

FACULTY OF COMPUTER APPLICATIONS

-
- **Course** : BCA
 - **Sem.** : 5
 - **Subject Code** : 05BC0503
 - **Subject** : **Object Oriented Analysis and Design**
 - **Objective** : Analysis and Design of Software Systems is a practical field that relies on a core set of standard concepts and principles. The objective of this course is to teach the students standard and tested techniques widely embraced by experienced analysts plus new and emerging tools and techniques that recent software professionals are expected to apply in the field. The students should be able to work with the standard object oriented software development methodology using UML specific rules and standards. The course is meant to give balanced exposure to both traditional and object oriented approaches to system analysis and design.
 - **Prerequisites** : Understanding of Object oriented languages and concepts

Unit No	Topics Covered	No of lectures required
1	Introduction <ul style="list-style-type: none"> ○ What is object orientation? What is OO Development? OO Themes, Usefulness of OO development ○ Modeling as design concept ○ Modeling, abstraction, The three models, Introduction to UML 	6
2	Structural modeling <ul style="list-style-type: none"> ○ Class modeling : Object and class concepts, links and association, generalization and inheritance, 	10

FACULTY OF COMPUTER APPLICATIONS

	<p>navigation of class models</p> <ul style="list-style-type: none"> ○ Advance class modeling : Advance object and class concepts, association ends, N-ary associations, Aggregation, abstract classes, multiple inheritance, metadata, reification, constraints, derived data, packages ○ Object diagrams 	
3	<p>Behavioral modeling</p> <ul style="list-style-type: none"> ○ Interactions, Use cases, Use case diagrams, Interaction diagrams : sequence diagram, collaboration diagram, Activity diagrams ○ State modeling : Events, states, transitions and conditions, state diagrams, state diagram behavior 	12
4	<p>Analysis</p> <ul style="list-style-type: none"> ○ Domain analysis : Overview, domain class model, domain state model, domain interaction model, iterating the analysis ○ Application analysis : Application interaction model, application class model, application state model, adding operations ○ ATM example 	10
5	<p>Design</p> <ul style="list-style-type: none"> ○ System design: Overview, performance, reuse plan, breaking system into subsystems, concurrency, managing data storage, handling global resources, software control strategy, boundary conditions, trade-off priorities, architectural styles, ○ Class design : Overview, bridging the gap, realizing use cases, designing algorithms, recursing downward, refactoring, design optimization, reification of behavior, adjusting inheritance, organizing class design ○ ATM example 	10

FACULTY OF COMPUTER APPLICATIONS

▪ **Course Outcomes**

- 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 modeling problems.

▪ **Main References:**

- Object Oriented Modeling and Design using UML, Michael Blaha and James Rumbaugh, Pearson Education, 2nd edition
- The Unified Modeling Language User Guide, Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education

▪ **Other References:**

- UML 2 Bible, Tom Pender, Wiley-Dreamtech
- The UML Reference Manual, Ivar Jacobson, James Rumbaugh, Grady Booch, Addison Wesley
- Web reference of Object Management Group (OMG)
<http://www.uml.org/>

▪ **Practical:**

- Practical on the basis of structural and behavioral diagrams of UML can be performed in Dia tool.
 - http://sourceforge.net/projects/dia-installer/?source=typ_redirect
-