Glossary

AMN Abstract Machine Notation

AOSE Aspect Oriented Software Engineering

ATM Automated Teller Machine

BCS British Computer Society

BRS Business Requirements Specification

BSC Balanced Scorecard

CAR Causal Analysis and Resolution

CBA IPI CMM Based Assessment Internal Process Improvement

CCB Change Control Board

CCS Calculus Communicating Systems

CICS Customer Information Control System

CM Configuration Management

CMM® Capability Maturity Model

CMMI® Capability Maturity Model Integration

COCOMO Constructive Cost Model

COPQ Cost of Poor Quality

COTS Customized Off-the-Shelf

CR Change Request

CSP Communicating Sequential Processes

DAR Decision Analysis and Resolution

DMAIC Define, Measure, Analyse, Improve, Control

DMADV Define, Measure, Analyse, Design, Verify

© Springer International Publishing AG 2017

G. O'Regan, *Concise Guide to Software Engineering*, Undergraduate Topics in Computer Science, DOI 10.1007/978-3-319-57750-0

DOORS Dynamic Object-Oriented Requirements System

DSDM Dynamic Systems Development Method

EAF Effort Adjustment Factor

ESA European Space Agency

ESI European Software Institute

FSF Free Software Foundation

FSM Finite State Machine

GG Generic Goal

GP Generic Practice

GQM Goal, Question, Metric

GUI Graphical User Interface

HP Hewlett-Packard

HR Human Resources

HTM Hyper-Text Mark-up Language

IaaS Infrastructure as a Service

IBM International Business Machines

IDE Integrated Development Environment

IEC International Electro technical Commission

IEEE Institute of Electrical and Electronic Engineers

IPM Integrated Project Management

ISEB Information System Examination Board

ISO International Standards Organization

JAD Joint Application Development

JVM Java Virtual Machine

KLOC Thousand Lines of Code

LCL Lower Control Limit

LDRA Liverpool Data Research Associates

LPF Logic of Partial Functions

LOC Lines of Code

MA Measurement and Analysis

MOD Ministry of Defence

MTBF Mean Time Between Failure

MTTF Mean Time to Failure

MTTR Mean Time to Repair

NATO North Atlantic Treaty Organization

OCL Object Constraint Language

ODC Orthogonal Defect Classification

OID Organization Innovation and Deployment

OMT Object Modelling Technique

OOD Object-oriented Design

OOSE Object-Oriented Software Engineering

OPD Organization Process Definition

OPF Organization Process Focus

OPP Organization Process Performance

OT Organization Training

PaaS Platform as a Service

PCE Phase Containment Effectiveness

P-CM People Capability Maturity Model

PI Product Integration

PL/1 Programming Language

PMBOK Project Management Book of Knowledge

PMI Project Management Institute

PMC Project Monitoring and Control

PMP Project Management Professional

PP Project Planning

PPM Project Portfolio Management

PPQA Process and Product Quality Assurance

Prince Projects In a Controlled Environment

PSP Personal Software Process

PVCS Polytron Version Control System

QPM Quantitative Project Management

RAD Rapid Application Development

RAG Red, Amber, Green

RCA Root Cause Analysis

RD Requirements Development

REQM Requirements Management

RFP Request for Proposal

ROI Return on Investment

RPM Rational Portfolio Manager

RSM Rational Software Modeller

RSKM Risk Management

RUP Rational Unified Process

SaaS Software as a Service

SAM Supplier Agreement Management

SCAMPI Standard CMMI Appraisal Method for Process, Improvement

SCM Software Configuration Management

SEI Software Engineering Institute

SEPG Software Engineering Process Group

SG Specific Goal

SLA Service Level Agreement

SLOC Source lines of code

SOA Service Oriented Architecture

SOW Statement of Work

SP Specific Practice

SPC Statistical Process Control

SPI Software Process Improvement

SPICE Software Process Improvement Capability dEtermination

SQA Software Quality Assurance

SRS System Requirements Specification

SSADM Structured Systems Analysis and Design Method

TDD Test Driven Development

TS Technical Solution

TSP Team Software Process

UAT User Acceptance Testing

UCL Upper Control Limit

UK United Kingdom

UML Unified Modelling Language

URS User Requirements Specification

VAL Validation

VDM Vienna Development Method

VDM♣ Irish School of VDM

VER Verification

VOB Version Object Base

VSS Visual Source Safe

XP Extreme Programming

Y2K Year 2000

References

1. ARC:06 Appraisal Requirements for CMMI V1.2. (ARC V1.2). SCAMPI Upgrade Team. TR CMU/SEI-2006-TR-011, August, 2006
2. I. Bhandari, A case study of software process improvement during development. IEEE Trans. Softw. Eng. **19**(12)
3. M.B. Chrissis, M. Conrad, S. Shrum, *CMMI for Development. Guidelines for Process Integration and Product Improvement*, 3rd edn. SEI Series in Software Engineering (Addison Wesley, Boston, 2011)
4. W. Edwards Deming, *Out of Crisis* (M.I.T. Press, Cambridge, 1986)
5. W. Humphry, *Managing the Software Process* (Addison Wesley, Boston, 1989)
6. J, Juran, *Juran's Quality Handbook* (McGraw Hill, New York, 1951)
7. Ministry of Defence, 00-55 (PART 2) *I* Issue 1, The Procurement of Safety Critical software in Defence Equipment, PART 2: Guidance. Interim Defence Standard, UK, 1991
8. F. O'Hara, Peer reviews—the key to cost effective quality. European SEPG, Amsterdam, 1998
9. Standard CMMI Appraisal Method for Process Improvement. CMU/SEI-2006-HB-002. V1.2, August 2006
10. CMMI Executive Overview. Presentation by the SEI. Software Engineering Institute, 2006
11. CMMI Impact. Presentation by Anita Carleton. Software Engineering Institute, August 2009

© Springer International Publishing AG 2017

G. O'Regan, *Concise Guide to Software Engineering*, Undergraduate Topics in Computer Science, DOI 10.1007/978-3-319-57750-0

Index

A
Agile development, 12
Analogy method, 32
Architecture design, 62
Ariane 5 disaster, 7
Audit escalation, 136
Audit meeting, 135
Audit planning, 134
Audit reporting, 135
Automated software inspections, 101
Axiomatic approach, 192

B
Bags, 217
Balanced scorecard, 143
Barriers to success, 249, 251
Baseline, 77
B method, 198
Booch method, 225
Business case, 29

C
C++, 72
Capability Maturity Model Integration (CMMI), 258, 276, 321
Categories of CMMI processes, 266
CCS, 200
Change control, 82
Change control board, 40
Change request, 40
CICS, 189
Clarity PPM, 282
Class diagrams, 229
Cleanroom methodology, 175
Clearcase, 78
Clear quest, 78
Cloud computing, 311
CMMI maturity levels, 261
CMMI process areas, 267
CMMI representations, 264
COCOMO, 280
Commuting diagram property, 222
Computer security, 180
Configuration control, 78
Configuration identification, 78
Configuration management, 75
Configuration management audits, 84
Configuration management plan, 81
Continuous representation, 264
Cost of poor quality, 90, 158
Cost predictor models, 32
CSP, 200
Customer care metrics, 155
Customer satisfaction metrics, 145
Customized off-the-shelf software, 70

D
Darlington nuclear power plant, 189
Data gathering for metrics, 160
Data reification, 221
Decomposition, 221
Def stan 00-55, 189
Dependability, 171, 178
Development quality metrics, 151
Distributed systems, 308
Document control management, 80
DOORS, 284

E
E-commerce testing, 118
Embedded systems, 312
Enterprise architect, 288
Escrow agreement, 127
Estimation, 30
Estimation in agile, 302
European space agency, 8
Expert judgment, 32

© Springer International Publishing AG 2017

G. O'Regan, *Concise Guide to Software Engineering*, Undergraduate Topics in Computer Science, DOI 10.1007/978-3-319-57750-0

F
Fagan inspection guidelines, 93
Fagan inspections, 5, 21, 92
Finite state machines, 200
Fishbone diagram, 162
Formal methods, 23
Formal specification, 186
Functional requirement, 50
Function-oriented design, 67
Function points, 32

G
Generic goals, 270
Generic practices, 271
Goal question metric, 141

H
Histograms, 164
Human resources and training metrics, 148

I
IEEE standards, 10
Information hiding, 65, 202
Inspection meeting, 98
Integrated development environment, 289
ISO 9001, 245

J
Java, 72

L
LDRA tool, 103, 291

M
Maintenance, 20
Mathematical proof, 193, 222
Measurement, 139
Microsoft project, 281
Model, 9
Model-oriented approach, 191
Mongolian hordes approach, 1

N
Non-functional requirement, 50

O
Object diagram, 230
Object modelling technique, 225
Object-oriented design, 67
Object-oriented programming, 71
Object-oriented software engineering, 225
Open source development, 70
Overture integrated development environment, 190

P
Pair programming, 303
Pareto chart, 165
Parnas, 5, 6, 17, 66, 201
Partial correctness, 199
Partial function, 215
Performance testing, 19
Personal software process, 247
Phase containment effectiveness, 101
Planview, 282
Planview enterprise, 282
PMBOK, 29, 323
Polytron Version Control System (PVCS), 78, 290
Postcondition, 197
Precondition, 197, 199
Predicate transformer, 199
Prince 2, 5, 20, 29, 43
Problem-solving techniques, 161
Process maturity models, 22
Process calculi, 200
Process improvement metrics, 146
Process mapping, 247
Process model, 244
Professional engineering association, 3
Professional engineers, 6
Project, 27
Project board, 29, 40
Project closure, 42
Project management, 21, 27
Project management metrics, 149
Project manager, 29
Project monitoring and control, 38
Project reporting, 42
Proof in Z, 222
Prototyping, 15, 49

Q
Quality audit metrics, 153
Quality center, 117, 293
Quality management, 36

R
Rational software modeler, 288
Rational unified process, 9, 11, 236, 324
Refinement, 186
Reification, 221
Request for proposal, 126
Requirements analysis, 54
Requirements elicitation, 51
Requirements management, 55
Requirements process, 48
Requirements validation, 54, 186
Requirements verification, 55

Requirement traceability, 56, 60
RequisitePro, 286
Risk management, 36

S
Safety critical systems, 182
SCAMPI appraisals, 275
Scatter graphs, 167
Schema calculus, 197
Schema composition, 218, 220
Schema inclusion, 218
Schemas, 218
Scientific revolutions, 191
Scrum methodology, 300
Sequence diagram, 231, 232
Sequences, 216
Service-oriented architecture, 309
Simula 67, 72
Six sigma, 20, 247
Software as a service, 310
Software crisis, 2, 24
Software design, 61, 61, 74
Software engineering, 2, 4, 7, 319
Software failures, 7
Software inspections, 87
Software process, 240
Software process improvement, 242
Software reliability, 171, 172, 174
Software reliability and defects, 173
Software reliability models, 176
Software reuse, 17, 71
Software testing, 18
Source code control management, 81
Specific goals, 270
Specific practices, 270
Spiral model, 10
Sprint planning, 14, 298
Standish group, 3, 24
State diagrams, 232
Statement of work, 127
Statistical process control, 168
Statistical usage testing, 176
Steering group, 252
Story, 13, 298, 304
Structured walkthrough, 91
Supplier selection, 123
System availability, 181
System modelling, 57
System testing, 18, 19

T
Team software process, 247
Test case design, 112
Test cases, 109
Test director, 292
Test driven development, 18, 119, 302
Test environment, 107
Test execution, 113
Test planning, 107, 111
Test process, 107
Test reporting, 114
Test tools, 110, 116
Traceability, 16, 135, 285

U
UAT testing, 19
UML activity diagram, 233
UML diagrams, 228
Unified modeling language, 225, 238
Unit testing, 18
Use-case diagram, 231
User-interface design, 68
User requirements, 47, 59, 61
User stories, 301

V
VDM, 187, 195, 197
Victor basili, 141
VIPER, 193
Visual source safe, 78, 290

W
Walter shewhart, 242
Waterfall model, 9
Watts Humphrey, 257
Weakest precondition, 199
Work breakdown structure, 32

Y
Y2K, 3, 7, 8
Y2K bug, 8

Z
Zermelo set theory, 198
Z specification, 197, 210
Z specification language, 197