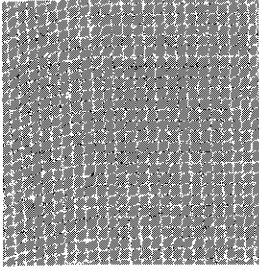


3rd
edition

REQUIREMENTS ANALYSIS AND SYSTEM DESIGN

Leszek A. Maciaszek





Contents

<i>Contents for case studies</i>	xxi
<i>Preface</i>	xxvii
<i>Guided tour</i>	xxxiv
<i>Publisher's acknowledgements</i>	xxxvi
1 The Software Process	1
Objectives	1
1.1 The nature of software development	2
1.1.1 The software development invariants	2
1.1.2 The software development "accidents"	4
1.1.2.1 Stakeholders	4
1.1.2.2 Process	6
1.1.2.2.1 Iterative and incremental process	6
1.1.2.2.2 Capability maturity model	7
1.1.2.2.3 The ISO 9000 family of quality standards	9
1.1.2.2.4 The ITIL framework	9
1.1.2.2.5 The COBIT framework	11
1.1.2.3 Modeling	12
1.1.2.3.1 Unified modeling language	13
1.1.2.3.2 CASE and process improvement	13
1.1.3 Development or integration?	14
Review quiz 1.1	15
1.2 System planning	15
1.2.1 The SWOT approach	16
1.2.2 The VCM approach	17
1.2.3 The BPR approach	19
1.2.4 The ISA approach	20
Review quiz 1.2	21
1.3 Systems for three management levels	22
1.3.1 Transactional processing systems	23
1.3.2 Analytical processing systems	24
1.3.3 Knowledge processing systems	25
Review quiz 1.3	26

1.4	The software development lifecycle	26
1.4.1	The development approach	27
1.4.1.1	The structured approach	27
1.4.1.2	The object-oriented approach	28
1.4.2	Lifecycle phases	30
1.4.2.1	Business analysis	30
1.4.2.1.1	Requirements determination	31
1.4.2.1.2	Requirements specification	31
1.4.2.2	System design	32
1.4.2.2.1	Architectural design	32
1.4.2.2.2	Detailed design	33
1.4.2.3	Implementation	33
1.4.2.4	Integration and deployment	34
1.4.2.5	Operation and maintenance	34
1.4.3	Activities spanning the lifecycle	35
1.4.3.1	Project planning	36
1.4.3.2	Metrics	37
1.4.3.3	Testing	37
	Review quiz 1.4	39
1.5	Development models and methods	39
1.5.1	The spiral model	40
1.5.2	IBM Rational Unified Process	41
1.5.3	Model-driven architecture	42
1.5.4	Agile software development	42
1.5.5	Aspect-oriented software development	45
	Review quiz 1.5	47
1.6	Problem statements for case studies	47
1.6.1	University enrolment	48
1.6.2	Video store	48
1.6.3	Contact management	49
1.6.4	Telemarketing	50
1.6.5	Advertising expenditure	51
1.6.6	Time logging	51
1.6.7	Currency converter	52
	Summary	53
	Key terms	54
	Multiple-choice test	56
	Questions	58
	Review quiz answers	60
	Multiple-choice test answers	60
	Answers to odd-numbered questions	61
2	Requirements Determination	66
	Objectives	66
2.1	From business processes to solution envisioning	67
2.1.1	Process hierarchy modeling	68

2.1.1.1	Processes and process decompositions	68
2.1.1.2	Process hierarchy diagram	68
2.1.2	Business process modeling	70
2.1.2.1	Flow objects, connecting objects, swimlanes and artifacts	70
2.1.2.2	Business process diagram	72
2.1.3	Solution envisioning	73
2.1.3.1	Solution envisioning process	74
2.1.3.2	Implementation strategies and capability architecture	76
	Review quiz 2.1	77
2.2	Requirements elicitation	77
2.2.1	System requirements	79
2.2.1.1	Functional requirements	79
2.2.1.2	Non-functional requirements	80
2.2.2	Traditional methods of requirements elicitation	81
2.2.2.1	Interviewing customers and domain experts	81
2.2.2.2	Questionnaires	83
2.2.2.3	Observation	84
2.2.2.4	Study of documents and software systems	84
2.2.3	Modern methods of requirements elicitation	85
2.2.3.1	Prototyping	85
2.2.3.2	Brainstorming	86
2.2.3.3	Joint application development	87
2.2.3.4	Rapid application development	88
	Review quiz 2.2	89
2.3	Requirements negotiation and validation	89
2.3.1	Out-of-scope requirements	89
2.3.2	Requirements dependency matrix	90
2.3.3	Requirements – risks and priorities	90
	Review quiz 2.3	91
2.4	Requirements management	91
2.4.1	Requirements identification and classification	92
2.4.2	Requirements hierarchies	93
2.4.3	Change management	93
2.4.4	Requirements traceability	94
	Review quiz 2.4	94
2.5	Requirements business model	94
2.5.1	System scope model	95
2.5.2	Business use case model	97
2.5.3	Business glossary	99
2.5.4	Business class model	101
	Review quiz 2.5	103
2.6	Requirements document	103
2.6.1	Document templates	103
2.6.2	Project preliminaries	104
2.6.3	System services	105

2.6.4	System constraints	105
2.6.5	Project matters	106
2.6.6	Appendices	107
	Review quiz 2.6	107
	Summary	107
	Key terms	108
	Multiple-choice test	110
	Questions	111
	Exercises: advertising expenditure	112
	Exercises: time logging	113
	Review quiz answers	113
	Multiple-choice test answers	114
	Answers to odd-numbered questions	115
	Solutions to exercises: AE	117
3	<i>Fundamentals of Visual Modeling</i>	122
	Objectives	122
3.1	The use case view	123
3.1.1	Actors	124
3.1.2	Use cases	124
3.1.3	Use case diagrams	126
3.1.4	Documenting use cases	127
	Review quiz 3.1	128
3.2	The activity view	128
3.2.1	Actions	130
3.2.2	Activity diagrams	130
	Review quiz 3.2	132
3.3	The structure view	133
3.3.1	Classes	133
3.3.2	Attributes	135
3.3.3	Associations	136
3.3.4	Aggregation	137
3.3.5	Generalization	138
3.3.6	Class diagrams	139
	Review quiz 3.3	140
3.4	The interaction view	141
3.4.1	Sequence diagrams	141
3.4.2	Communication diagrams	143
3.4.3	Class methods	144
	Review quiz 3.4	146
3.5	The state machine view	146
3.5.1	States and transitions	146
3.5.2	State machine diagrams	147
	Review quiz 3.5	149

3.6	The implementation view	149
3.6.1	Subsystems and packages	149
3.6.2	Components and component diagrams	150
3.6.3	Nodes and deployment diagrams	152
	Review quiz 3.6	153
	Summary	154
	Key terms	154
	Multiple-choice test	156
	Questions	157
	Exercises	158
	Exercises: video store	159
	Review quiz answers	161
	Multiple-choice test answers	161
	Answers to odd-numbered questions	162
	Solutions to odd-numbered exercises	163
	Solutions to odd-numbered exercises: video store	166
 4	 <i>Requirements Specification</i>	 169
	Objectives	169
4.1	Architectural prerogatives	170
4.1.1	Model-View-Controller	171
4.1.2	The Core J2EE architecture	172
4.1.3	Presentation-Controller-Bean-Mediator-Entity-Resource	173
4.1.3.1	The PCBMER layers	175
4.1.3.2	The PCBMER principles	176
	Review quiz 4.1	178
4.2	State specifications	178
4.2.1	Modeling classes 14	178
4.2.1.1	Discovering classes	179
4.2.1.1.1	Noun phrase approach	179
4.2.1.1.2	Common class pattern approach	180
4.2.1.1.3	Use case-driven approach	181
4.2.1.1.4	CRC approach	181
4.2.1.1.5	Mixed approach	182
4.2.1.1.6	Guidelines for discovering classes	182
4.2.1.1.7	Examples of discovering classes	183
4.2.1.2	Specifying classes	186
4.2.1.2.1	Naming classes	186
4.2.1.2.2	Discovering and specifying class attributes	186
4.2.1.2.3	Examples of discovering and specifying class attributes	187
4.2.2	Modeling associations	193
4.2.2.1	Discovering associations	193
4.2.2.2	Specifying associations	195
4.2.2.2.1	Example of specifying associations	195

4.2.3	Modeling aggregation and composition relationships	196
4.2.3.1	Discovering aggregations and composition	198
4.2.3.2	Specifying aggregations and compositions	199
4.2.3.3	Example of aggregation and composition specifications	199
4.2.4	Modeling generalization relationships	200
4.2.4.1	Discovering generalizations	201
4.2.4.2	Specifying generalization	201
4.2.4.3	Examples of generalization specifications	202
4.2.5	Modeling interfaces	202
4.2.5.1	Discovering interfaces	203
4.2.5.2	Specifying interfaces	204
4.2.5.3	Examples of interface specifications	204
4.2.6	Modeling objects	205
4.2.6.1	Specifying objects	205
4.2.6.2	Example of object specifications	205
	Review quiz 4.2	206
4.3	Behavior specifications	206
4.3.1	Modeling use cases	207
4.3.1.1	Specifying use cases	208
4.3.1.2	Examples of use case specifications	209
4.3.2	Modeling activities	213
4.3.2.1	Discovering actions	214
4.3.2.2	Specifying actions	214
4.3.2.3	Example of activity specifications	214
4.3.3	Modeling interactions	215
4.3.3.1	Discovering message sequences	216
4.3.3.2	Specifying message sequences	216
4.3.3.3	Examples of sequence specifications	216
4.3.4	Modeling operations	219
4.3.4.1	Discovering class operations	219
4.3.4.2	Specifying class operations	219
4.3.4.3	Examples of operation specifications	220
	Review quiz 4.3	221
4.4	State change specifications	221
4.4.1	Modeling object states	222
4.4.1.1	Discovering object states	222
4.4.1.2	Specifying object states	222
4.4.1.3	Example of state machine specifications	223
	Review quiz 4.4	223
	Summary	224
	Key terms	225
	Multiple-choice test	226
	Questions	228
	Exercises: video store	229
	Exercises: contact management	230

Exercises: university enrolment	231
Review quiz answers	232
Multiple-choice test answers	232
Answers to odd-numbered questions	233
Solutions to exercises: university enrolment	238
5 Moving from Analysis to Design	243
Objectives	243
5.1 Advanced class modeling	244
5.1.1 Extension mechanisms	244
5.1.1.1 Stereotypes	245
5.1.1.2 Comments and constraints	246
5.1.1.3 Tags	248
5.1.2 Visibility and encapsulation	250
5.1.2.1 Protected visibility	250
5.1.2.2 Accessibility of inherited class properties	251
5.1.2.3 Package and friendly visibility	254
5.1.3 Derived information	256
5.1.3.1 Derived attribute	256
5.1.3.2 Derived association	257
5.1.4 Qualified association	257
5.1.5 Association class versus reified class	258
5.1.5.1 Model with association class	259
5.1.5.2 Model with reified class	260
Review quiz 5.1	261
5.2 Advanced generalization and inheritance modeling	261
5.2.1 Generalization and substitutability	261
5.2.2 Inheritance versus encapsulation	262
5.2.3 Interface inheritance	262
5.2.4 Implementation inheritance	264
5.2.4.1 Proper use of implementation inheritance – extension inheritance	264
5.2.4.2 Problematic use of implementation inheritance – restriction inheritance	265
5.2.4.3 Improper use of implementation inheritance – convenience inheritance	266
5.2.4.4 The evils of implementation inheritance	267
5.2.4.4.1 Fragile base class	267
5.2.4.4.2 Overriding, down-calls and up-calls	268
5.2.4.4.3 Multiple implementation inheritance	270
Review quiz 5.2	271
5.3 Advanced aggregation and delegation modeling	271
5.3.1 Putting more semantics into aggregation	272
5.3.1.1 The <i>ExclusiveOwns</i> aggregation	272
5.3.1.2 The <i>Owns</i> aggregation	273

5.3.1.3	The <i>Has</i> aggregation	273
5.3.1.4	The <i>Member</i> aggregation	274
5.3.2	Aggregation as an alternative to generalization	275
5.3.2.1	Delegation and prototypical systems	276
5.3.2.2	Delegation versus inheritance	276
5.3.3	Aggregation and holons – some cerebral ammunition	277
	Review quiz 5.3	278
5.4	Advanced interaction modeling	278
5.4.1	Lifelines and messages	279
5.4.1.1	Accounting for basic technology	280
5.4.1.2	Visualizing technology contributions in interaction models	282
5.4.2	Fragments	284
5.4.3	Interaction uses	287
	Review quiz 5.4	288
	Summary	288
	Key terms	289
	Multiple-choice test	289
	Questions	290
	Exercises	291
	Exercises: time logging	292
	Exercises: advertising expenditure	293
	Review quiz answers	295
	Multiple-choice test answers	295
	Answers to odd-numbered questions	295
	Solutions to odd-numbered exercises	299
	Solutions to exercises: time logging	300
6	<i>System Architecture and Program Design</i>	304
	Objectives	304
6.1	Distributed physical architecture	305
6.1.1	Peer-to-peer architecture	306
6.1.2	Tiered architecture	307
6.1.3	Database-centered architecture	308
	Review quiz 6.1	309
6.2	Multilayer logical architecture	310
6.2.1	Architectural complexity	311
6.2.1.1	Spatial cognitive complexity	312
6.2.1.2	Structural complexity	313
6.2.1.2.1	Structural complexity of networks	313
6.2.1.2.2	Structural complexity of hierarchies	314
6.2.2	Architectural patterns	316
6.2.2.1	Façade	317
6.2.2.2	Abstract Factory	318
6.2.2.3	Chain of Responsibility	322
6.2.2.4	Observer	324

6.2.2.5 Mediator	327
Review quiz 6.2	331
6.3 Architectural modeling	331
6.3.1 Packages	331
6.3.2 Components	333
6.3.2.1 Component versus package	334
6.3.2.2 Component versus class and interface	335
6.3.3 Nodes	336
Review quiz 6.3	337
6.4 Principles of program design and reuse	338
6.4.1 Class cohesion and coupling	339
6.4.1.1 Kinds of class coupling	340
6.4.1.2 The Law of Demeter	340
6.4.1.3 Accessor methods and mindless classes	341
6.4.1.4 Dynamic classification and mixed-instance cohesion	344
6.4.2 Reuse strategy	348
6.4.2.1 Toolkit reuse	348
6.4.2.2 Framework reuse	348
6.4.2.3 Pattern reuse	349
Review quiz 6.4	349
6.5 Collaboration modeling	350
6.5.1 Collaboration	350
6.5.2 Composite structure	351
6.5.3 From use case to composite collaboration	353
6.5.4 From collaboration to interaction	357
6.5.5 From interaction to composite structure	362
Review quiz 6.5	362
Summary	362
Key terms	364
Multiple-choice test	366
Questions	367
Exercises: video store	368
Exercises: advertising expenditure	370
Review quiz answers	372
Multiple-choice test answers	372
Answers to odd-numbered questions	373
Solutions to exercises: advertising expenditure	376
 7 Graphical User Interface Design	 380
Objectives	380
7.1 Principles of GUI design	381
7.1.1 From GUI prototype to implementation	381
7.1.2 Guidelines for good GUI design	384
7.1.2.1 User in control	385
7.1.2.2 Consistency	385

- 7.1.2.3 Personalization and customization 386
 - 7.1.2.4 Forgiveness 387
 - 7.1.2.5 Feedback 387
 - 7.1.2.6 Aesthetics and usability 387
- Review quiz 7.1 388**
- 7.2 Desktop GUI design 388**
 - 7.2.1 Primary windows 388
 - 7.2.1.1 Row browsers 390
 - 7.2.1.2 Tree browsers 391
 - 7.2.2 Secondary windows 393
 - 7.2.2.1 Dialog boxes 393
 - 7.2.2.2 Tab folders 395
 - 7.2.2.3 Drop-down lists 395
 - 7.2.2.4 Message boxes 395
 - 7.2.3 Menus and toolbars 396
 - 7.2.4 Buttons and other controls 398
- Review quiz 7.2 399**
- 7.3 Web GUI design 400**
 - 7.3.1 Enabling technology for Web applications 400
 - 7.3.2 Content design 403
 - 7.3.2.1 Website to Web application continuum 403
 - 7.3.2.2 Forms 404
 - 7.3.3 Navigation design 409
 - 7.3.3.1 Menus and links 409
 - 7.3.3.2 Breadcrumbs and navigation panels 410
 - 7.3.3.3 Buttons 412
 - 7.3.4 Using GUI frameworks to leverage Web design 413
 - 7.3.4.1 The MVC dilemma 413
 - 7.3.4.2 Using Struts technology 415
- Review quiz 7.3 418**
- 7.4 Modeling GUI navigation 418**
 - 7.4.1 User experience storyboards 420
 - 7.4.2 Modeling UX elements 421
 - 7.4.3 Behavioral UX collaboration 422
 - 7.4.4 Structural UX collaboration 425
- Review quiz 7.4 427**
- Summary 427**
- Key terms 427**
- Multiple-choice test 428**
- Questions 429**
- Exercises: contact management 430**
- Exercises: telemarketing 432**
- Review quiz answers 434**
- Multiple-choice test answers 434**
- Answers to odd-numbered questions 434**
- Solutions to exercises: contact management 436**

8	<i>Persistence and Database Design</i>	441
	Objectives	441
8.1	Business objects and persistence	442
	8.1.1 Database management systems	442
	8.1.2 Levels of data models	443
	8.1.3 Integrating application and database modeling	444
	8.1.4 Underpinnings of object–database mapping	445
	Review quiz 8.1	446
8.2	Relational database model	446
	8.2.1 Columns, domains and rules	447
	8.2.2 Tables	448
	8.2.3 Referential integrity	450
	8.2.4 Triggers	452
	8.2.5 Stored procedures	453
	8.2.6 Views	454
	8.2.7 Normal forms	455
	Review quiz 8.2	456
8.3	Object-relational mapping	456
	8.3.1 Mapping entity classes	457
	8.3.2 Mapping associations	457
	8.3.3 Mapping aggregations	460
	8.3.4 Mapping generalizations	461
	Review quiz 8.3	464
8.4	Patterns for managing persistent objects	464
	8.4.1 Searching for persistent objects	465
	8.4.2 Loading persistent objects	467
	8.4.3 Unloading persistent objects	468
	Review quiz 8.4	468
8.5	Designing database access and transactions	469
	8.5.1 Levels of SQL programming	469
	8.5.2 Designing business transactions	471
	8.5.2.1 Short transactions	471
	8.5.2.1.1 Pessimistic concurrency control	471
	8.5.2.1.2 Levels of isolation	472
	8.5.2.1.3 Automatic recovery	473
	8.5.2.1.4 Programmable recovery	474
	8.5.2.1.5 Designing stored procedures and triggers	474
	8.5.2.2 Long transactions	475
	Review quiz 8.5	476
	Summary	476
	Key terms	477
	Multiple-choice tests	479
	Questions	480
	Exercises: contact management	480
	Exercises: telemarketing	481
	Review quiz answers	481

Multiple-choice answers	482
Answers to odd-numbered questions	482
Solutions to exercises: Contact management	484
9 Quality and Change Management	490
Objectives	490
9.1 Quality management	491
9.1.1 Quality assurance	492
9.1.1.1 Checklists, reviews and audits	492
9.1.1.2 Test-driven development	494
9.1.2 Quality control	495
9.1.2.1 Test concepts and techniques	496
9.1.2.2 Testing system services	498
9.1.2.3 Testing system constraints	500
9.1.2.3.1 Graphical user interface testing	501
9.1.2.3.2 Database testing	501
9.1.2.3.3 Authorization testing	502
9.1.2.3.4 Testing of other constraints	503
Review quiz 9.1	503
9.2 Change management	504
9.2.1 Instrumenting and managing change requests	505
9.2.1.1 Submitting a change request	506
9.2.1.2 Keeping track of change requests	507
9.2.2 Traceability	508
9.2.2.1 System features to use cases and use case requirements	509
9.2.2.2 Test plans to test cases and test requirements	510
9.2.2.3 UML diagrams to documents and requirements	510
9.2.2.4 Use case requirements to test requirements	511
9.2.2.5 Test requirements to defects	513
9.2.2.6 Use case requirements to enhancements	514
Review quiz 9.2	515
Summary	515
Key terms	515
Multiple-choice test	516
Questions	517
Review quiz answers	518
Multiple-choice test answers	518
Answers to odd-numbered questions	518
10 Tutorial-style Review and Reinforcement	521
Objectives	521
10.1 Use case modeling	522
10.1.1 Actors	522
10.1.2 Use cases	523
10.1.3 Use case diagram	524
10.1.4 Documenting use cases	525

10.2	Activity modeling	526
10.2.1	Actions	526
10.2.2	Activity diagram	528
10.3	Class modeling	529
10.3.1	Classes	529
10.3.2	Attributes	529
10.3.3	Associations	531
10.3.4	Aggregations	532
10.3.5	Generalizations	533
10.3.6	Class diagram	533
10.4	Interaction modeling	534
10.4.1	Sequence diagram	534
10.4.2	Communication diagram	537
10.5	State machine modeling	540
10.5.1	States and transitions	540
10.5.2	State machine diagram	541
10.6	Implementation models	542
10.6.1	Subsystems	542
10.6.2	Packages	543
10.6.3	Components	545
10.6.4	Notes	546
10.7	Object collaboration design	548
10.7.1	Use case design specifications	550
10.7.2	User interface prototyping	552
10.7.3	Sequence diagram	553
10.7.4	Design-level class diagram	555
10.8	Window navigation design	556
10.8.1	User experience (UX) elements	556
10.8.2	Behavioral UX collaboration	556
10.8.3	Structural UX collaboration	558
10.9	Database design	559
10.9.1	Object-relational mapping	559
10.9.2	Referential integrity design	559
	Summary	562
	Exercises: online shopping	562
Appendix:	Fundamentals of Object Technology	565
A.1	Real-life analogy	566
A.2	Instance object	566
A.2.1	Object notation	567
A.2.2	How do objects collaborate?	568
A.2.3	Identity and object communication	569
A.2.3.1	Persistent link	569
A.2.3.2	Transient link	570

A.2.3.3	Message passing	571
A.3	Class	571
A.3.1	Attribute	572
A.3.1.1	Attribute type that designates a class	572
A.3.1.2	Attribute visibility	573
A.3.2	Operation	574
A.3.2.1	Operations objects collaborate	574
A.3.2.2	Operation visibility and scope	575
A.3.3	Class object	575
A.4	Variables, methods and constructors	576
A.5	Association	578
A.5.1	Association degree	579
A.5.2	Association multiplicity	579
A.5.3	Association link and extent	580
A.5.4	Association class	581
A.6	Aggregation and composition	582
A.6.1	Buried reference	583
A.6.2	Inner class	583
A.6.3	Delegation	586
A.7	Generalization and inheritance	587
A.7.1	Polymorphism	588
A.7.2	Overriding versus overloading	589
A.7.3	Multiple inheritance	590
A.7.4	Multiple classification	590
A.7.5	Dynamic classification	591
A.8	Abstract class	592
A.9	Interface	593
A.9.1	Interface versus abstract class	593
A.9.2	Implementing an interface	594
A.9.3	Using an interface	594
	Summary	595
	Questions	597
	Answers to odd-numbered questions	598
	Bibliography	600
	Index	607