Page numbers in italics refer to figures and tables.

A
acknowledgment and rewards, 44
action plans in a new job, 15–16
Adaptive Software Development, 276
ad hoc development process, 270, 276
administrators
defect-tracking systems, 151
document management, 142
Adobe Dreamweaver, 96
agendas, meeting, 36, 183
agile development processes, 275–279
alpha releases, 104, 113, 114
ambiguity, in product definitions, 90
application programming interfaces (APIs), 158, 167, 342
assessment
project, 356–358
technology. See technology assessment
tools and methods. See development tools assessment
assumptions, in estimates, 226
attitude, poor, 45–46
automated code testing, 314
Axure RP, 96

B
backups, 134, 135–140
balancing work and life, 44
behavioral questions, 67
best practices
backups, 135–136
internationalization, 381–384
meetings, 182
source control versioning, 143–145
beta releases, 113, 115
big picture managers, 180
Borland SilkTest, 314
bosses. See CEOs
boundaries and empowerment, 33
bugs. See defects, product
build engineers, 147
build methods and processes, 145–147
build numbers, 126
business tasks, 362–364
business workflow diagrams, 22–23, 385–390

C
calendar days, estimating, 218–219, 255–256, 323–324
candidate selection, 65–66
designing a process, 66–67
interview preparation, 74–77
interview types, 77–79
reviewing and deciding, 79–80
screening, 73–74
trait evaluation, 67–73
CB (cost-benefit) analysis, 351–356
CEOs, 192, 194, 196
change control process, 212, 259–261
clickable user interface prototypes, 95–96
coaching, 40–42, 44–46, 75–77
code
  artifacts, 101–102
  automated testing of, 314
  custom, 106, 111
  managers writing, 62–63
  modules, disorganized, 106
  reusing prototype, 95

collaboration
  executive team, 201
  interdepartmental, 59–61, 184–186
  marketing and engineering, 97–99
  office layout and, 58–59
  product definition through, 86–89
  promoting culture of, 178–179
comments, file check-in, 144
communication
  with boss, 193–196
  vs. concentration, 56–58, 61
  with customers, 204–206
  with executive team, 200–201
  interdepartmental, 59–61
  office layout and, 58–59
  organizational structure and, 366, 368, 369, 370
  skills, evaluating, 71
  team management and, 33–37
company culture
  candidate selection and, 69
  corporate style, 179
  engineering team,
    perceptions of, 188
  interteam problem resolution and, 184–186
  management style, 180
  meetings and, 180–184
  new jobs and, 19
  peer relationships, 186–187
  understanding and influencing, 177–179
company growth. See also
  organizational structure
customization costs and, 344–347
interdepartmental relations and,
  184–185
meeting style and, 180–181
quality assurance staffing and, 308
role of development managers
  and, 1–2
stages, 3
system scalability and,
  158–160, 342
corporate style, 179
corporate values. See company values
corporate workflow diagrams, 22–23, 385–390
cost-benefit (CB) analysis, 351–356
crime and internationalization,
  380–381
crimes found by, 328–329
crystal Clear, 276
currency and internationalization,
  380–381
customers
  defects found by, 328–329
  early releases and, 114–115
  getting to know, 21–22
impact of retiring software on, 117–118
perception of product, 100–104
as a resource for future planning, 338–340
sales and proposals, 206–212
satisfaction of, 203–206
customer support, 93, 103, 212–213
customer-use models, 102
customization, 106, 111, 344–347

D
daily stand-up meetings, 35
data
automated testing for, 314
collection meetings, 183
reporting and analysis, 169–171
retiring software and
customer’s, 118
warehouses, 170–171
database design and
internationalization, 379–380, 382
dates and internationalization, 381
decision branching in process
modeling, 289, 290
decision drivers for ranking defects,
320–321
decision-making meetings, 183
defects, product
customer perceptions and, 100–101
customer status updates on, 122
failure mode analysis, 160–162
impact on quality and productivity,
330–333
quality assurance metrics, 324–330
ranking and selecting, 319–324
tools for tracking, 149–151,
309–312
delays, project, 237–238, 246–248
department, 12
dependencies, 249
dependent component
numbering, 128
detail managers, 180
development managers
promoting software engineers to,
61–63
role and definition of, 1–2
starting jobs. See new jobs, starting
development processes, 239–240,
267–270
ad hoc, 270, 276
agile, 275–279
customizing, 279–280
improving, 287–288, 295–299
introduction and enforcement,
281–284
iterative, 273
modeling, 288–295
selecting, 281
spiral, 274–275
test- and model-driven, 279
waterfall, 270–273
development teams
candidate selection. See candidate
selection
effective. See effectiveness in
development teams
management of. See managing
development teams
new managers of. See new jobs,
starting
promoting respect for, 188
development tools assessment,
133–135
backups, 134, 135–140
build methods, 145–147
defect tracking, 149–151, 309–312
release process, 147–149
selection, 151–154, 240
source control versioning, 142–145
user interface prototyping, 95–96
diagrams
  corporate business workflow, 22–23, 385–390
  process, 20
    in product definitions, 90
differential backups, 137–138
diffused decision-making
  managers, 180
discussions summaries, 14–15
disk-only backups, 139–140
documentation
  build processes, 146
  for customers, 102
  development, 140–142
  error handling, 162–163
  product specifications, 90
  of products to be revised, 105, 106–107
  release, 121
  software and system, 156–159
  third-party packages, 165–166
  Dreamweaver, Adobe, 96
dual-bar tracking Gantt charts, 254
  Dynamic Systems Development Method, 276

E

effectiveness in development teams
  communication, 59–61
  definitions of, 51–52
  office space, 58–59
  organization, 52–56
  promoting engineers to management, 61–63
  software engineers, 56–58
  email exchanges for problem resolution, 185
  emergency releases, 120
  employee reviews, 46–50
  employees, problems with, 44–46
  empowerment, 32–33
  encouragement, 42–44
  end of life, product, 116–119
  engineers. See development teams
  engineers, build, 147
  enthusiasm, 72, 206, 235, 242
  error handling, 162–163
  essay performance reviews, 47
estimates
  building, 217–224
  collecting project data for, 226–230
  quick and revisable, 87–88
  writing and delivering, 224–226
  estimation bias, 221–222
evaluation
  technology. See technology assessment
tools and methods. See development tools assessment
Evolutionary Development (Evo), 276
executive status summaries, 200
executive team, relationships with,
  191–192, 197–201
  CEO or boss, 192–196
  meetings, 200
experience, evaluating, 70
extreme programming (XP), 276–277

F

failure mode analysis, 160–162
Feature-Driven Development, 276
five rules of project management, 246–248
flexible management, 29–30
flexible project lead approach, 54–55
flexible project teams, 374–375
foundational skills, 67–68
full backups, 137–138
fun, having, 63
G

Gantt chart programs, 220, 249–255
Gentleman Farmer example, 130–131
Git, 144
GiveMeABreakFast Software example, 130–131
goals
coaching and individuals’, 40–42
project, 234–235
understanding boss’s, 193–194
growing
companies. See company growth
development teams. See candidate selection; effectiveness in
development teams
relationships. See relationships
GUIs (graphical user interfaces). See user interfaces (UIs)

H

handshake annual review, 49
hidden product features, 101
hierarchical organizational structure, 369–370
high-trust behaviors, 28–29
hiring. See candidate selection
hit-and-run interviews, 78
honeymoon period, 13
human factors engineers, 96–97
humor, 63

I

image projects, 352, 354
import/export design and internationalization, 382
incremental backups, 137–138
independent component numbering, 128
informal conversations, 200
information
gathering in new jobs, 14–16, 17–18
presentation meetings, 183
intellectual property (IP), 20, 133–135, 157–158
interaction framework for projects, 239
international support, 377–378
assessment of, 171–172
best practices, 381–384
considerations, 378–381
interruptions, software engineer, 56–58, 61
interviews, 74–79
IP (intellectual property), 20, 133–135, 157–158
isolating development teams, 61
iterative development process, 273
iterative steps in process models, 289, 290

J

JUnit, 314

K

kickoff meetings, 242–243
knowledge questions, 67

L

languages, foreign, 380. See also translation (language)
late performance reviews, 48
legacy code, 106–107
legal issues, 117, 380
lightweight development processes, 269–270, 275
limited releases, 113, 115
locales in internationalization, 381
localization. See international support
low-trust behaviors, 26–27, 28–29
lunch committee interviews, 78

M
macro variables and build identification, 145
maintenance time, calculating, 229–230
management style, 180
managers writing code, 62–63
managing development teams, 25
  coaching, 40–42
  communication, 33–37
  conflict resolution, 37–39
  five rules of project management, 246–248
  motivation, 42–44
  perspective on, 359–360
  problem employees, 44–46
  reviews and evaluations, 46–50
  training, 39–40
  values and, 26–33
marathon interviews, 77
marketing departments
  office organization and, 58–59
  overselling by, 100–101, 208–209
  product definition involvement, 86–89
  relationships with engineering teams, 85–88, 97–99
  release criteria and, 120
  understanding customers from, 21–22
  working with, to close deals, 206–212
markets, potential, 339–340
matrix organizational structure, 54, 370–372
mature products, identification of, 126
measurement systems and internationalization, 381
MediaWiki, 141–142
meetings
  company style, 180–182
  customer, 206
daily stand-up, 35
effective, 182–184
  one-on-one, 34–35
performance review, 47
  team, 36–37
weekly project status, 35
merging modified source files, 143
metric system and internationalization, 381
micromanaging, 32–33, 227
Microsoft
  Corporation, 72
  Excel, 5, 224, 255
  Project, 220, 250–251, 254
  Project Server, 142
  Visio, 96
  Visual Basic, 95
minimum-typical-maximum (min-typ-max) values, 258–259, 290, 293
minor release numbers, 126
mission statements, 19
model-driven development, 279
modeling project schedules, 257–259
modified waterfall development process, 271–273
modules, disorganized code, 106
motivation, 42–44

N
new job portrayals, 10–11
new jobs, starting, 9–10
  corporate business workflow, 22–23
  co-workers, 16–19
  customers, 21–22
information, collecting and summarizing, 14–16
learning the technology, process and products, 19–21
training, 13–14
urgent issues, 11–13
noise. See interruptions, software
engineer
normalization factors and pre-release
trends, 328
notes, file check-in, 144
numbering releases. See version
identification
NUnit, 314

O
office space, 58–59
offsite storage of backups, 138–139
one-on-one meetings, 34–35
one-page project assessment, 356–358
OpenOffice.org Calc, 5, 224, 255
organizational structure, 361–375. See
also company growth
development teams, 52–55
flexible project teams, 374–375
hierarchical, 369–370
hundred-plus–person company,
368–375
matrix, 54, 370–372
one-person company, 363
small product teams, 372–374
task types and, 362–363
twelve-person company, 365–366
twenty-four– to fifty-person
company, 366–368
two-person company, 364–365
overhead, calculating, 229–230
overselling products, 100–101
overtime, 44

P
pair programming, 276–277
parallel tasks in process modeling,
289, 290
passive alpha releases, 104
patch releases, 126, 127
peer relationships, 186–187
people skills, evaluating, 71
people topics, discussing, 33, 34
performance, job, 17, 46–50
performance reviews, 45–46
personality of job candidates, 71
personnel changes and
documentation, 157
pictures in product definitions, 90
Plone, 142
post-release review, 122, 123–124
presentation skills, evaluating, 71
prioritization
in action plans, 15
defect ranking, 319–322
of product requirements, 91
in projects, 238–239
of release standards, 120
of tasks, 12
problem employees, 44–46
problem resolution, interteam,
184–186
problem-solving ability, evaluating,
72–73
process diagrams, 20
product definition, 85–86
customer perceptions, 100–104
partnering with marketing teams,
97–99
process, 86–89
prototypes, 93–97
quality assurance team
involvement, 315–316
revising existing products, 105–107
specifications, 89–91, 92
whole product concept, 91–93
productivity projects, 352, 354
products
  assessment. See technology assessment
  avoiding hidden problems in, 341–343
  definition. See product definition
documentation, 156–158
learning, in new jobs, 19–21
overhauling, 343–344
planning. See strategic and product planning
  releases of. See releases
  scalability, 158–160
programmers. See development teams
  project leads, 53, 54–55
  project managers, 53, 370–372. See also project leads
projects
  assessing components, 357–358
  collecting data, 226–229
  communication in, 33, 35–36
  delays, 237–238
  estimating. See estimates
  frameworks, 239–240
  goals, understanding, 234–235
  kickoff, 242
  management, 246–248
  plans, 240–242
  priorities, setting, 238–239
  processes. See development processes
  selecting teams, 235–238
  timelines, 240
  tracking. See tracking projects
  proposals, 209–212
  prototypes, 90, 93–97, 274

Q

QA (quality assurance). See quality assurance (QA)

quality
  assessment, 306–307
  defining, 303–304
  encouraging company to value, 304–306
  importance of, 203–303
quality assurance (QA), 301–302
  building an effective team, 307–308
  communication with development team, 60–61
  impact of defects, 330–333
  key activities of, 314–319
  key processes of, 319–324
  language translation and, 382
  metrics for, 324–330
  tools and environment, 309–314
  quotes, for sales department, 207

R

relationships
  with customers, 206
  with customer service team, 212–213
  executive. See executive team, relationships with interdepartmental. See company culture
  with marketing, 85–88, 97–99
  peer, 186–187
releases
  early, 104, 112–115
  planning, 109–112
  planning product end of life, 116–119
  process for, 119–124
  Software for the Gentleman Farmer example, 130–131
  three-number release IDs, 125–126
  tools for, 147–149
  version identification, 124–129
remote work, 57–58
repair quality, measuring, 331–333
reporting, 169–171
request for quote (RFQ), 209–212
required projects, 352, 354
requirements. See product definition
resolution notes in defect-tracking, 151
respect for team members, 32
restorations, testing backup, 136
return-on-investment (ROI), 322–324
return-per-cost (RPC), 322–323, 324
revenue projects, 352, 354
review cycles. See product definition
rewards, staff, 44
RFQ (request for quote), 209–212
risk management, 261–264, 274
risk reduction projects, 352, 354
ROI (return-on-investment), 322–324
rollout, optimizing product, 344–347
rough quotes, 207, 221
RPC (return-per-cost), 322–323, 324
sales request processes, 207, 209–212
sales teams. See marketing department
scalability
products, 342
system, 158–160, 342
screening job candidates, 73–74
Scrum process, 277–279
SCV (source control versioning), 142–145
secrecy, 17–18
security assessment, 167–169
self-appraisals, 47
self-promotion, 192–193
selling the company to job candidates, 75, 76
sense of humor in job candidates, 73
sincerity with team members, 30–31
Six Sigma Analysis, 288
small product teams, 372–374
software development processes. See
development processes
software engineers. See development
teams
software escrow, 139
software life cycle tasks, 362–364
source control versioning (SCV), 142–145
source file controls, 143–145
specifications. See product definition
speed problems in databases, 169
spiral development process, 274–275
spreadsheets
cost-benefit analysis, 355
defect ranking and selecting, 322–324
downloading and customizing example, 4–5
estimates, 222–224
overhead calculation, 229–230
process modeling, 290–291, 293–298
project data, 227–229
project tracking, 249–250, 255–259
quality assurance metrics, 324–330
risk management, 262–264
testing tracking, 316–318
sprints (Scrum process), 277
staff managers, 55
staff substitutions, 236–237
standard practices for testing error
handling, 163
standards, project, 240
starting projects, 233–243
kickoff, 242–243
planning, 238–242
selecting project teams, 235–238
understanding goals, 234–235
story cards, 277

Growing Software
(C) 2009 by Louis Testa

Index 399
strategic and product planning, 337–338
cost-benefit (CB) analysis, 351–356
creating a whole product, 340–341
markets as a resource for, 338–340
one-page project assessments, 356–358
optimizing rollout and customization, 344–347
product roadmap, 351
technical time bombs, 341–343
technology overhauls, 343–344
stress interviews, 77
strict hierarchical managers, 180
success history, evaluating, 68–69
system diagrams, 157–158, 159
system flexibility and maintainability, assessment of, 164–165
system (product) documentation, 156–158
system (product) scalability, 158–160, 342

t

tasks
  dependencies, 249
  lists, 12, 218–220
  time estimates, 220–221
timelines, 240
team-company integration, 52
team effectiveness, 52
team meetings, 36–37
technical leads, 53
technical skills, evaluating, 67–68, 307–308
technical writing teams and office space, 58–59

technology
  learning, in new jobs, 19–21
  review checklist, 21
  training for team members, 39

technology assessment, 155–156
  avoiding hidden problems in, 341–343
data reporting and analysis, 169–171
error handling, 162–163
failure mode analysis, 160–162
international support, 171–172
overhauling, 343–344
security, 167–169
system APIs (application programming interfaces), 167
system documentation, 156–158
system flexibility and maintainability, 164–165
system scalability, 158–160
third party packages, 165–166
templates
  conventions used for, 5
  product definitions, 91, 92
  project assessment, one-page, 357
  project plan, 241
test plan, 316–318
test-driven development, 279
test harnesses, 314
testing
  automated, 313–314, 318–319
  building environments for, 312–313
  error handling, 163
  metrics for, 324–330
  patch releases and, 127
  planning and tracking, 316–317, 318
  scalability, 159
test pass reduction rate, 331–333
third party packages, 165–166
three-number release IDs, 125–126
360-degree review, 47
time estimates, 220–221
time-risk leverage, 263
time tracking, 226–229, 229–230
tracking product defects, 149–151, 309–312
tracking projects, 245–246
 change control process, 259–261
guidelines for project management, 246–248
maintenance and overhead, 229–230
making scheduling predictions and alterations, 257–259
risk management, 261–264
software approaches for, 249–257
time tracking, 226–229
training
 continuing, for team members, 39–40
development process, 282
internal staff on software updates, 122
in new jobs, 13–14
release process cross-, 148
trait evaluation of job candidates, 67–73
translation (language), 379, 382–383

U
   UIs (user interfaces). See user interfaces
   unplanned and unsupported features, 102–104
   urgent issues, 11–13
   usability engineers, 96–97
   use cases
      in API documentation, 167
      in product definition, 90
   user interfaces (UIs)
      automated testing, 314
defect ranking, 320, 321
designing, 96–97
   internationalization and, 383–384
   prototyping tools, 95–96

V
   vacation time and calculating overhead, 230
   values. See company values
   verbal estimates, 226
   version identification, 121, 124–129, 128

W
   waterfall development processes, 270–273
   weekly project status meetings, 35
   weighting defect rankings, 322–324, 326–329
   white box testing, 314
   whole product concept, 91–93, 340–341
   wikis, 141–142
   workflow, corporate business, 22–23, 385–390
   work habits and preferences, evaluating, 70
   work hours, estimating, 218–219, 256. See also calendar days, estimating
   work skills training, 39
   writing and delivering estimates, 224–226
   writing skills, evaluating, 71
   written estimates, 222–226

X
   XP (extreme programming), 276–277