



Object Oriented Analysis and Design using Unified Modeling Language



Er. Shiva Kunwar
Lecturer, GU

Lesson 1: Object Oriented Fundamentals (10hrs)

- Introduction
- Object Oriented Analysis and Design
- Iterative development and unified process
- Case Study
- Understanding Requirements
- Use Case modeling, Relating Use cases — include, extend and generalization
- Overview of the Unified Modeling Language: UML Fundamentals and Notations

Object Oriented Concept: Decomposition

- It is the process of partitioning the problem domain into smaller parts so that the overall complexity of the problem can be comprehended and tackled easily (divide and conquer approach)
- More manageable parts where each part focuses on a specific aspect or functionality.
- Widely used in software development cycle.
- Makes problem identifiable and solvable. Each module can be individually understood, designed, implemented, and tested.
- The working scope becomes more focused and manageable. Developers can work on individual modules independently, facilitating parallel development and collaboration.

Object Oriented Concept: Objects

- Object is a real-world entity that may have physical or conceptual existence.
- Each object consists of data and a set of functions.
- Each object has identity, behavior and state.
 - Identity is the instance ID of an object that distinguishes it from other objects in the system.
 - Behavior is the methods that the object performs.
 - State determines the characteristic properties of an object as well as the values of properties.
- For e.g.: Let transport be an object:
 - Identity - Bus
 - Behavior - move(), stop()
 - State - running, stopped

Object Oriented Concept: Class

- Class is a description of a set of objects that share the same characteristic properties and exhibit common behavior.
- Class is a collection of related objects.
- A class encapsulates the data and its behavior.
- Class constitutes of:
 - Attributes are the data items that define or describes object.
 - Operations are the functions that portray the behavior of objects.

Object Oriented Concept

- Method
 - Method is a mean by which objects can manipulate data.
 - The operations supported by an object are called its methods.
- Messages
 - A message is a method call from one object to another.
 - It is a mean by which object interacts.
 - Objects communicate with each other by sending messages.
 - If obj1 wants obj2 to execute one of its methods, obj1 sends a message to obj2.

Object Oriented Concept

- Subclass
 - specialization of the super class
 - A subclass can inherit both attributes and operations from a super class.
- Superclass
 - also called a base class, is a generalization of a set of classes that are related to it.

Object Oriented Concept

- Object-Oriented Programming is a programming paradigm that uses the concept of "objects" to organize and structure code.
- Objects encapsulate data (attributes) and behavior (methods) within a single unit, allowing for a more modular and reusable approach to software development.

Object Oriented Features

- **Abstraction:** Abstraction involves simplifying complex systems by modeling classes based on their essential characteristics. It focuses on what an object does rather than how it achieves its functionality.
- **Encapsulation:** Objects encapsulate data and methods, hiding the internal details and exposing only what is necessary.
- **Modularity:** Encapsulation contributes to modularity, making it easier to understand, maintain, and modify individual components without affecting the entire system.

Object Oriented Features

- **Inheritance:** Allows the creation of new classes by inheriting attributes and behaviors from existing classes, promoting code reuse.
- **Hierarchy:** Inheritance creates a hierarchy of classes, providing a clear organizational structure.
- **Polymorphism:** Objects of different classes can be treated as objects of a common base class, enabling flexibility and extensibility.

Object Oriented Analysis and Design

- **Extension of OOP Principles:**
- Object-Oriented Analysis and Design (OOAD) extends the principles of OOP to the entire software development process.
- OOAD involves analyzing a system, identifying objects and their interactions, and designing a modular and scalable solution based on these analyses.

Object Oriented Analysis and Design

- **Unified Process and Iterative Development:**
- OOAD often follows an iterative development approach, where the development process is broken down into smaller cycles.
- The Unified Process is a framework that guides iterative development, providing guidelines for managing and documenting the software development life cycle.

Object Oriented Analysis and Design

- **Role of OOAD in Software Development:**
- OOAD is crucial for creating robust and maintainable software systems by promoting a modular and scalable design.
- It involves iterative cycles of analysis, design, implementation, and testing to ensure the development of high-quality software.

Object Oriented Analysis and Design

- **Case Study:**
- Case studies are used in OOAD to apply theoretical concepts to real-world scenarios.
- They provide practical examples for students to analyze, design, and implement using OOAD principles.

Structured System

1. System is decomposed into functions.
2. Functions and data are modeled separately.
3. System components are dependent on each other.
4. Inheritance and polymorphism are impossible.
5. Reuse is limited.
6. It does not reflect real world entities.
7. Development process is linear.
8. Testing is done in implementation phase only.

Object Oriented System

1. System is decomposed into data objects.
2. Functions and data are modeled in one
3. System components are independent of each other.
4. Inheritance and polymorphism are possible.
5. There is no restriction for reuse.
6. It reflects real world entities.
7. Development process is iterative and incremental.
8. Testing is distributed evenly.

Object Oriented Analysis

- Analysis emphasizes an investigation of the problem.
- "OOA deals with requirements, rather than a solution."
- For example, if a new online trading system is desired, how will it be used? What are its functions?
- Typical question starts with What...? like "What will my program need to do?",
- "What will the classes in my program be?" and "What will each class be responsible for?" .
- Hence, OOA cares about the real world and how to model this real world without getting into much detail.

Object Oriented Analysis

- It is the procedure of identifying software requirements and developing software specifications in terms of object model that comprises interacting objects.
- Requirements are organized around objects integrating both data and functions.
- The tasks involved are as follows:
 - Identify objects
 - Organize objects by creating object model diagram
 - Define object attributes
 - Define object actions
 - Describe interaction between objects
- It maps the problem domain directly into a model.

Object Oriented Analysis

- A typical implementation of OOA is to create an object model for an application.
- The object model might describe names, relationships, behaviors, and characteristics of each object in the system.
- With this information established for each object, the design process that follows is much simpler.
- The common models used in OOA are Use Cases and Object Models.

Object Oriented Design

- Design emphasizes a conceptual solution (in software and hardware)
- It fulfils the requirements, rather than its implementation.
- For example, a description of a database schema and software objects.
- Design ideas often exclude low-level or "obvious" details obvious to the intended consumers.
- The question typically starts with How...? like "How will this class handle it's responsibilities?", "How to ensure that this class knows all the information it needs?"

Object Oriented Design

- It is the process of implementing conceptual models produced during analysis phase.
- The technology independent conceptual models are mapped into implementing classes, constraints are identified, and interfaces are designed.
- It results a model for the solution of a problem domain.
- The tasks involved are as follows:
 - Define object life cycle
 - Define class relationships
 - Define service logic
 - Complete class definitions and associations

Object Oriented Analysis and Design

- It is technical approach for analyzing and designing an application by applying object-oriented programming.
- During analysis phase, you try to determine which objects you need and how to organize them.
- During design phase, you take the analyzed objects and add constraints to make them fit within the software and hardware that you are developing for.
- If OOA is the what, then OOD is the how.

Analysis

1. Classes
 - ♦ Who am I? What is the same/different?
 - ♦ What do I contain? Who am I associated with?
2. Attributes
 - ♦ What do I need to know?
3. Behaviors
 - ♦ What can I do?
4. Collaborations
 - ♦ What help do I need? Who needs my help?

Design

1. Decide how to implement
 - ♦ Classes
 - ♦ State
 - ♦ Behavior
 - ♦ Collaborations
2. Add Implementation-Specific Components
 - ♦ Human interaction
 - ♦ Data management
 - ♦ Other implementation areas

Object Oriented Analysis and Design

- During object-oriented analysis, there is an emphasis on finding and describing the objects—or concepts—in the problem domain. For example, in the case of the library information system, some of the concepts include Book, Library, and Patron.
- During object-oriented design, there is an emphasis on defining software objects and how they collaborate to fulfill the requirements. For example, in the library system, a Book software object may have a title attribute and a getChapter method
- Finally, during implementation or object-oriented programming, design objects are implemented, such as a Book class in Java.

How are OOA,OOD and OOP related?

- The product of OOA serves as the models from which we may start an OOD ,
- The product of OOD can then be used as blueprint for completely implementing a system using OOP methods.
- OOA is concerned with developing an object model of the application domain,
- OOD is concerned with developing an object-oriented system model to implement requirements,
- OOP is concerned with realizing an OOD using an OO programming language such as Java or C++.

END OF LECTURE

SHIVA.KUNWAR@HOTMAIL.COM

+977-9819123654

PREVIEW FOR LECTURE 3

OOAD MODELING
ITERATIVE DEVELOPMENT
UNIFIED PROCESS
UNDERSTANDING REQUIREMENTS