

UNIT I

Chapter 1 : Introduction to OOM and UML 1-1 to 1-25

Syllabus : Introduction to Object Oriented Methodology
 : Study of various design methodologies like Object Oriented Design by Booch, Object Modeling Techniques by Rumbaugh, Object-Oriented Analysis by Codd Yourdon and Object-Oriented Software Engineering by Ivar Jacobson

Unified Approach : Unification of Booch, Rumbaugh and Jacobson methodologies, Object - Oriented Analysis, Object Oriented Design, Iterative Development & Continuous Testing, Modelling based on UML , Layered Approach

Unified Modeling Language : Introduction to Modeling and UML2.0, MDA, UML2.0 Structure, UML Building Blocks, UML common Mechanisms, Introduction to all UML2.0 Diagram notational Techniques, 4+1View

1.1 Introduction 1-1

1.2 Object Oriented Methodology..... 1-2

1.2.1 Object Oriented Design –Booch 1-2

1.2.2 Object Modeling Techniques – Rumbaugh..... 1-3

1.2.3 Object – Oriented Analysis - Cood Yourdon 1-5

1.2.4 Object – Oriented Software Engineering – Ivar Jacobson 1-5

1.2.4(A) Object Oriented Business Engineering 1-5

1.2.4(B) Object-Oriented Software Engineering 1-6

1.2.5 Object Oriented Analysis and Design Methodology 1-6

1.3 Unified Approach..... 1-7

1.3.1 Object Oriented Analysis..... 1-8

1.3.2 Object-Oriented Design..... 1-8

1.3.3 Iterative Development and Continuous Testing..... 1-8

1.3.4 Modeling based on the Unified Modeling Language.....1-8

1.3.5 Layered Approach to Software Development1-9

1.4 Unified Modeling Language..... 1-10

1.4.1 Introduction to Unified Modelling Language1-10

1.4.2 UML Supports Requirements Modelling..... 1-11

1.4.3 Model Driven Architecture1-12

1.4.4 UML2.0 Structure1-12

1.4.5 Building Blocks of the UML..... 1-13

1.4.5(A) Things in UML1-13

1.4.5(B) Relationship in UML..... 1-16

1.4.5(C) Diagrams in UML1-16

1.4.6 UML Common Mechanisms1-17

1.4.7 Introduction to all UML2.0 Diagram Notational Techniques..... 1-19

1.4.8 4+1 View of UML Diagrams1-24

UNIT II

Chapter 2 : Object Oriented Analysis 2-1 to 2-44

Syllabus : Object Oriented Analysis Process : Use Case Modeling: Actor Identification, Actor Classification, Actor Generalization, Use Case Identification, Uses/Include/Extend Association, Writing a formal use case, Forward Engineering (Use case realization)

Class Modeling : Approach for identifying class, Approaches for identifying classes, Class pattern approach, Class Responsibilities, Collaboration Approach, Naming Classes, Class associations Generalization specialization relationship, Aggregation and Composition Relationships



<p>2.1 Object Oriented Analysis Process..... 2-1</p> <p>2.2 Use Case Modeling..... 2-2</p> <p>2.2.1 Use Case Diagram 2-2</p> <p>2.2.2 Actor and Use Case Identification 2-2</p> <p>2.2.3 Use Case Relationship..... 2-5</p> <p>2.2.3(A) Include..... 2-5</p> <p>2.2.3(B) Extend 2-6</p> <p>2.2.3 (C) Generalizations..... 2-7</p> <p>2.2.4 Writing a formal use case 2-8</p> <p>2.2.5 Forward Engineering (Use case realization)2-11</p> <p>2.2.6 Solved Examples2-11</p> <p>2.3 Class Modelling..... 2-15</p> <p>2.3.1 Class Structure2-15</p> <p>2.4 Method of Identifying Classes..... 2-16</p> <p>2.4.1 Common Class Pattern2-16</p> <p>2.4.2 CRC (Class Responsibilities Collaboration)2-17</p> <p>2.4.3 Use of Noun Verb Analysis2-18</p> <p>2.4.4 Identifying Boundary , Entity and Control Classes.....2-19</p> <p>2.4.5 Boundary Class2-19</p> <p>2.4.6 Control Classes.....2-20</p> <p>2.4.7 Entity Classes.....2-21</p> <p>2.5 Class Diagram..... 2-22</p> <p>2.5.1 Class Structure2-22</p> <p>2.5.2 Identifying Attributes2-24</p> <p>2.5.3 Identifying Operation2-24</p> <p>2.5.4 Relationship between Classes.....2-25</p> <p>2.5.4(A) Association.....2-26</p> <p>2.5.4(B) Generalization/Specialization</p>	<p>Relationship.....2-29</p> <p>2.5.4(C) Aggregation and Composition Relationships.....2-29</p> <p>2.5.4(D) Composite Relationships2-30</p> <p>2.5.4(E) Difference between Aggregation and Composition.....2-31</p> <p>2.5.5 Solved Examples2-32</p> <p>2.6 Object Diagram2-40</p> <p>2.7 Package Diagram2-42</p> <div style="text-align: center; border: 1px solid black; padding: 5px; width: fit-content; margin: 20px auto;">UNIT III</div> <hr/> <p>Chapter 3 : Interaction and Behavior Modeling 3-1 to 3-33</p> <div style="border: 1px solid black; padding: 10px; margin-top: 10px;"> <p>Syllabus : Activity Diagram : Activity and Actions, Activity Edge, Decision and Merge Points, Fork-Join, Control Flow, Constraints on Action, Swim Lanes.</p> <p>Sequence Diagram : Context, Objects and Roles, Links, Object Life Line, Message or stimulus, Activation/Focus of Control, delete object, Modelling Interactions.</p> <p>Collaboration Diagram : Objects and Links, Messages and stimuli, Active Objects, Communication Diagram, Iteration Expression, Parallel Execution, Guard Expression, Timing Diagram.</p> <p>State Diagram : State Machine, Triggers and Ports, Transitions and conditions, Initial and Final State, nested state, Composite States, Submachine States.</p> </div> <p>3.1 Overview of Interaction Diagram 3-1</p> <p>3.2 Activity Diagram..... 3-2</p> <p>3.2.1 Component of Activity Diagram3-3</p> <p>3.2.1(A) Activity and Actions3-3</p> <p>3.2.1(B) Initial and Final Activity3-3</p> <p>3.2.1(C) Activity Edge.....3-4</p>
---	--

3.2.1(D) Decision and Merge Points 3-4

3.2.1(E) Forking and Joining..... 3-4

3.2.1(F) Input and Output Pins 3-5

3.2.1(G) Activity Partitions..... 3-5

3.2.1(H) Constraints on Action 3-5

3.2.1(I) SwimLanes..... 3-6

3.2.1(J) Creating the Diagram..... 3-7

3.3 Sequence Diagram 3-11

3.3.1 Elements of Sequence Diagram.....3-12

3.3.1(A) Objects and Roles.....3-12

3.3.1(B) Link.....3-12

3.3.1(C) Object Life Line3-12

3.3.1(D) Message or Stimulus.....3-12

3.3.1(E) Focus of Control.....3-14

3.3.1(F) End of a Lifeline3-14

3.3.1(G) Conditional Message3-14

3.3.2 Creating Sequence Diagram.....3-14

3.3.3 Modeling Interactions3-14

3.4 Collaboration Diagram/ Communication Diagram 3-17

3.4.1 Elements of Collaboration Diagram.....3-18

3.4.2 Creating Collaboration Diagram.....3-18

3.4.3 Sequence and Collaboration Diagrams.....3-19

3.4.4 Communication Diagram3-20

3.4.4(A) Parallel Execution.....3-20

3.4.4(B) Iteration Expression.....3-20

3.4.4(C) Guard Expression3-21

3.4.5 Timing Diagram.....3-21

3.5 State Diagrams 3-23

3.5.1 Element of State Diagram3-24

3.5.1(A) State Machine3-24

3.5.1(B) Initial and Final State.....3-26

3.5.1(C) Triggers and Ports3-26

3.5.1(D) Transitions3-26

3.5.1(E) Composite States3-27

3.5.1(F) Submachine States.....3-28

UNIT IV

Chapter 4 : Object Oriented Design Process 4-1 to 4-32

Syllabus : Object Oriented Design Process : Designing Business Layer: Object Oriented Constraints Language (OCL), Designing Business Classes: The Process, Designing Well Defined Class Visibility, Attribute Refinement, Method Design Using UML Activity Diagram, Packaging and Managing Classes.

Designing Access Layer : Object Relational Systems, Object Relation Mapping, Table Class Mapping, Table - Inherited Classes Mapping, Designing the Access Layer Classes: create mirror classes, identify access layer class relationships, eliminate redundant classes, create method classes.

Designing View Layer : View Layer Classes Design, Identifying View Classes by Analyzing Use Cases, Macro-Level Design Process – identify view layer objects, and build prototype for view layer Interface.

Test Usability and User satisfaction : Component and Deployment Design using Component and Deployment Diagram.

4.1 Introduction..... 4-1

4.2 Object Oriented Design Process 4-2

4.3 Designing Business Layer 4-4

4.3.1 Object Oriented Constraints Language 4-4

4.4 Designing Business Classes 4-5



4.4.1	The Process	4-5	4.9.1	Elements of a Deployment Diagram	4-21
4.4.2	Designing Well Defined Class Visibility	4-5	4.9.2	When to use Deployment Diagram	4-22
4.4.3	Attribute Refinement,.....	4-5	UNIT V		
4.4.4	Method Design Using UML Activity Diagram.....	4-6	Chapter 5 : Software Design Principles and Patterns		
4.4.5	Packaging and Managing Classes	4-6	5-1 to 5-32		
4.5	Designing Access Layer	4-7	Syllabus : Introduction and need of Design Principles : General Responsibility Assignment Software Patterns (GRASP): Introduction, Creator, Information Expert, Low coupling, Controller, High Cohesion, Polymorphism, Pure fabrication, Indirection, Protected Variations. Introduction to GOF design patterns : Types of design patterns: Creational Pattern : Singleton, Factory		
4.5.1	Object Relational Systems	4-7	Structural Pattern : Adapter, Façade Behavioral Patterns: Strategy, State		
4.5.2	Table Class Mapping.....	4-9	5.1	Introduction and Need of Design Principles.....	5-1
4.5.3	Object Relation Mapping.....	4-9	5.2	General Responsibility Assignment Software Patterns (GRASP)	5-5
4.5.4	Table - Inherited Classes Mapping.....	4-11	5.2.1	Creator.....	5-6
4.5.5	Design Access Layer Classes : The Process.....	4-13	5.2.2	Information Expert.....	5-7
4.6	Designing View Layer	4-13	5.2.3	Low Coupling.....	5-8
4.6.1	View Layer Classes Design	4-14	5.2.4	Controller	5-8
4.6.2	Identifying View Classes by Analyzing Use Cases	4-14	5.2.5	High Cohesion	5-9
4.6.3	Macro-Level Design Process.....	4-14	5.2.6	Polymorphism.....	5-9
4.6.4	Macro Level Design Process	4-15	5.2.7	Pure Fabrication	5-10
4.6.5	Micro Development Process.....	4-16	5.2.8	Indirection.....	5-10
4.6.6	Build Prototype for View Layer Interface	4-16	5.2.9	Protected Variations.....	5-11
4.7	Test Usability and User satisfaction	4-16	5.3	Introduction to GOF Design Patterns	5-11
4.7.1	Principles to Support Usability.....	4-17	5.3.1	What is Design Patten ?	5-11
4.7.2	User Interface Design Process	4-17	5.3.2	Why to Use Design Patten?	5-12
4.8	Component Diagram	4-18			
4.8.1	Types of Components / Elements of Component Diagram	4-19			
4.8.2	Interface.....	4-20			
4.9	Deployment Diagram.....	4-21			

<p>5.3.3 Design Patten Template.....5-13</p> <p>5.4 Types of Design Patterns 5-13</p> <p>5.5 Creational Pattern : Singleton, Factory 5-16</p> <p>5.5.1 Singleton Pattern5-17</p> <p>5.5.2 Factory Pattern.....5-18</p> <p>5.6 Structural Pattern: Adapter, Façade..... 5-21</p> <p>5.6.1 Adapter Pattern.....5-21</p> <p>5.6.2 Facade Pattern5-24</p> <p>5.7 Behavioral Patterns : Strategy, State..... 5-27</p> <p>5.7.1 Strategy5-27</p> <p>5.7.2 State5-29</p> <p>5.7.3 State Design Pattern Benefits.....5-32</p>	<p>6.4.3(B) Table Class Mapping.....6-11</p> <p>6.4.3(C) Object Relation Mapping.....6-11</p> <p>6.4.3(D) Table – Inherited Classes Mapping6-13</p> <p>6.5 Designing Client / Server Software Architectures6-14</p> <p>6.5.1 Client/ Service Software Architectural Structure Patterns.....6-15</p> <p>6.5.2 Architectural Communication Patterns for Client/Server Architectures6-16</p> <p>6.6 Designing Service Oriented Software Architectures6-19</p> <p>6.6.1 What is SOA ?.....6-19</p> <p>6.6.2 Design Principles for Services6-20</p> <p>6.6.3 Technology Support for Service-Oriented Architecture6-21</p> <p>6.6.4 SOAP Protocol Stack.....6-22</p> <p>6.6.5 SOA Patterns.....6-23</p> <p>6.6.6 Software Architectural Transaction Patterns.....6-24</p> <p>6.6.7 Service Coordination in Service-oriented Architecture6-24</p> <p>6.7 Designing Component Based Software Architectures6-25</p> <p>6.7.1 Principles of Component-Based Design.....6-25</p> <p>6.7.2 Advantages of Component-Based Architecture6-26</p> <p>6.7.3 Modelling Components with UML6-26</p> <p>6.7.4 Group Message Communication Patterns.....6-27</p> <p>6.8 Designing Real-Time Software Architectures6-28</p>
---	---

UNIT VI

Chapter 6 : Software Architectural Design 6-1 to 6-33

Syllabus : Anatomy of Software Architecture, Quality attributes in architecture design, Designing Object-Oriented Software Architecture, Designing Client/Server Software Architecture, Designing Service-Oriented Architectures, Designing Component-Based Software Architectures, Designing Concurrent and Real-Time Software Architectures. Product Line Architecture design

<p>6.1 Anatomy of Software Architecture 6-1</p> <p>6.2 Overview of Software Architecture 6-2</p> <p>6.3 Quality Attributes in Architecture Design 6-3</p> <p>6.4 Object-Oriented Architecture 6-4</p> <p>6.4.1 Object Oriented Analysis..... 6-5</p> <p>6.4.2 Object-Oriented Design..... 6-7</p> <p>6.4.3 From Static Models to Relational Database Design 6-9</p> <p>6.4.3(A) Object Relational Systems 6-9</p>
--



6.8.1	Real Time System Characteristics.....6-28	6.9	Product Line Architecture design.....6-31
6.8.2	Tasks in Real Time systems6-29	6.9.1	Object-Oriented Software Life Cycle for
6.8.3	Design Pattern for Real-Time Software		Software Product Lines6-31
	Architectures6-30	➤	Case Studies C-1 to C-65
6.8.4	Control Patterns for Real-Time Software		
	Architectures6-31		
