
Contents

List of Acronyms	xi
Preface	xv
PART I Introduction	1
1 Introduction	3
1.1 The evolution of medical purpose software	3
1.2 Product quality and software quality	4
1.3 On the need for quality in medical purpose software	7
1.4 Regulatory environments	11
1.5 Verification and validation	13
1.6 Structure of the book	14
PART II Regulations	17
2 EU MDD 93/42/EEC	19
2.1 Background	19
2.2 Content of the Directive 93/42/EEC	20
2.3 The approval process for software as a medical device	24
2.3.1 Qualification	25
2.3.2 Classification	28
2.3.3 Selection of the Authorized Representative and notified body	29
2.3.4 Implementation of a quality management system	29
2.3.5 Documenting software as a medical device	29
2.3.6 Auditing by the notified body	30
2.3.7 Display of the CE marking	30
3 FDA title 21 of US CFR	31
3.1 The role of the Food and Drug Administration	31
3.2 Content of the Codes of Federal Regulation 21 CFR	32
3.3 The approval process for Software as a Medical Device	35
3.3.1 Qualification	35
3.3.2 Classification	37
3.3.3 Implementation of a Quality Management System	38
3.3.4 Documenting the Software as a Medical Device	39
3.3.5 FDA clearance and premarket approval	40

4	Regulations for other markets	43
4.1	Regulatory environment and approval process in Australia	43
4.2	Regulatory environment and approval process in Brazil	44
4.3	Regulatory environment and approval process in Canada	46
4.4	Regulatory environment and approval process in China	47
4.5	Regulatory environment and approval process in Japan	47
4.6	Regulatory environment and approval process in Russia	49
PART III	Standards	51
5	ISO 13485: medical devices—quality management systems—requirements for regulatory purposes	53
5.1	Introduction	53
5.2	Contents	54
5.2.1	The Quality Management System	55
5.2.2	Management responsibility	57
5.2.3	Resource management	58
5.2.4	Product realization	58
5.2.5	Measurement, analysis, and improvement	59
5.3	ISO 13485:2016 versus other Quality Systems	60
5.4	ISO 13485 certification	65
5.5	Use of ISO 13485 in each jurisdiction	66
6	ISO 14971: medical devices—application of risk management to medical devices	69
6.1	Introduction	69
6.2	Contents	70
6.3	Risk concepts applied to medical devices	74
6.4	Examples of hazards, foreseeable sequences of events and hazardous situations	76
6.5	Risk-management methods and tools	78
6.5.1	Failure mode effects analysis	78
6.5.2	Failure mode, effects, and criticality analysis	79
6.5.3	Fault tree analysis	81
6.5.4	Hazard analysis and critical control points	81
6.5.5	Hazard operability (HAZOP) analysis	82
6.5.6	Preliminary hazard analysis	82
6.5.7	Markov analysis	82
6.6	Use of ISO 14971:2007 in each jurisdiction	84
7	IEC 62304: medical device software—software life-cycle processes	87
7.1	Introduction	87
7.2	Content	89
7.2.1	Software Development Process	89
7.2.2	Maintenance process	90

7.2.3	Software risk management process	90
7.2.4	Software configuration management process	92
7.2.5	Software problem resolution process	92
7.3	Use of IEC 62304 in each jurisdiction	92
8	IEEE 1012 and ISO/IEC 29119: standards for software verification	95
8.1	IEEE Std 1012 for system and software verification and validation	95
8.1.1	Integrity levels	97
8.1.2	Common V&V activities	97
8.1.3	Software V&V activities	97
8.2	ISO/IEC 29119 software testing	99
8.2.1	ISO/IEC 29119-1: concepts & definitions	100
8.2.2	ISO/IEC 29119-2: test processes	100
8.2.3	ISO/IEC 29119-3: test documentation	104
8.2.4	ISO/IEC 29119-4: test techniques	104
8.2.5	ISO/IEC 29119-5: keyword-driven testing	105
PART IV	Verification and validation techniques	107
9	Static testing	109
9.1	Introduction and background	109
9.2	Static testing	110
9.3	Static analysis	111
9.3.1	Control flow analysis	111
9.3.2	Data dependence analysis	114
9.3.3	Control dependence analysis	120
10	Dynamic testing	121
10.1	Introduction	121
10.2	Specification-based testing technique	122
10.2.1	Equivalence partitioning	122
10.2.2	Boundary value analysis	123
10.2.3	State transition testing	124
10.2.4	Cause–effect graphing and decision table testing	125
10.2.5	Syntax testing	127
10.2.6	Combinatorial test techniques	128
10.2.7	Scenario testing and use case testing	131
10.2.8	Random testing	132
10.3	Structure-based testing technique	132
10.3.1	Statement testing	132
10.3.2	Branch/decision testing	133
10.3.3	Condition testing	135
10.3.4	Data flow testing	135
10.4	Error-guessing testing technique	136
10.4.1	Error-guessing	136

11 Formal verification	137
11.1 Introduction and background	137
11.2 Formal specification	138
11.2.1 Ambient calculus and ambient logic	138
11.2.2 Linear temporal logic	142
11.3 Model checking	143
11.4 Static and dynamic (formal) verification	145
11.5 Summary	145
 PART V Techniques, methodologies, and engineering tasks for the development, configuration, and maintenance	 147
12 Prescriptive software development life cycles	149
12.1 Software as a product	149
12.2 Software development strategies	150
12.3 Waterfall models	152
12.3.1 The waterfall	152
12.3.2 The V-model	153
12.4 Evolutionary models	154
12.4.1 Prototype models	154
12.4.2 The incremental model	156
12.4.3 The spiral model	156
12.5 Choosing the best software development model	159
 13 Agile software development life cycles	 161
13.1 The Agile Manifesto	161
13.2 Scrum	164
13.2.1 Roles	164
13.2.2 Events	165
13.3 Agile testing practices	166
13.3.1 Test-Driven Development	166
13.3.2 Acceptance Test-Driven Development	168
13.3.3 Behavior-Driven Development	169
13.4 Agile in a regulated environment	170
 14 Project management	 173
14.1 Introduction	173
14.2 Initiating	175
14.3 Planning	177
14.3.1 Setting the goals	177
14.3.2 Assigning the responsibilities	178
14.3.3 Defining the scope	178
14.3.4 Planning time and costs	182
14.4 Executing	184

14.5 Monitoring and controlling	186
14.6 Closing	187
15 Risk management	189
15.1 Risk assessment overview	189
15.2 Risk assessment workflow	192
15.3 Static versus dynamic safety risk scenarios	196
15.4 Probabilistic risk model	199
15.5 Application to the case study	200
15.5.1 Safety critical factor identification	200
15.5.2 Risk analysis	201
15.5.3 Risk scenario development	202
15.5.4 Probabilistic risk model	204
15.5.5 PRM analysis and risk evaluation	206
16 Requirements management	209
16.1 Background	209
16.2 Types of requirements	210
16.3 Requirements development	213
16.3.1 Requirements elicitation	214
16.3.2 Requirements specification	214
16.3.3 Requirements verification and validation	217
16.4 Requirements traceability	217
17 Design controls and development management	219
17.1 Background	219
17.2 Design controls	220
17.3 Design control and development templates	221
17.3.1 Intended use template	222
17.3.2 Risk management file template	224
17.3.3 Software development plan template	224
17.3.4 Software requirements specification template	225
17.3.5 Software architectural design template	226
17.3.6 Software detailed design template	227
17.3.7 Test plan template	228
17.3.8 Test case specification template	229
17.3.9 Test procedure specification template	229
17.3.10 Test incident report template	230
17.3.11 Test summary report template	231
17.3.12 Review report template	231
17.3.13 Meeting report template	231
18 Test management and defect management	233
18.1 Software testing principles	233
18.2 Software testing strategies	234

18.3 A software testing process	235
18.3.1 Test planning, monitoring, and control	236
18.3.2 Test analysis	237
18.3.3 Test design	237
18.3.4 Test implementation	238
18.3.5 Test execution	238
18.3.6 Test evaluation exit criteria	239
18.3.7 Test closure	239
18.4 Test metrics	239
18.5 Defect management	243
19 Change management, configuration management, and change management	245
19.1 Change management	245
19.2 Configuration management	249
19.3 Incident management	251
PART VI Conclusions	255
20 Conclusions	257
20.1 Perspectives	257
20.2 Criticality	262
20.3 Conclusions	265
References	267
Index	271