
INDEX

A

Acceptance tests
 automated, 103–105
 communication and, 103
 continuous integration and,
 110–111
 definition of, 100
 developer's role in, 106–107
 extra work and, 105
 GUIs and, 109–111
 negotiation and, 107–108
 passive aggression and, 107–108
 timing of, 105–106
 unit tests and, 108–109
 writers of, 105–106

Adversarial roles, 26–29

Affinity estimation, 146–147

Ambiguity, in requirements, 98–100

Apologies, 12

Apprentices, 183

Apprenticeship, 180–184

Arguments, in meetings, 126–127

Arrogance, 22

Automated acceptance testing,
 103–105

Automated quality assurance, 14

Avoidance, 131

B

Blind alleys, 131–132

Bossavit, Laurent, 89

Bowling Game, 89

Branching, 191

Bug counts, 197

Business goals, 160

C

Caffeine, 128

Certainty, 80

Code
 control, 189–194
 owned, 163
 3 AM, 59–60
 worry, 60–61

Coding Dojo, 89–93

Collaboration, 20, 157–166

Collective ownership, 163–164

- Commitment(s), 47–52
 - control and, 50
 - discipline and, 53–56
 - estimation and, 138
 - expectations and, 51
 - identifying, 49–50
 - implied, 140–141
 - importance of, 138
 - lack of, 48–49
 - pressure and, 152
- Communication
 - acceptance tests and, 103
 - pressure and, 154
 - of requirements, 95–100
- Component tests
 - in testing strategy, 116–117
 - tools for, 199–200
- Conflict, in meetings, 126–127
- Continuous build, 197–198
- Continuous integration, 110–111
- Continuous learning, 19
- Control, commitment and, 50
- Courage, 81–82
- Craftsmanship, 184
- Creative input, 65–66, 129
- Crisis discipline, 153
- Cucumber, 200
- Customer, identification with, 21
- CVS, 191
- Cycle time, in test-driven development, 78
- D**
- Deadlines
 - false delivery and, 73
 - hoping and, 71
 - overtime and, 72
 - rushing and, 71–72
- Debugging, 66–69
- Defect injection rate, 81
- Demo meetings, 126
- Design, test-driven development and, 82–83
- Design patterns, 18
- Design principles, 18
- Details, 201–203
- Development. *see* test driven development (TDD)
- Disagreements, in meetings, 126–127
- Discipline
 - commitment and, 53–56
 - crisis, 153
- Disengagement, 70
- Documentation, 82
- Domain, knowledge of, 21
- “Done,” defining, 73, 100–103
- “Do no harm” approach, 11–16
 - to function, 11–14
 - to structure, 14–16
- Driving, 70
- E**
- Eclipse, 195–196
- Emacs, 195
- Employer(s)
 - identification with, 21
 - programmers *vs.*, 159–162
- Estimation
 - affinity, 146–147
 - anxiety, 98
 - commitment and, 138
 - definition of, 138–139
 - law of large numbers and, 147
 - nominal, 142
 - optimistic, 141–142
 - PERT and, 141–144
 - pessimistic, 142
 - probability and, 139
 - of tasks, 144–147
 - trivariate, 147

Expectations, commitment and, 51
Experience, broadening, 93

F

Failure, degrees of, 174
False delivery, 73
FitNesse, 199–200
Flexibility, 15
Flow zone, 62–64
Flying fingers, 145
Focus, 127–129
Function, in “do no harm”
 approach, 11–14

G

Gaillot, Emmanuel, 89
Gelled team, 168–170
Git, 191–194
Goals, 26–29, 124
Graphical user interfaces (GUIs),
 109–111
Green Pepper, 200
Grenning, James, 145
GUIs, 109–111

H

Hard knocks, 179–180
Help, 73–76
 giving, 74
 mentoring and, 75–76
 pressure and, 154–155
 receiving, 74–75
“Hope,” 48
Hoping, deadlines and, 71
Humility, 22

I

IDE/editor, 194
Identification, with employer/
 customer, 21
Implied commitments, 140–141

Input, creative, 65–66, 129
Integration, continuous, 110–111
Integration tests
 in testing strategy, 117–118
 tools for, 200–201
IntelliJ, 195–196
Interns, 183
Interruptions, 63–64
Issue tracking, 196–197
Iteration planning meetings, 125
Iteration retrospective meetings, 126

J

JBehave, 200
Journeymen, 182–183

K

Kata, 90–91
Knowledge
 of domain, 21
 minimal, 18
 work ethic and, 17–19

L

Lateness, 71–73
Law of large numbers, 147
Learning, work ethic and, 19
“Let’s,” 48
Lindstrom, Lowell, 146
Locking, 190

M

Manual exploratory tests, in testing
 strategy, 118–119
Masters, 182
MDA, 201–203
Meetings
 agenda in, 124
 arguments and disagreements in,
 126–127
 declining, 123

- Meetings (*continued*)
demo, 126
goals in, 124
iteration planning, 125
iteration retrospective, 126
leaving, 124
stand-up, 125
time management and, 122–127
- Mentoring, 20–21, 75–76, 174–180
- Merciless refactoring, 15
- Messes, 132–133, 152
- Methods, 18
- Model Driven Architecture (MDA), 201–203
- Muscle focus, 129
- Music, 63
- N**
- “Need,” 48
- Negotiation, acceptance tests and, 107–108
- Nominal estimate, 142
- Nonprofessional, 8
- O**
- Open source, 93
- Optimistic estimate, 141–142
- Optimistic locking, 190
- Outcomes, best-possible, 26–29
- Overtime, 72
- Owned code, 163
- Ownership, collective, 163–164
- P**
- Pacing, 69–70
- Pairing, 64, 154–155, 164
- Panic, 153–154
- Passion, 160
- Passive aggression, 34–36, 107–108
- People, programmers *vs.*, 159–164
- Personal issues, 60–61
- PERT (Program Evaluation and Review Technique), 141–144
- Pessimistic estimate, 142
- Pessimistic locking, 190
- Physical activity, 129
- Planning Poker, 145–146
- Practice
background on, 86–89
ethics, 93
experience and, 93
turnaround time and, 88–89
work ethic and, 19–20
- Precision, premature, in requirements, 97–98
- Preparedness, 58–61
- Pressure
avoiding, 151–153
cleanliness and, 152
commitments and, 152
communication and, 154
handling, 153–155
help and, 154–155
messes and, 152
panic and, 153–154
- Priority inversion, 131
- Probability, 139
- Professionalism, 8
- Programmers
employers *vs.*, 159–162
people *vs.*, 159–164
programmers *vs.*, 163
- Proposal, project, 37–38
- Q**
- Quality assurance (QA)
automated, 14
as bug catchers, 12
as characterizers, 114–115
ideal of, as finding no problems, 114–115

problems found by, 12–13
 as specifiers, 114
 as team member, 114

R

Randori, 92–93
 Reading, as creative input, 65
 Recharging, 128–129
 Reputation, 11
 Requirements
 communication of, 95–100
 estimation anxiety and, 98
 late ambiguity in, 98–100
 premature precision in, 97–98
 uncertainty and, 97–98
 Responsibility, 8–11
 apologies and, 12
 “do no harm” approach and, 11–16
 function and, 11–14
 structure and, 14–16
 work ethic and, 16–22
 RobotFX, 200
 Roles, adversarial, 26–29
 Rushing, 40–41, 71–72

S

Santana, Carlos, 89
 “Should,” 48
 Shower, 70
 Simplicity, 40
 Sleep, 128
 Source code control, 189–194
 Stakes, 29–30
 Stand-up meetings, 125
 Structure
 in “do no harm” approach, 14–16
 flexibility and, 15
 importance of, 14
 SVN, 191–194
 System tests, in testing strategy, 118

T

Task estimation, 144–147
 Teams and teamwork, 30–36
 gelled, 168–170
 management of, 170
 passive aggression and, 34–36
 preserving, 169
 project-initiated, 169–170
 project owner dilemma with,
 170–171
 trying and, 32–34
 velocity of, 170
 Test driven development (TDD)
 benefits of, 80–83
 certainty and, 80
 courage and, 81–82
 cycle time in, 78
 debut of, 77–78
 defect injection rate and, 81
 definition of, 13–14
 design and, 82–83
 documentation and, 82
 interruptions and, 64
 three laws of, 79–80
 what it is not, 83–84
 Testing
 acceptance
 automated, 103–105
 communication and, 103
 continuous integration and,
 110–111
 definition of, 100
 developer’s role in, 106–107
 extra work and, 105
 GUIs and, 109–111
 negotiation and, 107–108
 passive aggression and, 107–108
 timing of, 105–106
 unit tests and, 108–109
 writers of, 105–106

- Testing (*continued*)
 automation pyramid, 115–119
 component
 in testing strategy, 116–117
 tools for, 199–200
 importance of, 13–14
 integration
 in testing strategy, 117–118
 tools for, 200–201
 manual exploratory, 118–119
 structure and, 15
 system, 118
 unit
 acceptance tests and, 108–109
 in testing strategy, 116
 tools for, 198–199
- TextMate, 196
- Thomas, Dave, 90
- 3 AM code, 59–60
- Time, debugging, 69
- Time management
 avoidance and, 131
 blind alleys and, 131–132
 examples of, 122
 focus and, 127–129
 meetings and, 122–127
 messes and, 132–133
 priority inversion and, 131
 recharging and, 128–129
 “tomatoes” technique for, 130
- Tiredness, 59–60
- “Tomatoes” time management
 technique, 130
- Tools, 189
- Trivariate estimates, 147
- Turnaround time, practice
 and, 88–89
- U**
- UML, 201
- Uncertainty, requirements and, 97–98
- Unconventional mentoring, 179.
 see also mentoring
- Unit tests
 acceptance tests and, 108–109
 in testing strategy, 116
 tools for, 198–199
- V**
- Vi, 194
- W**
- Walking away, 70
- Wasa, 91–92
- Wideband delphi, 144–147
- “Wish,” 48
- Work ethic, 16–22
 collaboration and, 20
 continuous learning and, 19
 knowledge and, 17–19
 mentoring and, 20–21
 practice and, 19–20
- Worry code, 60–61
- Writer’s block, 64–66
- Y**
- “Yes”
 cost of, 36–40
 learning how to say, 52–56