External Learning

Object Oriented Methodology:

It is a new system development approach, encouraging and facilitating re-use of software components. It employs international standard unified Modeling Language from the object Management group(OMG). Using this methodology, a system can be developed on a component basis, which enables the effective re-use of existing components., it facilities the sharing of its other system components.

There are three types of object oriented methodologies :

↳ Object Modeling Techniques (OMT)

↳ Object process Methodology (OPM)

↳ Rational Unified process (RUP)

1. Object Modeling Techniques (OMT):

This uses three different models that are combined in a way that is analogous to the older structured methodologies.
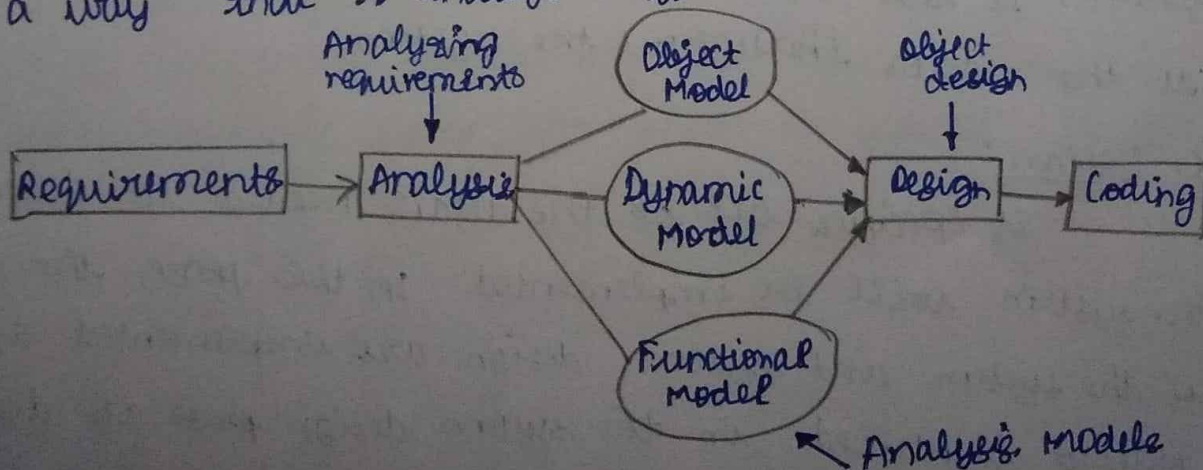
← Analysis Models

a. Analysis:

The main goal of the analysis is to build models of the world. The requirements of the users, developer and managers provide the information needed to develop the initial problem statement.

b. OMT Models:

b(i) Object Model

It depicts the object classes and their relationships or class diagram - which represents the static structure of the system. This model does not pay attention to Object's dynamic nature.

b(ii) Dynamic Model

It captures the behavior of the system over time and the flow control and events in the Event-trace diagrams and state transition diagrams. It shows the changes occuring in the states of various objects with events that occur in the system.

b(iii) Functional Model

It describes the data transformations of the system. It also describes the flow of data & changes that occur to the data throughout the system.

c. Design:

It specifies all of the details needed to describe how the system will be implemented. In this phase, the details of the system and system design are implemented and the objects identified in the system design phase are designed.

2. Object Process Methodology (OPM)

It is also second generation methodology. It has only one diagram called Object Process Diagram (OPD) which is used for modeling the structure, function and behavior of the system.

It consists of three main processes:

↳ Initiating : It determines high level requirements, the scope of the system and the resources that will be required.

↳ Developing: It involves the detailed analysis, design and implementation of the system.

↳ Deploying: It introduces the system to the user & subsequent maintenance of system.

3. Rational Unified Process (RUP)

It consists of four phases:

↳ Inception
↳ Elaboration
↱ Construction
↳ Transition

Each iteration consists of nine areas of discipline. A discipline depends upon the phase in which, iteration taking place.

Objectives of OOAD Object oriented methologies:

↳ To encourage greater reuse.

↳ To produce more detailed specification of system constraints.
↳ To have fewer problems with validations.

Benefits:

↳ Easier to produce and understand design

↳ Simplicity and Reusability.

↳ Increased quality and maintenance.

↳ Scalable, modularity and modifiability.

↳ Client / server Architecture

Software Quality Assurance:

It is a process which works parallel to development of software. It is a methodology that determines the extent to which a software product is fit for use. The activities that are included for determining software quality are:

↳ Auditing
↳ Development of standards and guidelines.
↳ production of reports.
↳ Review of quality system.

Quality Factors:

Correctness - Determines whether the software requir -ements are appropriately met.

Usability - Determines whether the software can be used by different categories of users.

Portability - Determines whether the software can run different hardware devices.

Maintainability - Determines ease at which errors can corrected and updated in modules.

Reusability - Determines whether the modules / classes can be reused for developing other software products.

Impact of object orientation on Testing:-

Most current software testing are congruent with functional software. Object orientation makes the testing easier by introducing concepts like modularity, reusability, etc. A unit of software is either tested against specification or some criteria to execute identified parts.

In object oriented system testing has three levels: Unit testing, subsystem testing.

Unit testing:

Here, individual classes are tested. It is seen whether the class attributes are implemented as per design and whether the methods & interfaces are error-free.

Subsystem testing:

Here, it involves testing a particular module or a subsystem & is the responsibility of the subsystem lead. This involves testing the associations within the subsystem as well as interaction of the sub-system with the outside.

System testing:

It involves testing the system as whole. System tests are used as regression tests when assembling new release.

Develop Test cases and Test plans:

Test case is a document that lays out the following for a singular test scenario:

↳ Test data
↳ Scenarios
↳ Description
↳ Procedures/inputs
↳ Testing environment
↳ Expected & actual results

A test plan is a comprehensive document that lays out all major activities associated with a particular testing project.

It includes:
↳ Scope of project
↳ Target market
↳ Goals & objectives
↳ Assumptions
↳ Testing environment
↳ Deliverables
↳ Testing cycle start/end date
↳ Testing end data
↳ Major risks & Handling.

There are
Three types of plan:-

↳ Master test plan: Has multiple levels of testing.

↳ Phase test plan: Handles any one phase of testing.

↳ Specific test plan: Designed for non-functional testing.
                      (E$ Security testing')

Writing a test plan (steps):

↳ Analysing the product
↳ Design test strategy
↳ Define test objectives
↳ Define test criteria

- → Resource planning.
- → Plan test environment.
- → Schedule and estimation
- → Determine test deliverables.

Few Guidelines

- → Specificity
- → Avoid redundancy
- → Use Lists & tables
- → Avoid lengthy paragraphs
- → Delete unnecessary sections
- → Update the plan.

Importance:

- → It helps to determine and validate the quality of software applications.

- → It helps people to understand the test details related to the outside like developers, customers, etc.