirements creep, 278–279  
on risks, 340, 503  
Judgment in linguistic meta-models, 106  
Justification for fit criteria, 206–207

K
Kelvin, Lord, 203  
Keywords in mind maps, 122  
Kickoff. See Blastoff  
Kliban, B., 67  
Knowledge sources, 442  
Knowledge vs. specification, 225–227

L
Laddering, 387–388  
Languages  
in functional requirements, 162–164  
maintainability requirements for, 186–187  
for prototypes, 293  
Lasdon, Denys, 111  
Latency requirements, 482  
Latour, Bruno, 51  
Launch. See Blastoff  
Laws  
maintainability requirements for, 186  
of robotics, 183  
Lawyers, 193  
Leakage requirements, 278  
Learning requirements, 479–480  
Legal requirements, 192–195, 197, 495–497  
compliance, 495–496  
fit criteria for, 216–217  
government, 194  
standards, 194, 496  
Legalities as stakeholders, 53  
Lessons learned, 31  
Level of detail  
in functional requirements, 160–161, 251  
in management summaries, 351  
Library domains, 314  
Lifelike work situations, prototypes for, 292–293  
Linguistic meta-models, 106  
Links in mind maps, 122–123
Lister, Tim, 64, 431
Litigation costs, 192–193
Little, Todd, 7
Logical files, 517
Longevity requirements, 486
Look and feel requirements, 176–178
appearance, 476
fit criteria for, 211
style, 476–477
Loops for prototyping, 297–301
Loudness, measuring, 206
Low-fidelity prototypes, 288–292
blastoff for, 37
building, 410
defined, 442
testing, 413–414
Low-level functional requirements, 167
Low-tech approaches, 15, 242–243
Loyalty
connectedness for, 112
disappearance of, 111

M

Maiden, Neil, 116, 146
Maintainability requirements, 186–187, 196,
215, 489–490
Maintenance users in specification template,
459–460
Major risks, 442
Mambo system, 293
Managed risk, 255
Management as stakeholders, 52
Management review in Quality Gateway, 281
Management summaries, 351–352
Mandated constraints
in reusing requirements, 304
in risk analysis, 341
in specification template, 234–237, 460–465
Map suppliers in domains of interest, 43
Maps
mind, 122–124
stakeholder, 46
Market forces as stakeholders, 53
Marketing department as stakeholders, 47
Marketing summaries, 352
Martin, Steve, 7–8
Mass market products, prototypes for, 294
Masters, apprenticeships with, 385
Materials for specification reviews, 323
McConnell, Steve, 8
McMenamin, Steve, 93, 286, 293
Meaningfulness, completeness tests for, 265
Meanings. See also Terms and terminology
ambiguous, 339–340
for deliverables, 346
in functional requirements, 162–164
in linguistic meta-models, 106
in specification template, 237–238, 465–466
in Volere Requirements Process Model, 437–449
Measurability, fit criteria for, 203
Measurable goals, 57–59
Measurements
effort estimates, 62–63
and fit criteria, 204, 206, 271
function point counting. See Function point
counting
and goals, 231–232, 455–456
for prototyping, 298, 300–301
specification reviews for, 342
usability, 181
Meeting locations, 442
Meeting schedules, 442
Merges in activity diagrams, 144
Meta-models, linguistic, 106
Metaphors, prototypes for, 287
Migration to new products, 254, 501–502
Mind maps, 122–124
Missing requirements
completeness tests for, 264–265
custodial, 429
defined, 442
identifying, 427
specification reviews for, 324–325
Misuse cases, scenarios for, 147–148
Mobile telephones, 247
Models, 3–4
apprenticeships with, 102
business use cases, 328
data dictionaries for, 238, 466
domain, 307, 440
for functional requirements, 168–169
requirements knowledge, 225–226
stakeholder involvement in, 105
in trawling, 98
vs. writing, 27
Modified data for new systems, 502

N

Names
for patterns, 313
in retrospectives, 362
Naming conventions
in blastoff, 37, 61–62
in reusing requirements, 305
in specification template, 237–238, 465–466
Napoleonic wars, 335
Negative scenarios, 147–148
Negative stakeholders, 53
Negotiations in agile software development, 6
Neurolinguistic programming (NLP), 106
New problems in specification template, 254, 498–500
New requirements, prototypes for, 414
Nix, Lynne, 381
NLP (neurolinguistic programming), 106
Nominalization in linguistic meta-models, 106
Non-events
  in blastoff, 377
  identifying, 328
Non-functional requirements, 10, 171–172, 174–176
  agility guide for, 172
  cultural and political, 190–192, 216, 494–495
  defined, 442
  finding, 195–199
  fit criteria for, 208–217, 402
  identifying, 401
  indirect contributions of, 269
  legal, 192–195, 216–217, 495–497
  look and feel, 176–178, 211, 476–477
  maintainability, 186–187, 215, 489–490
  operational and environment, 184–186, 214–215, 487–489
  performance, 182–184, 213–214, 482–486
  product failure, 209–210
  security, 187–190, 215–216, 491–494
  vs. solutions, 199–201
  specification reviews for, 324
  in specification template, 251–252, 476–496
  subjective tests, 210–211
  support, 186–187, 490–491
  in templates, 13, 197
  usability and humanity, 178–182, 212–213, 477–481
  use cases in, 174
Normal case scenarios, 140–142
Normal operators as stakeholders, 49
Note taking, 123
Notebooks, 363
Nouns
  in document archeology, 388–389
  in linguistic meta-models, 106
Numbers
  for requirement identification, 244
  in subjective tests, 211

O
Objectives
  blastoff, 371, 438
  of prototypes, 443
Objects
  life history, 296–297
  for reuse, 318
Observations
  in trawling, 101, 103–104
  videos for, 124
Off-the-shelf (OTS) products
  as constraint, 61, 236, 462–463
  functional requirements for, 167
  in specification template, 253, 497–498
Onion diagrams, 46
Open issues, 252–253, 496–497
Open source applications, 61
Operational environment for prototyping, 299
Operational problems in specification template, 229
Operational requirements, 184–186, 196, 214–215, 487–489
Operational support, 51
Operational work area, 46
Opportunities
  from change, 357
  defined, 439
  prototyping, 428
  in purpose, 230
  in specification template, 230
  in trawling, 393
Optimism, problems from, 63
Organization maintainability requirements, 186
Organizing thoughts, mind maps for, 122
Originators
  identifying, 279
  in writing requirements, 245
Osborn, Alex, 118, 387
Outcomes
  from business use case workshops, 114
  in use cases, 218
Output business use cases, 514–515
Output flows in business events, 77–78
Output in scope, 41
Outside world in event-driven use cases, 72
Outsourcing requirements, 167

P
Partial specifications, 322
Participants, 438
Partitions
for business events, 75, 311–312
context, 377
in creativity workshops, 116
in event-driven use cases, 69
in specification template, 470–472
work, 240, 471
Partner systems and applications
constraints from, 236, 462
fit criteria for, 214
in operational requirements, 185
Passwords
in nonfunctional requirements, 200
as problem, 110
Patterns, 307–308
from abstraction, 313–317
business event, 309–313
collections of, 308–309
across domains, 315–317
in linguistic meta-models, 106
for specific domains, 314–315
in trawling, 103–104
Peer review, 281
Pena, William, 242
People, requirements from, 93
Perceived solutions vs. system essence, 25
Performance requirements, 182–184, 196
capacity, 485–486
fit criteria for, 213–214
longevity, 486
precision and accuracy, 484
reliability and availability, 484–485
robustness and fault-tolerance, 485
safety-critical, 483
scalability and extensibility, 486
speed and latency, 482
Perl language for prototypes, 293
Personal notebooks, 363
Personalization, 182, 479
Personas
in specification template, 233
for stakeholders, 49
in trawling, 119–121
Pfleeger, Shari Lawrence, 187
Phases, project, 347
Phones
mobile, 247
usability of, 283
Photographs in trawling, 124–125
Photoshop usability, 180
Physical arrangements in blastoff, 371–372
Physical entities in domains of interest, 43
Physical environment, expected, 487
Pictures
for low-fidelity prototypes, 290
in storyboards, 295
Piggybacking on brainstorming ideas, 118
Planning
prototypes, 408
for specification reviews, 323
Planning tasks in specification template, 500–501
Plots in scenarios, 136
Points for clarification, 443
Policy as system essence, 25
Policy domains, 43
Politeness requirements, 480–481
Political beneficiaries as stakeholders, 52
Political correctness, 192
Political requirements, 190–192, 197, 216, 494–495
Portable devices requirements, 185
Possibilities in specification template, 230
Potential of products, prototypes for, 292
Potential requirements
defined, 443
formalized, 259
identifying, 399
writing requirements from, 225
Potential stakeholders, 443
Potentially reusable requirements, 305
Preamble in specification template, 452
Precision requirements, 484
Preconceptions in trawling, 109
Preconditions
for business use cases, 139
in scenario templates, 148
Preliminary cost estimates, 23
Printing press, 117
Priorities of users, 458–459
Prioritizing requirements
factors in, 333–334
grading in, 335
specification reviews for, 333–337
spreadsheets for, 335–337
timing of, 334
in writing, 247
Privacy requirements, 492–493
Private individual reviews for retrospectives, 417
Probability in risk analysis, 340
Problem identification in trawling, 109–110
Process, 17–19
adapting, 31–33, 345–347
in agile software development, 5
agility guide for, 19–20
case study, 21–23
Process (cont.)
in context, 20–21
incremental and iterative, 30–31
prototyping in, 25–26
Quality Gateway, 28–29
retrospectives, 29
reusing requirements, 29
reviewing specifications, 29–30
scenarios in, 25
trawling, 24–25
Volere. See Volere Requirements Process Model
writing requirements, 26–28
Product-centric approach for business events, 75
Product development as stakeholder, 47
Product scope, 472–473
boundaries, 241, 472
defined, 443
in risk analysis, 341
Product use cases, 86–88
actors in, 89–90
associations with, 349
for functional requirements, 158–160
low-fidelity prototypes for, 289
scenarios for, 150–152
in scope, 241, 472–473
traceability of, 267
Product users in specification template, 457–460
Productivity, usability for, 179–180
Productization requirements, 488
Products
changes to, 357
character of, 173
copyable, 498
failures in, fit criteria for, 209–210
trawling for, 129–131
Programming languages for prototypes, 293
Progressive prioritization, 334
Project blastoff. See Blastoff
Project constraints
in blastoff, 61, 378
defined, 443
in risk analysis, 341
Project drivers
in risk analysis, 340–341
in specification template
clients, customers, and stakeholders, 232–233, 456–457
purpose, 229–232, 454–456
users, 233–234, 457–460
in templates, 13
Project history, 443
Project intention, 444
Project issues. See Issues
Project participant comments, 444
Project purpose
defined, 444
fit criteria for, 219
Pronouns
avoiding, 339
in functional requirements, 164
Protagonists in negative scenarios, 147
Prototyping, 283–285
agility guide for, 285–286
blastoff for, 37
building effort in, 444
defined, 444
diagrams, 407, 409, 412
evaluating, 414
high-fidelity, 292–294, 410–411
for look and feel, 178
loops for, 297–301
low-fidelity, 288–292, 410
metrics, 444
for new and changed requirements, 414
for nonfunctional requirements, 197–198
object life history, 296–297
opportunities, 428, 444
plan context of, 445
planning, 408
in process, 25–26
for reality, 286–288
storyboards, 294–295
in subjective tests, 210
testing, 413–414
Public Company Accounting Reform and Investor Protection Act, 194
Public opinion as stakeholder, 53
Public seminars for specification template, 453
Publishing, 350–351
contractual documents, 351
management summaries, 351–352
marketing summaries, 352
organizing, 353
requirements specification, 347
user reviews, 352
Purpose
in blastoff, 35, 374
of prototypes, 298
in reusing requirements, 304
in risk analysis, 340
in specification template, 229–232, 454–456
Purpose, advantage, and measurement (PAM) approach, 60

Q

Quality Function Deployment (QFD), 275, 338
Quality Gateways, 28–29, 259–260
agility guide for, 260–261
for completeness, 263–265
for consistent terminology, 238, 267–268
for customer value, 274–275
diagrams, 404
completeness reviews, 406
fit criteria reviews, 405
gold-plate identification, 406
relevance reviews, 406
viability reviews, 406
effects of, 260–262
for fit criterion, 28, 270–271
for gold plating, 275–276
implementing, 279–281
for relevancy, 268–270
for requirements creep, 276–279
for requirements vs. solutions, 273–274
in specification reviews, 321–322
for traceability, 265–267
for viability, 272–273
working with, 262–263
Quantifiable benefits as goals, 231, 455–456
Quantified findings, 445
Quantifiers in linguistic meta-models, 106
Quantifying risks, 431–432
Quantity of brainstorming ideas, 118, 386
Questions, 446
for interviews, 105–106, 387
in trawling, 396–397

R

Rabbit projects
in agile software development, 7
blastoff for, 38
event-driven use cases for, 67
fit criteria for, 204
functional requirements for, 156
nonfunctional requirements for, 172
process, 19
prototyping for, 285
Quality Gateway for, 261
scenarios for, 135
specification reviews for, 322
for trawling, 94
writing requirements for, 224

Ranges
fit criteria for, 213–214
uncertainty, in function point counting, 521

Rationale
for fit criteria, 206–207
in writing requirements, 245
Raw materials in business events, 355
Readability criteria, 212–213
Ready-made products, 497
REALbasic for prototypes, 293
Reality, prototyping for, 286–288
Reasonable goals, 57
Reasoning for requirements, 339
Record elements, 517
Red zones, 163
Reduced cost as goal, 231
Reengineering in trawling, 99
Reference step in CRUD checks, 330–331
Rejected requirements, 445
Related patterns, 313
Relationships in mind maps, 122–123
Releases
in prioritizing requirements, 335
requirements for, 489
Relevancy
Quality Gateway for, 268–270
reports for, 423
in requirements knowledge model, 226
reviewing, 406, 445
Relevant facts and assumptions
in blastoff, 37
defined, 445
in reusing requirements, 305
in risk analysis, 341
in specification template, 238, 467
Reliability requirements, 183, 484–485
Religious observances, 192
Repairing errors, cost of, 262
Reports
for blastoff, 380
defined, 438
for retrospectives, 362–363, 420–421, 443
Requirement interaction summaries, 446
Requirement measurement, 446
Requirement questions, 446
Requirements
agile software development, 4–8
defined, 445–446
evolution of, 11
issues. See Issues
overview, 1–3
purpose of, 8–9
Requirements (cont.)
shell, 14–15
vs. solutions, 273–274
and systems modeling, 3–4
templates for, 11–14
types of, 9, 176, 244, 446, 453
Volere process. See Volere Requirements Process Model
Requirements bait, prototypes for, 286
Requirements creep
Quality Gateway for, 276–279
relevancy for, 268
Requirements filters, 446
Requirements knowledge model, 225–226
Requirements leakage, 29, 278
Resources
for function point counting, 521–522
requirements for, 183
Results analysis for prototyping, 300
Retrospectives, 31, 360
comments, 443
diagrams, 415–416, 419, 422
facilitators for, 360–362, 417–418
factors in, 360
filtration criteria for, 423–424
group meetings for, 417
private individual reviews, 417
reports for, 362–363, 420–421, 443
review meetings for, 420
running, 360–362
Reusable components, 497–498
Reusable requirements, 447
Reuse libraries, 447
Reusing requirements, 29, 303
description, 303–306
domain analysis in, 317–318
patterns in. See Patterns product use cases, 88
sources of, 306–307
trends in, 317–318
Reverse-engineering, 126
Reviewed specifications, 447
Reviewing
blastoff results, 380–381
completeness, 406
fit criteria, 405
relevance, 406
requirements specifications, 29–30, 321–322
agility guide for, 322–323
for ambiguity, 339–340
assembling specifications, 437
for business use cases, 325–332
for conflicts, 337–339
for customer value, 332–333, 427–428
design form, 437
diagrams, 425–426, 430, 433, 436
effort, 434–435
estimation input, 434
inspections for, 323–324
interaction, 428
for measurements, 342
missing, 324–325, 427
missing custodial, 429
for prioritizing requirements, 333–337
prototyping opportunity, 428
for risks, 340–342, 431–432
retrospectives, 420
viability, 406
Risks, avoiding, 360
Risks and risk analysis, 64
in blastoff, 37, 63–64, 375–376
checklists for, 340, 431, 448
constraints in, 341
of damage, 183
defined, 447
drivers in, 340–341
functional requirements in, 341
reviewing, 30, 431–432
specification reviews for, 340–342
in specification template, 254–255, 502–503
Road engineering as domain of interest, 43–44
Roads as domains of interest, 42
Robotics, laws of, 183
Robustness requirements, 183, 485
Rogers, Susan, 117
Rules
in business use case workshops, 115–116
maintainability requirements for, 186
S
Sabotage, 273
Safety-critical requirements, 483
Safety inspectors as stakeholders, 53
Safety requirements, 183
Sarbanes-Oxley Act (SOX), 194
Satellite broadcasting domain, 314–315
Scalability requirements, 183, 486
Scale of measurement for fit criteria, 206, 271
Scandals, financial, 194
Scenarios, 135
agility guide for, 135–136
airline check-in agent, 137–144
for alternative cases, 144–145
in business use case workshops, 113–115
diagramming, 142–144
for exception cases, 145–146
negative, 147–148
normal case, 140–142
in process, 25
for product use case, 150–152
templates for, 137, 148–149
in trawling, 391
what if?, 146–147
Schedules
as constraints, 236, 464
as domains of interest, 42
meeting, 442
Scope
in blastoff, 35, 40–45
boundaries in, 241, 472
in business use cases, 526
in creativity workshops, 116
defining, 32
domains of interest in, 42–44
in event-driven use cases, 69–70, 87
first-cut work context in, 44–45
in function point counting, 509–510
in functional requirements, 166, 240–241, 468–473
lead requirements analysts for, 22
product, 241–243, 472–473
in reusing requirements, 305
in risk analysis, 341
in specification template, 240–241, 468–470
in trawling, 100
Scripts, storyboard, 294–295
Second International Workshop on Software Reusability, 318–319
Security experts as stakeholders, 45
Security requirements, 187, 197
access, 491
“and no more”, 189–190
auditing, 189, 493
availability, 188
confidentiality, 187–188
fit criteria for, 215–216
immunity, 493–494
integrity, 188–189, 492
privacy, 492–493
Self check-out supermarkets, 112
Self-documentation
in legal requirements, 193
names for, 237
Seminars for specification template, 453
Seriousness factor in prototyping, 300
Service as goal, 231
Service technicians in specification template, 459–460
Shells, 14–15
for specifications, 27, 454
in writing requirements, 241–243
“Should”, avoiding, 339
Showstoppers, 255
Simulations
prototypes as, 287, 300
for subjective tests, 210
Skeletons, 446–447
Sketches
for low-fidelity prototypes, 290
for process model, 347
Smith, Delia, 307
Snow cards
for specifications, 27
working with, 242–243
Sobel, Dava, 223
Sociology analysis templates, 523–529
Soft systems in trawling, 129
Software
look and feel of, 177
off-the-shelf products. See Off-the-shelf (OTS) products
for prototypes, 292
safety requirements for, 183
Software development
agile, 4–8
errors in, 262
Solutions and solution constraints
in blastoff, 60–61
vs. essence, 107
fit criteria for, 203, 219–220
in patterns, 313
vs. requirements, 165, 199–201
in specification template, 235, 257, 460–461, 506
Sorting prioritization categories, 335
Sound, measuring, 206
SOX (Sarbanes-Oxley Act), 194
Special-interest groups, 54
Special words in functional requirements, 164
Specifications
defined, 447
for functional requirements, 157
in rabbit projects, 19
retrospective reports for, 420–421
reviewing. See Reviewing templates for. See Volere requirements specification template
tools for, 27
Speed requirements, 182, 482
Spelling in cultural requirements, 192
Spolsky, Joel, 9
Sponsors

executive, 47
in specification template, 233
as stakeholders, 47
Spreadsheets, 335–337
Stahl, Leslie Hulet, 124
Stakeholder analysis templates, 523–529
Stakeholder map templates, 523–524
Stakeholders
acceptability of requirements to, 272
in agile software development, 5
in blastoff, 35–36, 45–55, 376–377
clients as, 47–48
completeness tests for, 265
customers as, 48–49
finding, 54–55
in functional requirements, 161
identifying, 376–377, 523
interviewing, 104–106
in process, 21
prototypes for, 284, 287, 290–291, 299
in reusing requirements, 304, 306
in risk analysis, 341
for scenarios, 138–140, 149
in specification template, 233, 457
traceability of, 267
in trawling, 96–97
users as, 49–51
wants and needs of, 448
Stakeholders-goal-scope (SGS) trinity, 40
Standard setters as stakeholders, 53
Standards
branding, 176, 211
legal requirements, 194
in specification template, 496
State transition diagrams, 296
States in object life history, 296–297
Stored data in function point counting, 510–511, 517–520
Stories. See Scenarios
Storyboards, 294–295
Strategic plans, 448
Structure observations in trawling, 103–104
Style requirements, 476
Subject matter experts as stakeholders, 52
Subjective interpretation, 267
Subjective tests, fit criteria for, 210–211
Subjects in domains of interest, 42
Subtypes in function point counting, 517–518
Summaries
in creativity workshops, 116
management, 351–352
marketing, 352
Supermarkets, self check-out, 112
Support requirements, 186–187, 196, 490
Supporting materials in writing, 248
System constraints
defined, 448
formalizing, 400
System experience, 448
System terminology, 448
Systems in event-driven use cases, 69–70
Systems modeling
apprenticeships with, 102
in requirements gathering, 3–4

T
Tables of contents in templates, 12, 227–229, 451–452
Tasks in specification template, 254, 500–501
Team review in Quality Gateway, 281
Technical experts as stakeholders, 54
Technicians in specification template, 459–460
Technological fossils, 75
Technological requirements, 157, 164–165
Technological skills, 272
Technology
deciding on, 355–356
in trawling, 108–109, 131–132
for wikis, 126
Telephones
mobile, 247
usability of, 283
Templates, 11–14
defined, 447
for nonfunctional requirements, 197
for scenarios, 137, 148–149
sociology analysis, 523–529
for specifications. See Volere requirements
specification template
stakeholder analysis, 523–529
stakeholder map, 523–524
Terms and terminology
ambiguous, 339–340
blastoff for, 37, 377
for deliverables, 346
in functional requirements, 162–164
in linguistic meta-models, 106
Quality Gateway for, 238, 267–268
in specification template, 237–238, 465–466
in stakeholder interviews, 105
system, 448
Index

for traceability, 267
in Volere Requirements Process Model, 437–449

Test cases in functional requirements, 218

Testability
fit criteria for, 453
of goals, 232, 455–456

Testers in event-driven use cases, 90

Testing
completeness, 263–265
in extreme programming, 203
fit criteria for, 204, 270–271
prototyping, 413–414
Quality Gateways for, 28
requirements, 453–454
traceability, 265–267
in user environment, 299–300

Thermal map suppliers
as cooperative adjacent systems, 85–86
in domains of interest, 43

Thomsett, Rob, 231

Thought organization, mind maps for, 122

Three strikes approach, 200

Throughput requirements, 183

Throwaway prototypes, 287

Time constraints in blastoff, 61

Time in product failure measurements, 210

Time-triggered business events, 74

Time-triggered business use cases, 515–516

Tolerances
for fit criteria, 207, 271
in subjective tests, 210

Tools, 347–348
in agile software development, 5
free, 243
mapping to purpose, 348–350

Traceability, 353
for business events, 267, 353–356
testing, 265–267

Training in specification template, 256, 505

Transformational thinking, 117

Transition diagrams, 296

Transitions in object life history, 296–297

Translated data for new systems, 502

Translators, analysts as, 94

Travel, customer bookings for, 112

Trawling for requirements, 24–25, 93

adjacent systems in, 393
agility guide for, 93–94
apprenticeships in, 101–103, 385
brainstorming in, 117–119, 386–387
business event knowledge in, 395
business use case workshops in, 113–116
in business use cases, 96–98
clarification questions in, 396–397
creativity workshops in, 116–117, 391
current situation in, 98–101, 108, 385
diagrams, 97, 383–384, 392, 394
document archeology in, 126–128, 388–389
essence of work in, 107–109, 386
event models in, 390–391
family therapy in, 128
for innovation, 110–113
interviews in, 104–106, 387–388
mind maps in, 122–124
observations in, 101, 103–104
personas in, 119–121
photographs in, 124–125
problem identification in, 109–110
product determination in, 129–131
responsibility for, 94–96
scenario models in, 391
soft systems and viewpoints in, 129
techniques for, 132–134, 395–396, 448
technology in, 131–132
use case boundaries in, 393
use case workshops in, 389–390
video in, 124–125, 389
wallpaper in, 124
wikis, blogs, and discussion forums for, 125–126

Triage in prioritizing requirements, 335

Triggers
for business use cases, 139
in scenario templates, 148

Truck depots as domains of interest, 44

Trucking as domain of interest, 42

Turner, Richard, 7

Typefaces, measuring, 206

Types, requirement, 9, 176, 244, 446, 453

U

Uncertainty range in function point counting, 521

Understandability requirements, 480–481

Understanding requirements, importance of, 348

Unduplicated attributes, 516

Unified Modeling Language (UML)
activity diagrams
for functional requirements, 168
for scenarios, 143–144
objects in, 318
use case diagrams, 511

Universal cures, 31

Universal quantifiers, 106

Unmanaged risk, 254–255
Unnecessary features and requirements, 275–276, 406
Unqualified adjectives and adverbs, 339
Unspecified patterns, 106
Update step in CRUD checks, 330–331
Upper Class service, 131
Usability requirements, 50, 178–182
  accessibility, 481
ease of use, 477–479
  fit criteria for, 212–213
  learning, 479–480
  personalization and internationalization, 479
  in prototyping, 300
  for telephones, 283
  understandability and politeness, 480–481
Usage feedback, 448
Use cases
  business. See Business use cases
defined, 448
event-driven. See Event-driven use cases
  fit criteria in, 218–219
  in nonfunctional requirements, 174, 195–197
  product. See Product use cases
  in scope, 241, 472–473
  in trawling, 389–390, 393
  UML use case diagrams, 511
  in writing requirements, 244
User business in specification template, 229–230, 454–455
User documentation in specification template, 256, 504–505
User environment, testing in, 299–300
User-friendliness as requirement, 208
User groups, 449
User management as stakeholder, 47
User problems for new products, 499
User reviews, publishing, 352
Users
  interviewing, 387–388
  personas for, 119–121
  priorities of, 458–459
  relevancy of, 270
  in reusing requirements, 304
  in risk analysis, 341
  in specification template, 233–234, 457–460
  as stakeholders, 49–51

V
Value as goal, 231
Verbs, 106

Version numbers in prioritizing requirements, 335
Viability
  within constraints, 272–273
  reviewing, 406
Video in trawling, 124–125, 389
Viewpoints in trawling, 129
Virgin Atlantic, 131
Viruses, 190
Visual Basic for prototypes, 293
Volere Requirements Process Model, 1, 15–16, 365–366
blastoff
  business terminology in, 377
  communication in, 372
  constraints in, 378
  context partitioning in, 377
  diagrams for, 369–370, 373, 379
  domains of interest in, 378
  first-cut risk analysis in, 375–376
  initial estimates in, 382
  non-events in, 377
  objectives, 371
  physical arrangements in, 371–372
  purpose determination in, 374
  reports for, 380
  results review, 380–381
  stakeholders in, 376–377
  work context in, 374
prototyping
  diagrams, 407, 409, 412
  evaluating, 414
  follow-up, 381
  high-fidelity, 410–411
  low-fidelity, 410
  for new and changed requirements, 414
  planning, 408
  testing, 413–414
Quality Gateway, 404–406
retrospectives
  diagrams, 415–416, 419, 422
  facilitators for, 417–418
  filtration criteria for, 423–424
  group meetings for, 417
  private individual reviews, 417
  reports for, 420–421
  review meetings for, 420
reviewing requirements
  assembling specifications, 437
  for customer value, 427–428
  design form, 437
  diagrams, 425–426, 430, 433, 436
effort, 434–435
estimation input, 434
interaction, 428
missing, 427
missing custodial, 429
prototyping opportunity, 428
for risks, 431–432
summary, 368
terms used in, 437–449
trawling for requirements
adjacent systems in, 393
apprenticeships in, 385
brainstorming in, 386–387
business event knowledge in, 395
clarification questions in, 396–397
creativity workshops in, 391
current situation in, 385
diagrams, 383–384, 392, 394
document archeology in, 388–389
essence of work in, 386
event models in, 390–391
interviews in, 387–388
scenario models in, 391
techniques for, 395–396
use case boundaries in, 393
use case workshops in, 389–390
video in, 389
working with, 366–367
writing requirements
composite requirements identification, 400
customer value, 402–403
dependencies and conflicts identification, 403
diagrams, 398
fit criteria, 401–402
formalizing requirements, 400
functional requirements identification, 399–400
nonfunctional requirements identification, 401
potential requirements identification, 399
system constraint formalization, 400
Volere requirements specification template, 227
constraints in, 234
assumptions in, 239–240, 467–468
from environment, 236, 463–464
facts in, 238, 467
mandated, 234–237, 460–465
naming conventions and definitions, 237–238, 465–466
off-the-shelf products, 236, 462–463
data requirements in, 475
functional requirements in, 240–241, 249–251, 468–474
nonfunctional requirements in, 251–252
cultural and political, 494–495
legal, 495–497
look and feel, 476–477
maintainability, 489–490
operational and environment, 487–489
performance, 482–486
security, 491–494
support, 490–491
usability and humanity, 477–481
preamble, 452
project drivers in, 229–234
clients, customers, and stakeholders in, 232–233, 456–457
purpose, 229–232, 454–456
users, 233–234, 457–460
project issues
costs, 504–505
new problems, 254, 498–500
off-the-shelf solutions, 253, 497–498
open, 252–253, 496–497
risks, 254–255, 502–503
solution ideas, 506
tasks, 254, 500–501
user documentation and training, 256, 504–505
requirements types, 453
shell in, 454
for specification reviews, 324
tables of contents for, 227–229, 451–452
testing requirements, 453–454
W
Waist-High Shelf pattern, 308–309
Waiting room, 256, 505–506
Wallpaper in trawling, 124
Warning messages, 193
Weather as domain of interest, 42
Weather forecasting service
as autonomous adjacent systems, 84
in domains of interest, 43
Weather stations in domains of interest, 43
Web-based products
look and feel of, 177
for prototypes, 293
Weights for prioritizing requirements, 337
Weinberg, Jerry, 8
What if? scenarios, 146–147
Whiteboards, 19
Wider environment in stakeholder maps, 46
Wikis, 243
for nonfunctional requirements, 195
for trawling, 125–126
Wild ideas in brainstorming, 118
Wittenberg, Ethel, 216
Words. See Terms and terminology
Work
context
  in blastoff, 374
  defined, 449
  in scope, 44–45
  in trawling, 97
  in event-driven use cases, 70–72
partitioning. See Partitions
reengineering, 99
  in scope, 40
Work areas, measuring, 508
Work description and demonstration, 449
Work knowledge, 449
Working models in trawling, 98
Workplace environment, constraints from, 236, 463–464
Workshops, use case, 113–114, 389–390
  business rules in, 115–116
  outcomes from, 114
  scenarios in, 113–115
  videos for, 125
World changes, 358
Writely tool, 243
Writing requirements, 26–28, 223
  agility guide for, 223–225
  atomic requirements in, 243–248
  composite requirements in, 400
  customer value in, 246–247, 402–403
  dependencies and conflict identification in, 403
  diagrams, 398
  fit criteria, 245–246, 401–402
  formalizing requirements, 400
  functional requirements, 399–400
  knowledge vs. specification in, 225–227
  nonfunctional requirements, 401
  potential requirements in, 225, 399
  shells in, 241–243
  system constraint formalization, 400
  templates for. See Volere requirements specification template
Y
Yourdon, Ed, 65