

GUJARAT TECHNOLOGICAL UNIVERSITY

Master in Computer Application (Integrated MCA)

Year II – (Semester-IV) (W.E.F. January 2015)

Subject Name: UML & Object-oriented Modeling

Subject Code: 4440604

1. Learning Objectives:

In this syllabus, students will learn to apply the Unified Modelling Language (UML) to fundamental object-oriented analysis and design concepts.

Object Oriented Analysis and Design Using UML present the concepts and techniques necessary to effectively use system requirements to drive the development of a robust design model, It includes architecture, objects, classes, components, stereotypes, relationships and all supporting diagrams.

The UML is used throughout the project lifecycle to capture and communicate analysis and design decisions. Students will understand OOAD Concepts, learn to represent it with UML and document it using UML modelling tool. The Session will be concept & Case study driven and wherever necessary tool will be used.

2. Prerequisites:

Knowledge of any Object Oriented Programming Language and System Development Life Cycle.

3. Contents:

This course uses the industry-standard Unified Modeling Language (UML) as a means of depicting OO software design and providing team members with a common notation and vocabulary for communicating their ideas. This course will teach students the basic concepts of the object-oriented paradigm using the Unified Modeling Language (UML). Students will be introduced to concepts such as abstraction, polymorphism and encapsulation, and will learn how to use these concepts in object-oriented software development.

Unit No.	Course Content	No. of Lectures
I	Why We Model?: The importance of modelling, principles of modelling, Introduction of UML: Overview, Conceptual Model of UML, Software Architecture, Software Development Life Cycle, Classes, Relationships, Common Mechanisms of UML.	08
II	Class Diagrams: Terms and Concepts, Common Modeling Techniques, Advanced Classes, Advanced Relationships, Interfaces, Types and Roles, Packages	10
III	Instances, Object Diagrams, Basic Behavioral Modeling: Interactions, Use cases, Use Case Diagrams, Interaction Diagrams, Activity Diagrams	10
IV	Advanced Behavioral Modeling: Events and Signals, State Machines, State Diagrams, Architectural Modeling: Components, Deployment,	10

	Collaborations, Component Diagrams, Deployment Diagrams,	
V	Generate Use-case Diagram, Class Diagram, Sequence Diagram, Collaboration Diagram, Activity Diagram, State Chart Diagram, Component Diagram, Deployment Diagram for the following systems. ▪ Student Registration System ▪ Courier Tracking System ▪ Online Shopping System ▪ Online Pizza ordering System ▪ Online Job Portal System	12

Total Lectures: 50

4. Reference Book(s):

The Unified Modeling Language User Guide By Grady Booch, James Rumbaugh, Ivar Jacobson Publisher Pearson Education

5. Suggested Additional Reading:

- a. UML 2 Bible by Tom Pender Publisher Wiley-dreamtech
- b. UML 2 and the Unified Process Practical Object-Oriented Analysis and Design Second Edition by Jim Arlow, LLa Neustadt Publisher Pearson Education
- c. Web reference: By Object Management Group (OMG) <http://www.uml.org/>

6. Chapter wise Coverage from Main Reference Book(s):

1	Unit – I	Chapter: 1, 2, 4, 5, 6
2	Unit – II	Chapter: 7, 8, 9, 10, 11, 12
3	Unit – III	Chapter: 13, 14, 15, 16, 17, 18, 19
4	Unit – IV	Chapter: 20, 21, 24, 25, 26, 27, 29, 30
5	Unit – V	Practical implementation of all diagram in “Dia” (Diagramming Software) in terms of Case Studies

UML Diagram Tool:

Dia (diagramming software):

Dia is free and open source general-purpose diagramming software, developed originally by Alexander Larsson. Dia uses a controlled single document interface (SDI) similar to GIMP and Inkscape.

It can be downloaded from http://sourceforge.net/projects/dia-installer/?source=typ_redirect

7. Accomplishment of the student after completing the course:

After successful completion of this course the students will be able to discriminate what the UML is, what it is not, and why the UML is relevant to the process of developing software-intensive systems. They will be master the vocabulary, rules and idioms of the UML and, in general will be able to use the language effectively in System Development process. They will be able to understand how to apply the UML to solve a number of common modelling problems.