Object Oriented Software Engineering

OOSE is a software design technique that is used in software design in OOP.

It consists of two terms – object oriented, and software engineering.

Object Oriented

It is a collection of information that itself act as a singular entity. It allows the user to focus completely on the task rather than on the tools.

For example – C++, etc.

With the help of this, reusability as well as abstraction is possible.

The necessity of developing a maintaing a large-size, complex, and varied functionalities software system has caused us to look for new approaches of software design and development.

The conventional approaches like Waterfall Model may not be very useful due to non-availability of iterations, no provision of reuse, and difficulty in incorporating changing requirements.We may also build every software system from scratch that results into a costly software syste, including very high maintenance cost.

An object oriented approach may address such issues, that's why it has become very popular in desiging,developing, and maintaining large size software systems.

Software Engineering

> It is a profession dedicated to designing, implementing and modifying so that the software is more affordable, maintainable, faster to build, and high quality.

OR

> The establishment and use of some engineering principles in order to obtain economically developed software that is reliable and works efficiently on real machines.

Software

It is a combination of programs, documentation and operating manual.

Program

A certain set of instructions that are written for a specific purpose. It may contain statements to enhance the readability of the program.

Documentation

Documentation is created and used during development. It is used to explain the code, what it does, and why it has been coded in a certain way.

Operating Manual

Explains to the customer how the software is to be used. It is delivered along with the software to the customer, at the time of release.

The use of use cases was introduced in Object Oriented Methodology.

Characteristics of Software

Bathtub and software curve bs

Object Oriented Basic Concepts

1. Classes
2. Objects
3. Data Abstraction
4. Encapsulation
5. Inheritance
6. Polymorphism

Classes

> A class represents a template for different objects and describes how these objects are structured internally. Objects of the same class have the same definition, both for the operations, and for the information structures.

OR

> It is a collection of objects and it doesn't take any space in memory. It is also called a blueprint, or a logical entity.

There are two types:

- *Pre-defined*

  Their logic is already written somewhere, and we can use it by importing. For example - Scanner, Console, etc. in Java

- *User-defined*

  The logic for these classes is defined by the programmer.

Encapsulation

> The wrapping up of data and functions into a single unit. It is also knownb as information hiding concept.

Inheritance

> Deriving a new class from existing class in such a way that the new class can access all the features and properties of the existing class.

The existing class is called parent class, super class, base class. The new class is called child class, sublcass, derived class.

Data Abstraction