Lesson 4: Object Oriented Modeling

Aniruddha Kudalkar

Objective

- Object Oriented In Depth
- UML Overview
- Case Study 1
- Case Study 2
- Case Study 3

Object Oriented In Depth

Object Oriented Principles

OOP resemble with real world concerns

- Class and Object
- Principles
- Relations

Class and Object

Foundation of Object Oriented Principles

class

a class is an extensible program-code-template for creating objects, providing initial values for state (member variables) and implementations of behavior (member functions or methods).

object

In the object-oriented programming, object can be a combination of variables, functions, and data structures; In Non OOP, an object can be a variable, a data structure, a function, or a method, and as such, is a value in memory referenced by an identifier.

Interface

term interface is used to define an abstract type that contains no data but defines behaviours as method signatures. An interface is thus a type definition; anywhere an object can be exchanged, the type of the object to be exchanged can be defined in terms of one of its implemented interfaces.

Object Oriented Principles

Pillar of Object Oriented Programming

Encapsulation

Bundling of data with the methods that operate on that data, or the restricting of direct access to some of an object's components.

Abstraction

Abstraction is used to hide the values or state of a structured data object inside a class, preventing direct access to them by clients in a way that could expose hidden implementation details or violate state invariance maintained by the methods.

Inheritance

mechanism of basing an object or class upon another object (prototype-based inheritance) or class (class-based inheritance), retaining similar implementation. Also defined as deriving new classes (sub classes) from existing ones such as super class or base class and then forming them into a hierarchy of classes.

Polymorphism

is the provision of a single interface to entities of different types or the use of a single symbol to represent multiple different types.

Relations

Entities follow some relations

Types

- One To One
- One to Many
- Many To Many

UML Overview

UML

The Unified Modeling Language (UML) is a general-purpose, developmental, modeling language in the field of software engineering that is intended to provide a standard way to visualize the design of a system.

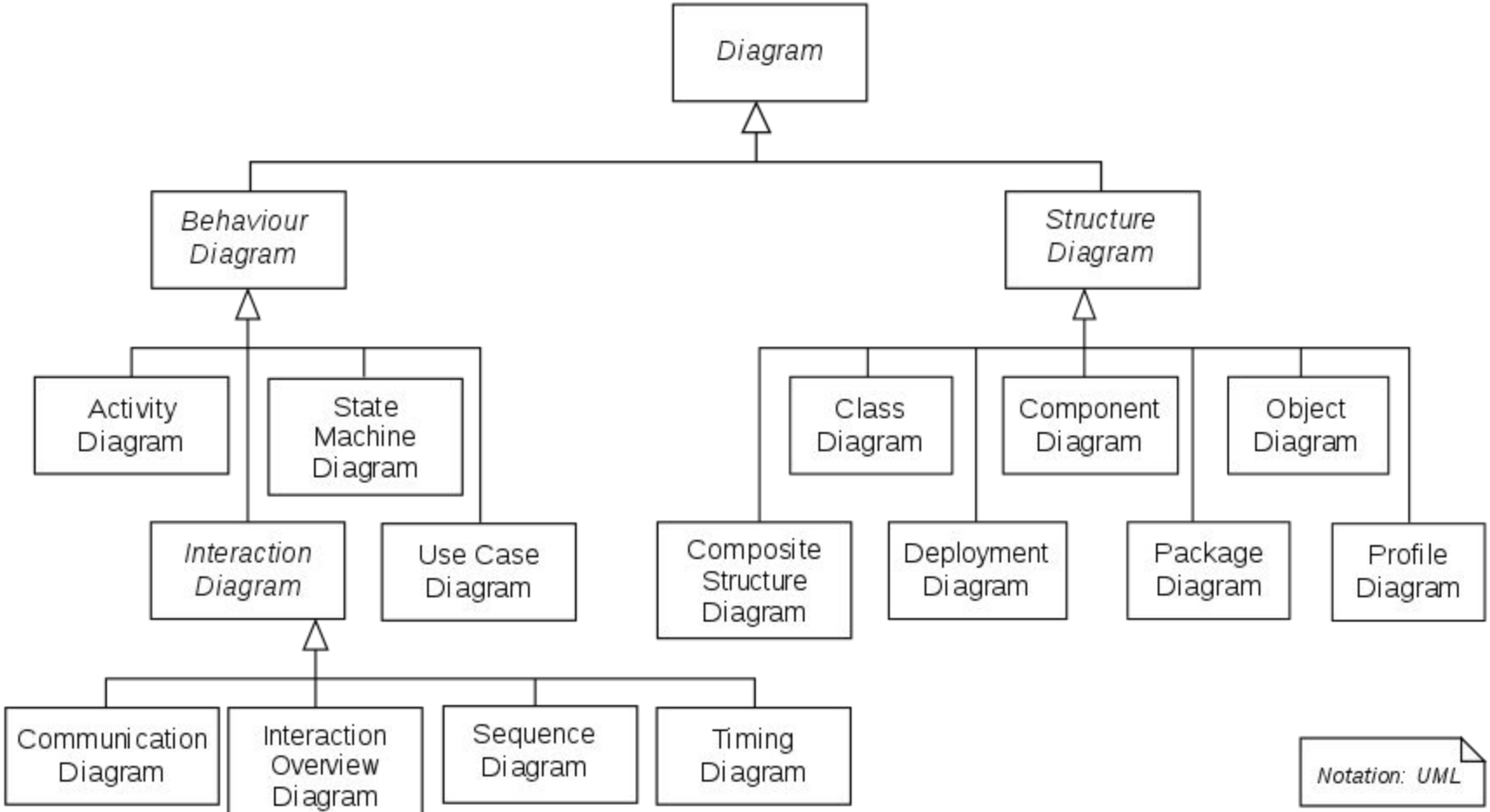
Diagram
Behaviour Diagram
Structure Diagram
Activity Diagram
State Machine Diagram
Interaction Diagram
Use Case Diagram
Class Diagram
Component Diagram
Object Diagram
Composite Structure Diagram
Deployment Diagram
Package Diagram
Profile Diagram
Communication Diagram
Interaction Overview Diagram
Sequence Diagram
Timing Diagram
Notation: UML

UML

The Unified Modeling Language (UML) is a general-purpose, developmental, modeling language in the field of software engineering that is intended to provide a standard way to visualize the design of a system.

Structure Diagrams

Structure diagrams represents the static aspects of the system. It emphasize the things that must be present in the system being modeled. Since structure diagrams represent the structure, they are used extensively in documenting.

Behaviour Diagrams

Behavior diagrams represent the dynamic aspect of the system. It emphasizes what must happen in the system being modeled. Since behavior diagrams illustrate the behavior of a system, they are used extensively to describe the functionality of software systems.

Interaction Diagrams

a subset of behavior diagrams, emphasize the flow of control and data among the things in the system being modeled.

UML Diagrams

Structural

- Class
- Component
- Composite
- Deployment
- Object
- Package
- Profile

Behavioural

- Activity
- Communication
- Interaction Overview
- Sequence
- Timing
- State
- Use Case

Case Studies

Implementation

- User Registration
- Basic Shopping Cart
- Very Basic Expense Manager

User Registration

- Show Menu
- Register User
- Login
- All Users
- Delete User

Small Digital Wallet

- Top Up
- Check Balance
- Friend List
- Send Money
- Receive Money

Basic Shopping Cart

- Display Menu
- Item details
- Display Cart
- Add to cart
- Remove From Cart
- Total Bill

Thanks,

LEARN, CODE, EARN

Credits

- https://en.wikipedia.org/
- https://docs.oracle.com/