
Contents

Foreword.....	xxi
Preface	xxiii
Glossary of Acronyms	xxvii
Acknowledgements	xxix
Author.....	xxxix
Unique Features	xxxv
1 Software Engineering Fundamentals with Object Orientation	1
Learning Objectives.....	1
Introduction to Software Engineering	1
Learning and Adopting Software Engineering.....	2
Importance of Modeling.....	4
Software Engineering Fundamentals	5
Programs, Classes, Objects, and Data	5
The Six Fundamentals (Cornerstone) of Software Engineering.....	6
Classification (Grouping).....	7
Abstraction (Representing)	7
Encapsulation (Modularizing)	9
Association (Relating).....	10
Inheritance (Generalizing)	10
Polymorphism (Executing)	11
Software Engineering: A Historical Perspective	12
Evolution of Modeling	12
About the UML and Its Purpose	14
UML Usage	15
Common Errors in Interpreting Software Engineering Fundamentals and How to Rectify Them	16
Discussion Questions.....	17
Team Project Case Study	17
Endnotes.....	18
2 Review of 14 Unified Modeling Language Diagrams.....	19
Learning Objectives.....	19
List and Nature of UML Diagrams	19
Nature and Basics of UML Diagrams	22
Brief Review of UML Diagrams	22
Use Case Diagrams.....	22

Activity Diagrams.....	24
Class Diagrams.....	25
Sequence Diagrams.....	25
Interaction Overview Diagrams.....	26
Communication Diagrams.....	27
Object Diagrams.....	27
State Machine Diagram.....	28
Composite Structure Diagrams.....	30
Component Diagrams.....	30
Deployment Diagrams.....	31
Package Diagrams.....	31
Timing Diagrams.....	33
Profile Diagrams.....	33
Differences in List of UML Diagrams.....	34
Common Errors in Understanding UML Diagrams and How to Rectify Them.....	35
Discussion Questions.....	36
Team Project Case Study.....	36
Endnotes.....	37
3 Software Projects and Modeling Spaces: Package Diagrams.....	39
Learning Objectives.....	39
Understanding Different Types and Sizes of UML-Based Projects.....	39
Project Types and UML.....	39
Project Sizes and UML.....	41
Organizing the Project.....	41
Identifying Business Objectives.....	41
Dividing a Project into Smaller, Manageable Parts.....	42
Prioritization of Requirements.....	42
The Three Modeling Spaces in Software Engineering.....	44
Modeling of the Problem Space.....	44
Modeling of Solution Space.....	45
Modeling of Architectural Space.....	45
Mapping UML to Modeling Spaces.....	46
Package Diagrams.....	48
What Is a Package in UML?.....	48
Creating Package Diagrams.....	48
Namespaces.....	50
Strengths of Package Diagrams.....	50
Weaknesses of Package Diagrams.....	50
Common Errors in Organizing Project Packages and How to Rectify Them.....	51
Discussion Questions.....	52
Team Project Case Study.....	52
Endnotes.....	53
4 The Software Development Life Cycle and Agility.....	55
Learning Objectives.....	55
Process in Developing Software.....	55

UML and Process.....	56
Process Elements.....	56
Software Development Life Cycles.....	58
Iterative, Incremental, and Parallel Process in Software Development.....	59
Iterative.....	60
Incremental.....	60
Parallel.....	60
Time and Effort Distribution in Iterations.....	60
Agile in Software Development.....	62
The Agile Manifesto.....	62
Scrum—An Agile Approach.....	63
Roles, Ceremonies, and Artifacts.....	63
Roles.....	64
Ceremonies.....	65
Artifacts.....	65
Charts.....	67
Disciplined Agile Development.....	67
Composite Agile Method and Strategy.....	68
Common Errors in SDLC and Agile Use and How to Rectify Them.....	69
Discussion Questions.....	70
Team Project Case Study.....	71
Endnotes.....	71
5 Use Case Models-1: Actors and Use Cases.....	73
Learning Objectives.....	73
Use Case Modeling in the Problem Space.....	73
Actors.....	74
How to Find Actors?.....	74
Actor Variations.....	75
Primary versus Secondary Actors.....	75
Direct versus Indirect Actors.....	76
Abstract versus Concrete Actors.....	76
Clarifying Actor-Class Confusion.....	77
Actor Documentation.....	78
Actor Documentation for “A10-Patient”.....	79
Actor Documentation for “A60-Doctor”.....	80
Use Cases.....	81
What Is a Use Case?.....	81
Use Case Variations.....	81
Finding Use Cases.....	81
Use Case Documentation.....	82
Use Case Documentation Template.....	82
Example: Use Cases in the Hospital Management System.....	84
Brief Use Case Documentation for HMS.....	84
Detailed Use Case Documentation for HMS.....	87
Use Case “RegistersPatient”.....	88
Use Case “MaintainsCalendar”.....	89

Use Case "BooksConsultation"	90	"BooksConsultation" Activity Diagram	113
Use Case "PaysBill"	91	"PaysBill" Activity Diagram	115
Strengths and Weaknesses of Use Cases and Actors	92	Strengths and Weaknesses of Activity Diagrams	116
Strengths of Use Cases	92	Strengths of Activity Diagrams	116
Weaknesses of Use Cases	93	Weaknesses of Activity Diagrams	117
Relating Use Cases to Packages	93	Interaction Overview Diagram	118
Relating Use Cases to Functional Testing	94	Notations of an Interaction Overview Diagram	118
Common Errors in Modeling Actors and Use Cases and How to Rectify Them	94	Naming an Interaction Overview Diagram	118
Discussion Questions	95	Interaction Overview for "Consultation Details"	119
Team Project Case Study	96	Interaction Overview for "Accounting"	119
Endnotes	96	Strengths and Weaknesses of Interaction Overview Diagrams	120
6 Use Case Models-2: Use Case Diagrams and Requirements Modeling	97	Strengths of Interaction Overview Diagrams	120
Learning Objectives	97	Weaknesses of Interaction Overview Diagrams	120
Use Case Diagrams	97	Business Process Modeling	121
Notations of a Use Case Diagram	98	Common Errors in Activity Diagrams, Interaction Overview Diagrams, and	
Boundary	98	Business Process Models and How to Rectify Them	125
Notes	98	Discussion Questions	126
Actor	98	Team Project Case Study	126
Use Case	99	8 Class Models-1: Classes and Business Entities	127
Relationships	99	Learning Objectives	127
Use Case Relationships	99	Understanding Business Entities, Classes, and Objects	127
Include	99	Classes and Business Entities	128
Extends	100	Identifying and Naming Classes	129
Inherits (Generalize)	100	Class Identification by Use Case Analysis	129
Naming a Use Case Diagram	101	Class Identification by Sequence Diagrams	130
Use Case Diagrams for Hospital Management System	101	Naming a Class as a Business Entity	130
"Patient Maintenance" Use Case Diagram	101	Analyzing the "RegistersPatient" Use Case to Identify Classes/Business Entities	131
"Calendar Maintenance" Use Case Diagram	102	Class Definitions	134
"Consultation Details" Use Case Diagram	103	Class Documentation Template	134
"Accounting" Use Case Diagram	103	Documenting the Patient Class	135
Strengths and Weaknesses of Use Case Diagrams	105	Class Notation in UML	135
Strengths of Use Case Diagrams	105	Class Attributes	136
Weaknesses of Use Case Diagrams	106	Class Operations (Methods)	137
Common Errors in Use Case Diagrams and How to Rectify Them	106	Naming Conventions for Attributes and Operations	137
Discussion Questions	107	Visibilities on a Class	137
Team Project Case Study	108	Designing a Class in the Solution Space	138
Endnotes	108	Class Identification in Design (MOSS)	140
7 Activity Diagrams, Interaction Overview Diagrams, and Business Process Models	109	Strengths and Weaknesses of Classes	141
Learning Objectives	109	Strengths of Classes	141
Introduction	109	Weaknesses of Classes	141
Activity Diagrams	110	Common Errors in Classes and Business Entities and How to Rectify Them	142
Notations of Activity Diagrams	110	Discussion Questions	143
Naming an Activity Diagram	111	Team Project Case Study	143
Activity Diagrams for Hospital Management System	112	9 Class Model-2: Basic Class Diagram	145
"RegistersPatient" Activity Diagram	112	Learning Objectives	145
"MaintainsCalendar" Activity Diagram	113	Class Diagrams	145

Notations of Class Diagrams.....	145	Understanding Class Relationships.....	172
Inheritance Relationship in a Class Diagram.....	146	Notations on an Advanced Class Diagram in the Solution Space.....	172
Association Relationship in a Class Diagram.....	147	Class-to-Class Relationships.....	172
Aggregation Relationship in a Class Diagram.....	148	Advanced Relationships in a Class Diagram in Design.....	173
Multiplicities in Class Diagrams.....	149	Association Relationship in Design.....	174
Class Diagrams for Hospital Management System.....	150	Dependency Relationship in Design.....	175
“Patient Details” Class Diagram.....	150	Interface and Realization Relationship in Design.....	176
“Staff Details” Class Diagram.....	151	Aggregation Relationship in Design.....	177
“Consultation Details” Class Diagram.....	152	Implementing the Relationships: By References and By Value.....	178
“Accounting” Class Diagram.....	153	Parameter Visibility.....	179
Strengths of Class Diagrams.....	154	Multiplicities and Object Diagrams.....	180
Strengths and Advantages of Class Diagrams.....	154	Multiplicities in Design.....	180
Weaknesses of Class Diagrams.....	155	Object Diagrams Interpreting Multiplicities.....	180
Common Errors in Basic Class Diagram and How to Rectify Them.....	155	Collection Class and Multiplicities.....	181
Discussion Questions.....	156	Inheritance and Polymorphism in Design.....	182
Team Project Case Study.....	156	Incorporating Polymorphism in Design.....	182
10 UML’s Extensibility Mechanisms: Notes, Stereotypes, Constraints, and Tags.....	159	Multiple Inheritance.....	185
Learning Objectives.....	159	Incorporating Errors and Exceptions in Design.....	186
UML’s Extensibility Mechanisms.....	159	Attribute Identification, Naming, and Definition.....	187
Notes.....	160	Naming Attributes.....	187
Stereotypes.....	160	Discovering Attributes.....	188
Entity Class.....	163	Attribute (Data) Types.....	188
Boundary Class.....	163	Attribute Values.....	189
Control Class.....	163	Common Errors in Designing Attributes.....	189
Table Classes.....	163	Operation Identification, Naming, and Signature.....	190
Utility Classes.....	164	Understanding an Operation in a Class.....	190
User-Defined Classes.....	164	Naming Operations.....	191
Abstract Classes.....	164	Understanding Operation Signatures.....	191
Interfaces, Roles, and Types.....	164	Common Errors in Modeling Advance Class Designs and How to Rectify Them.....	192
Stereotypes for Attributes and Operations.....	165	Discussion Questions.....	192
Attribute Stereotypes.....	165	Team Project Case Study.....	193
Operation Types.....	165		
Manager Operations.....	165	12 Interaction Modeling with Sequence Diagrams.....	195
Implementer Operations.....	165	Learning Objectives.....	195
Access Operations.....	165	Interaction Modeling.....	195
Helping Operations.....	166	About Sequence Diagrams.....	195
Profile Diagram.....	166	Sequence Diagrams in Detail.....	196
Constraints.....	166	Notations on a Sequence Diagram.....	196
Tagged Value.....	166	Creating a Basic Sequence Diagram.....	198
Common Errors in UML’s Extensibility Mechanisms and How to Rectify Them.....	167	Relating Sequence Diagrams to Class Diagrams.....	198
Discussion Questions.....	168	Advancing Sequence Diagrams from Analysis to Design.....	199
Team Project Case Study.....	169	Understanding Focus of Control and Return Message.....	200
Endnote.....	169	Creating and Destroying an Object.....	201
11 Class Model-3: Advanced Class Designs.....	171	Sequence Diagrams in Hospital Management System.....	201
Learning Objectives.....	171	Sequence Diagrams in the Problem Space.....	201
Introduction.....	171	Design-Level Sequence Diagrams in the Solution Space.....	204
		Registering a Patient Sequence Diagram in Design.....	206

Updating a Calendar Sequence Diagram in Design	207
“Changing Booking Times” Sequence Diagram in Design	207
“Paying a Bill” Sequence Diagram in Design	207
Strengths and Weaknesses of Sequence Diagrams.....	209
Common Errors in Interaction Modeling with Sequence Diagrams and How to Rectify Them	210
Discussion Questions.....	211
Team Project Case Study	212
13 Database Modeling with Class and Sequence Diagrams.....	215
Learning Objectives.....	215
Introduction to Persistence	215
Persistence Mechanisms—Databases	216
Data Storage Mechanisms	216
Object-Oriented Databases	216
NoSQL Database	217
Relational Databases	217
Using Relational Databases in Object-Oriented Designs	218
Challenge of Storing Objects in Relational Tables.....	218
Mapping OO Classes to Relational Tables	219
Basic Persistence Functions (CRUD).....	220
Robustness in Persistence Design.....	221
Separating Persistence Operations from Business Logic	221
Robustness in Design Keeping Relational Storage and Objects Separate.....	222
Inheritance Relationship and Relational Tables	223
Mapping Associations in Relational Tables	224
Multiplicities, Association Class, and Link Table	225
Mapping Aggregation: Composition and Shared Aggregation.....	228
Shared Aggregation and Reference Table.....	228
Persistence in Practice for HMS.....	229
Persistence Design for Patient-Related Classes.....	229
Additional Example of Persistence Design in HMS.....	230
Incorporating Database Interface Pattern in HMS Persistence Design.....	230
Common Errors in Interpreting Database Modeling and How to Rectify Them	232
Discussion Questions.....	232
Team Project Case Study	233
Endnote	233
14 Dynamic Modeling with State Machine Diagrams.....	235
Learning Objectives.....	235
Introduction to Dynamic Modeling with State Machine Diagrams.....	235
State Machine Diagrams for Dynamic Modeling	236
Notations of State Machine Diagrams.....	236
State Machine Diagrams for Patient Object in Problem Space.....	237
“Patient” State Machine Diagram	239
“Consultation” State Machine Diagram	239
“Bill Payment” State Machine Diagram	239
Advanced State Machine Diagram for Patient Object in HMS in Solution Space.....	239
State Machine Diagram for “Patient” in HMS.....	240
State Machine Diagram for “Patient_Form,” Boundary Object in HMS	242
State Machine Diagram for “ConsultationManager,” a Control Object in HMS	243
Steps in Building a State Machine Diagram	244
Common Errors in Modeling State Machine Diagrams and How to Rectify Them	245
Discussion Questions.....	246
Team Project Case Study	246
Endnote	247
15 Advanced Software Engineering Design Concepts: Reuse, Granularity, Patterns, and Robustness	249
Learning Objectives.....	249
Introduction	249
Reusability in Software Engineering.....	250
Levels of Reuse.....	250
Code-Level Reuse.....	250
Design-Level Reuse	251
Analysis-Level Reuse	251
Reuse Strategies in Software Projects	251
Encapsulation Facilitates Reuse.....	252
Reuse as a Culture.....	252
Generalization versus Specialization in Reuse	253
Granularity in Object-Oriented Design.....	253
Design Patterns in Software Design Engineering.....	254
What Are Patterns?	254
Origins of Patterns	255
Structure of a Pattern	255
Using Patterns in the Solution and Architectural Modeling Spaces.....	255
Robustness in Design.....	257
Dependencies of Classes	257
Identifying Lack of Robustness	257
Rules of Robustness.....	258
Incorporating Robustness in Design	258
System Architecture and Design Process.....	260
Common Errors in Reuse, Granularity, Patterns, and Robustness and How to Rectify Them	262
Discussion Questions.....	263
Team Project Case Study	264
Endnotes.....	264
16 Interface Specifications: Prototyping.....	265
Learning Objectives.....	265
Introduction to Interfaces	265
Specifying Interface Requirements	266
Interface Specifications for HMS	267
User Interface Specifications for HMS	267

Printer Interface Specifications for HMS.....	268
External System Interfaces for HMS	269
Examples of User Interface Designs for HMS (Initial Iteration)	269
Specifying the Flow of User Interfaces (HMS Example)	270
Mobile Applications Interfaces.....	273
Printer Interfaces.....	273
User Interface Design Considerations.....	276
Organizing Interface Classes	276
Usability in GUI Design	277
User Categories in GUI Design.....	278
Prototyping.....	279
Functional Prototype	279
Technical Prototype	280
Architectural Prototype.....	280
Prototyping and Quality	280
Common Errors in Interface Specifications and Prototyping and How to Rectify Them.....	281
Discussion Questions.....	281
Team Project Case Study	282
Endnotes.....	282
17 Implementation Modeling with Component, Deployment, and Composite Structure Diagrams.....	285
Learning Objectives.....	285
Introduction	285
Component Diagrams	286
Understanding a Component.....	286
Relevance of Component-Based Software Development	286
Types of Components.....	287
Representing Components with UML	287
Component Characteristics and Types	288
Component Diagrams for HMS	289
Practical Component Diagram Showing Interdependencies and Packages for HMS	289
Strengths and Weaknesses of Component Diagram.....	290
Composite Structure Diagram.....	291
Deployment Diagrams.....	292
UML Notations on a Deployment Diagram	292
Process Around Implementation Diagrams.....	295
Common Errors in Implementation Modeling with Component, Deployment, and Composite Structure Diagrams and How to Rectify Them	296
Discussion Questions.....	296
Team Project Case Study	297
Endnote	297
18 Quality of UML Models with Syntax, Semantic, and Aesthetic Checks.....	299
Learning Objectives.....	299
Introduction	299
Quality Management, Assurance, and Control (Testing)	300
Quality Assurance and Model Quality.....	300
Verification and Validation.....	301
Syntax, Semantics, and Aesthetics Verify and Validate Artifacts, Diagrams, and Models.....	301
Application of Syntax, Semantics, and Aesthetics to UML Notations	302
Quality Models—Syntax	302
Quality Models—Semantics	303
Quality Models—Aesthetics	304
Quality Techniques and V&V Checks.....	304
Levels of Syntax, Semantics, and Aesthetics as Applied to UML-Based Diagrams	305
Syntactic Checks and UML Elements (Focus on Correctness)	306
Semantic Checks and UML Diagrams (Focus on Completeness and Consistency) ..	306
Aesthetic Checks and UML Models (Focus on Symmetry and Consistency)	307
Common Errors in Quality Assurance and Testing of UML Models and How to Rectify Them	307
Discussion Questions.....	308
Team Project Case Study	309
Endnotes.....	309
19 Software Testing: Plan, Design, and Execute.....	311
Learning Objectives.....	311
Introduction	311
Testing Needs in a Project.....	311
Various Types of Testing	312
Test Strategy Influencing Factors.....	313
Organizing the Testing of Software	314
Test Planning	314
Traceability Matrix.....	315
Use-Case-Based versus Class-Based Test Design	316
Test Approaches	317
Visibility of Testing—Black Box versus White Box Testing	317
Automation of Testing—Manual versus Automated	317
Slicing of Tests—Vertical (Functional) or Horizontal (Technical).....	318
Partitioning of Data—Equivalence Partition and Boundary Values.....	318
Test Architecture	318
Test Designs.....	319
Test Designs in Solution Space.....	319
Test Design Format	319
Test Designs for Components	320
Reusability in Test Designs	320
Test Cases in Solution Space.....	321
Test Case Format	321
Test Data.....	322
Masking and Blending of Test Data	322
Acceptance Test Cases for Hospital Management System	322
Test Case for “RegistersPatient”	323

Test Case for "MaintainsCalendar"	324
Test Case for "BooksConsultation"	324
Test Case for "PaysBill"	325
Test Case for "PaysBillOnInternet"	326
Test Case for "CashChequePayment"	327
Class-Based Approach to Test Cases in the Solution Space	327
Test Harnesses	327
Verifying Test Cases	328
Operational (NFR) Testing	329
Some Operational Tests	329
Common Errors in Testing in Solution Space and How to Rectify Them	330
Discussion Questions	331
Team Project Case Study	331
Endnotes	332
20 Nonfunctional (Operational) Requirements Specification and Application	333
Learning Objectives	333
Nonfunctional (Operational) Requirements	333
NFRs and UML	334
Source of NFRs	334
Types of Nonfunctional Parameters	335
Composite Agile Method and Strategy and Prototyping for NFRs	337
NFR Categories: Qualities and Constraints	337
NFR Challenges	338
Capturing NFRs in CAMS	339
NFR Levels	340
Performance	341
Response Times and Performance	341
Outsourced Projects and Performance	342
Bandwidth	343
Scalability	343
Scalability and Hardware	344
HMS Example of Scalability Requirement	344
Volume	344
Operating System	344
Mobile OS	345
Accessibility	345
Reliability and Maintenance	346
Environment	346
Legal and Compliance	347
Security	347
Usability and User Experience	349
Applying Usability Requirements to Software Solutions	349
Designing to Prevent Errors	350
Big Data (Velocity, Variety)	350
Cloud	350
Common Errors in Handling NFRs and How to Rectify Them	351
Discussion Questions	352
Team Project Case Study	353
Endnotes	353
21 Emerging Information Technologies and Modeling	355
Learning Objectives	355
Emerging Information Technologies and Modeling	355
SMAC Significance	356
Service Orientation (Analytics, Utilities)	357
Internet of Things	357
Mobile and Social Media Applications	358
Cloud Integration	358
Virtual and Augmented Reality	359
Robotics and Machine Learning	359
Modeling the Not Only SQL Databases	359
Service Orientation Based on the Cloud	360
Designing with Services	361
Core Elements of Web Services	362
XML/SOAP	362
Web Services Description Language	362
Universal Description, Discovery, and Integration	363
Web Services and Modeling	363
Modeling and Usage of Web Services	363
Web Service Metamodels and Dynamics	364
Model-Driven Architecture and Web Services	365
Executable UML	365
Discussion Questions	366
Team Project Case Study	366
Endnotes	366
Appendix A: Case Study Problem Statements for Team Projects	369
Bibliography	375
Index	379

Note: Appendix A, including additional case studies, Appendix B: Mid-Term, and Appendix C: Final Exam are available online at <https://www.crcpress.com/9781138297432>.