Object Oriented Testing

Group # 8

Represented By: Haris Jamil
Discipline: Information Technology
5th Semester

OO testing Activities

- Review OOA and OOD models
- Class testing after code is written
- Integration testing within subsystems
- Validation testing based on OOA use-cases

Outlines

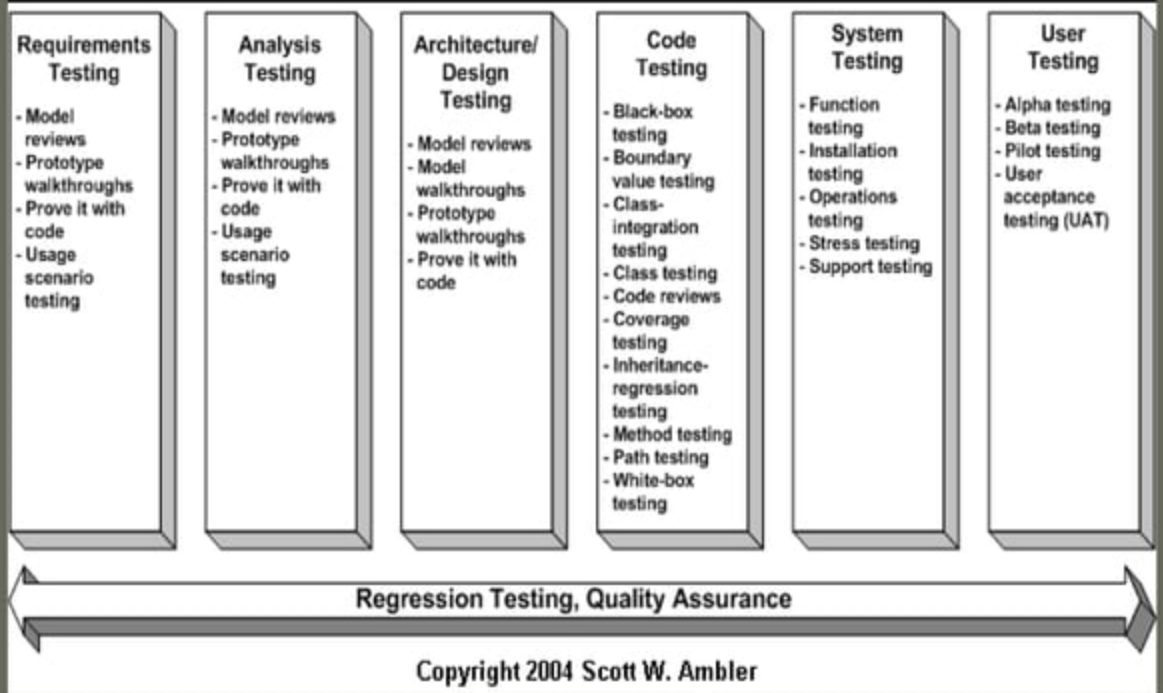
- OO-Testing.
- Requirement Testing
- Analysis and Design Testing
- Code Testing
- User Testing
- Integration Tests
- System Tests
- Testing Methods

What is OOT ?

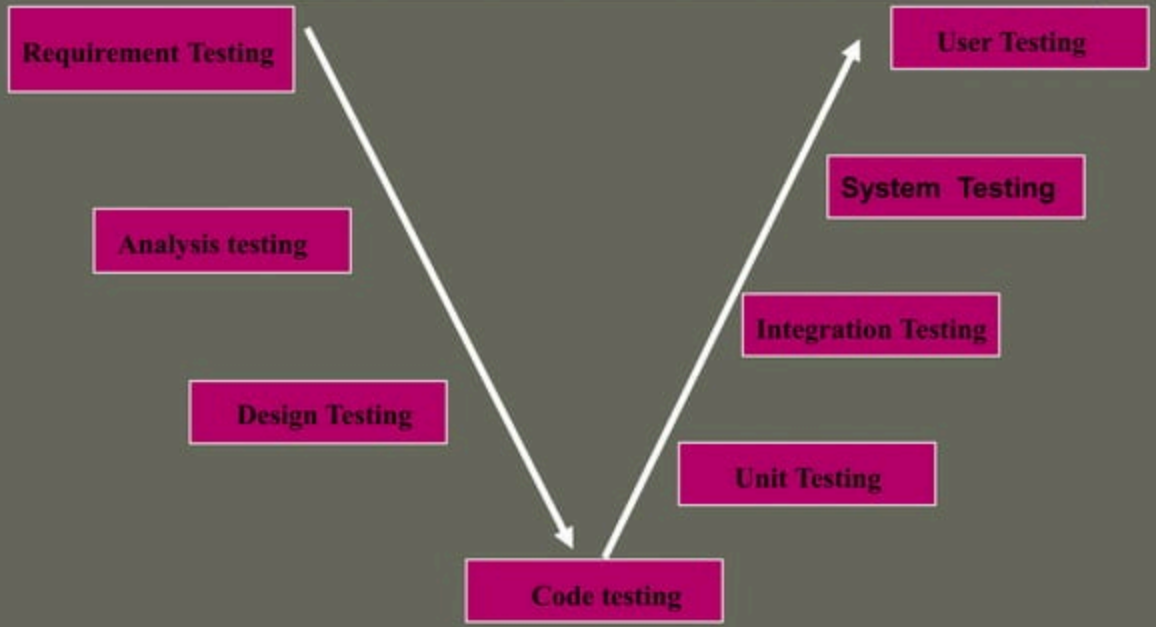
- Object-Oriented Testing is a collection of testing techniques to verify and validate object-oriented software.
- Testing takes place

- To complete the OOT cycle mention below testing are required.
  - Requirement Testing
  - Analysis and Design Testing
  - Code Testing
  - Integration Tests
  - System Tests
  - User Testing
- It's used to discuss test plans and execution for projects.

FLOOT - Diagram



The Structured Testing Pyramid



Requirements based Testing :

Requirements-based testing is a testing approach in which test cases, conditions and data are derived from requirements. It includes functional tests and also non-functional attributes such as performance, reliability or usability.

Requirements Testing process:

- Testing must be carried out in a timely manner.
- Testing process should add value to the software life cycle, hence it needs to be effective.
- Requirement testing process needs to be efficient as well.
- Testing must provide the overall status of the project, hence it should be manageable.

Stages in Requirements based Testing:

- **Defining Test Completion Criteria** - Testing is completed only when all the functional and non-functional testing is complete.
- **Design Test Cases** - A Test case has five parameters namely the initial state or precondition, data setup, the inputs, expected outcomes and actual outcomes.
- **Execute Tests** - Execute the test cases against the system under test and document the results.
- **Verify Test Results** - Verify if the expected and actual results match each other.
- **Verify Test Coverage** - Verify if the tests cover both functional and non-functional aspects of the requirement.
- **Track and Manage Defects** - Any defects detected during the testing process goes through the defect life cycle and are tracked to resolution. Defect Statistics are maintained which will give us the overall status of the project.

Analysis Testing :

Analysis testing is the process of looking at something that can be used to derive information. In analysis we collect information that is needed in order to start the test, In it we make documentation on requirements, design specifications, product risk analysis,
architecture and interfaces.

Analysis testing is used to understand what the system should do once built.
Sometimes analysis testing can be based on user's experienced knowledge about system or software which may not be documented.

Design Testing :

By design testing we create a plan how to implement an idea and technique.
So, **Design Testing** creating a set of inputs for the given software that will provide a set of expected outputs.

They are two main **Design Testing Techniques :**

- Static Techniques
- Dynamic Techniques

Static & Dynamic Testing Techniques

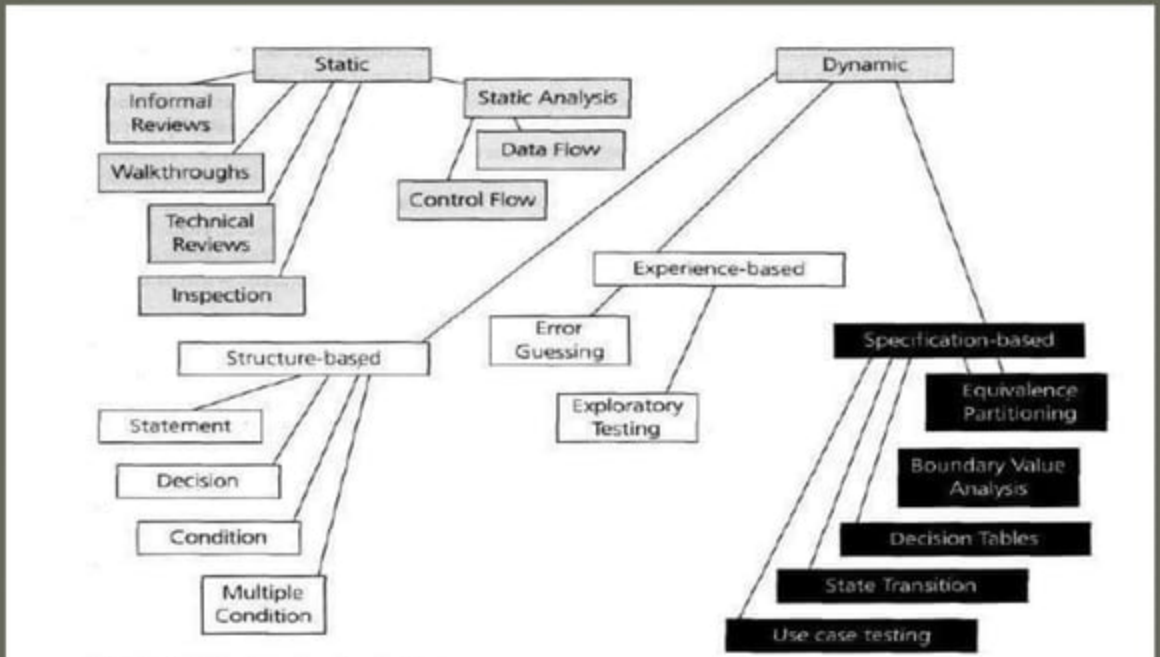


FIGURE 4.1 Testing techniques

•Static testing is the testing of the software work products manually, or with a set of tools, but they are **not executed**.
•It starts early in the Life cycle and so it is done during the verification process.
•**It does not need computer as the testing of program is done without executing the program.** For example: reviewing, walk through, inspection, etc.
•Most static testing techniques can be used to 'test' any form of document including source code, design documents and models, functional specifications and requirement specifications.

•Dynamic testing technique needs computer for testing.

•It is done during Validation process.

•The software is tested by executing it on computer.

•Example of this **Dynamic Testing Technique**: Unit testing, integration testing, system testing.

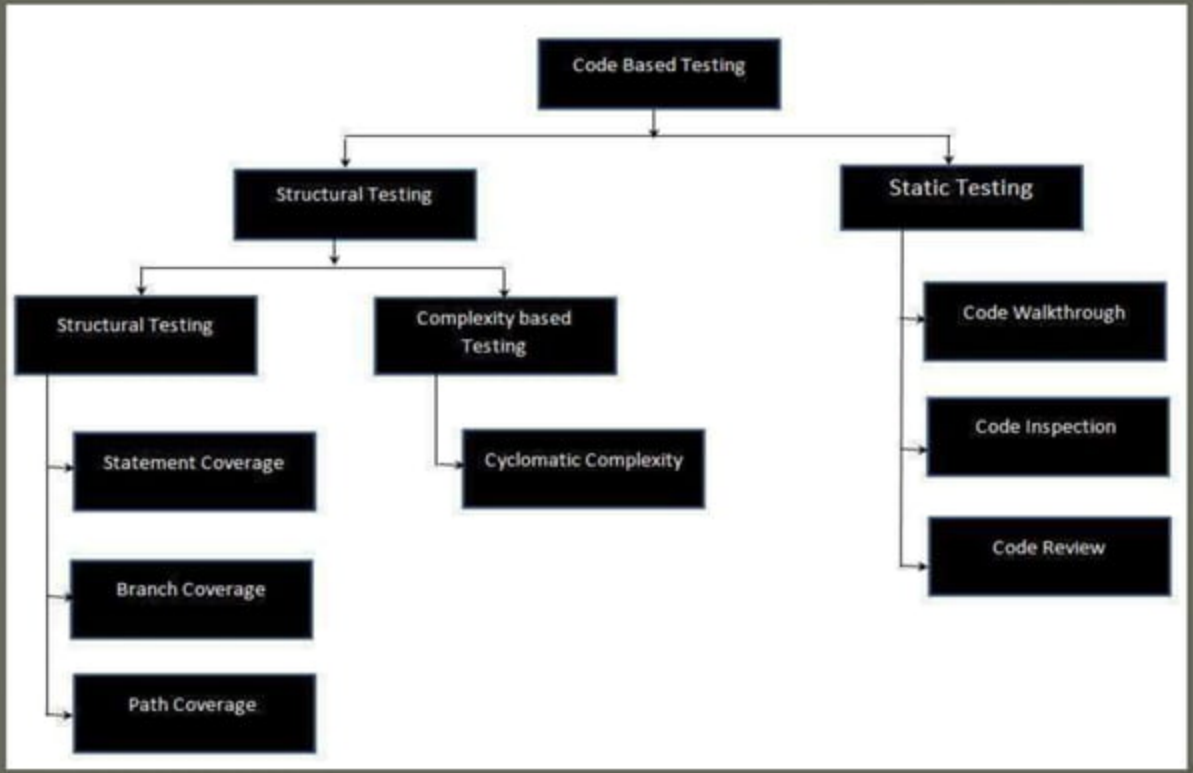
Code-Based Testing:

Code-based testing corresponds to the testing that is carried out on code development, code inspection, unit testing in software development process.

The Code-based testing consists of following testing:

- Dynamic Testing - Statement coverage, Branch coverage, Path coverage.
- Checking for Complexity of Code using techniques like Cyclomatic Complexity.
- Static Testing - Code Inspection, Code Walkthrough, Code Review, Code Audit.

Code Based Testing Techniques:



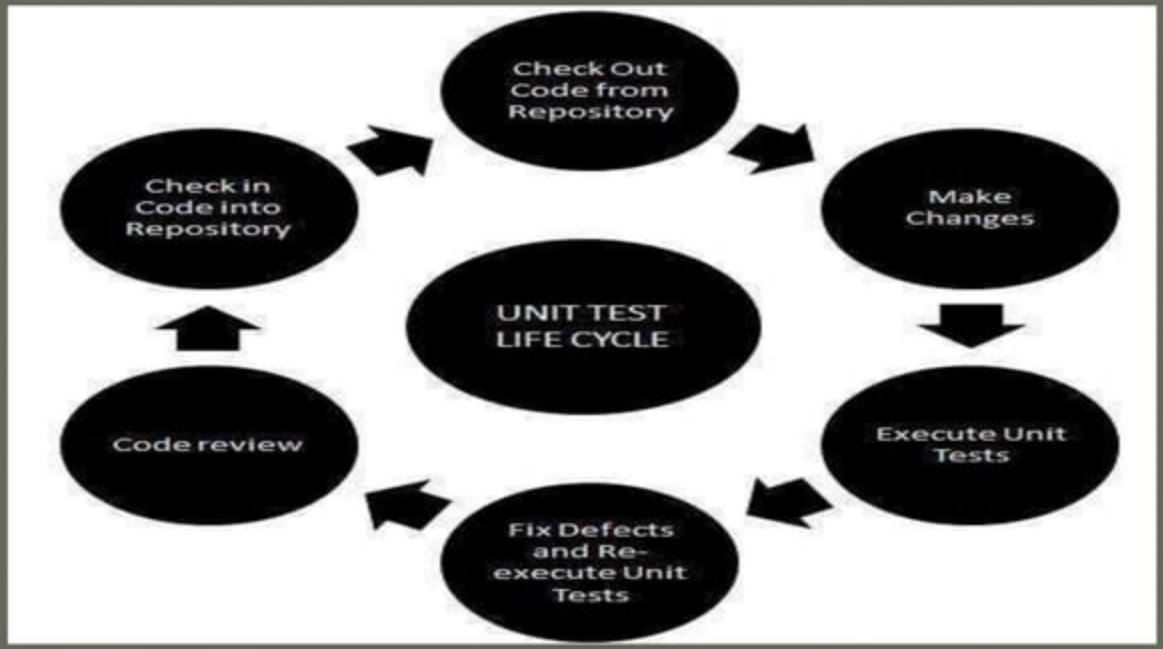
Unit Testing :

Unit testing, a testing technique in which individual modules or units of source code are tested by the developer to find out the issues and to make sure that code meets its design and requirements and behaves as expected.

Unit Testing - Advantages:

- Reduces Defects in the Newly developed features or reduces bugs when changing the existing functionality.
- Reduces Cost of Testing as defects are captured in very early phase.
- Improves design and allows better refactoring of code.
- Unit Tests, when integrated gives the quality.

Unit Testing Life Cycle :



Unit Testing Techniques:

- Black Box Testing
- White Box Testing
- Gray Box Testing

Integration Testing :

The software **testing phase** in which individual software modules are combined and **tested** as a group. It occurs after unit **testing** and before validation **testing**.

Purpose of Integration Testing :

The purpose of integration testing is to verify the functional, performance, and reliability between the modules that are integrated.

Integration Strategies:

- Big-Bang Integration
- Top Down Integration
- Bottom Up Integration
- Hybrid Integration

System Testing :

The process of performing a variety of test on a system to explore functionality or to identify problems.
Client test the system to find out the bugs in system/software.

For example : A tester may put a wrong input in a field that basically designed to only accept states of USA , for just check how the system will respond to the incorrect input.

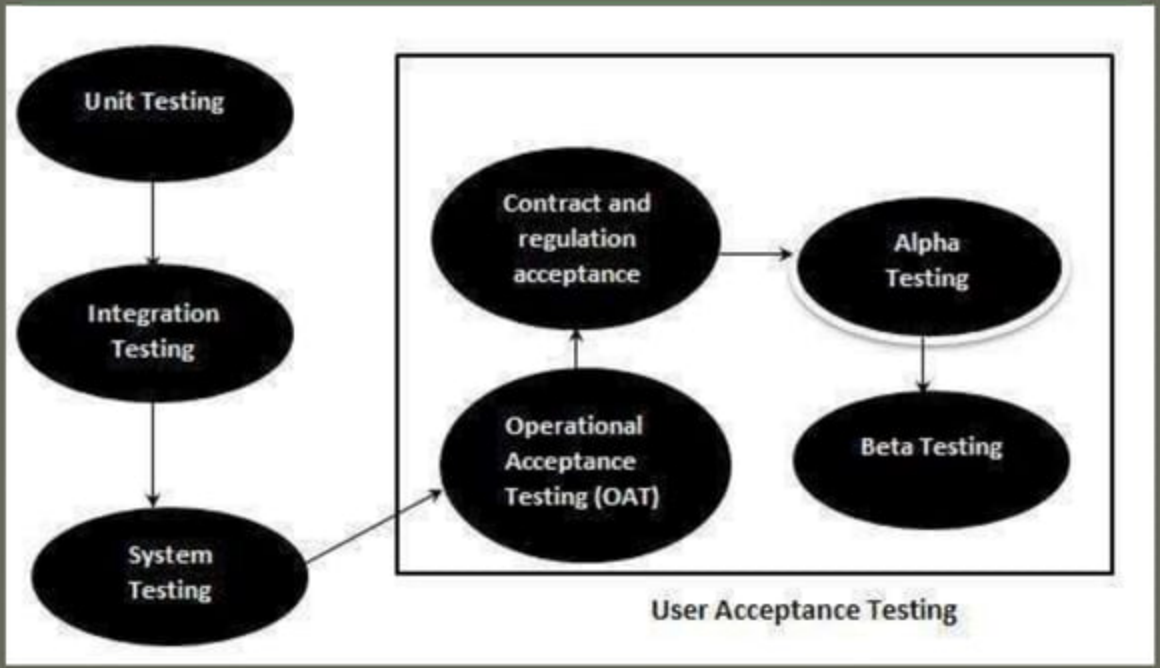
Purposes of System Testing :

- System testing is most often the final test to verify that the system to be delivered meets the specification and its purpose.
- System testing is carried out by specialists testers or independent testers.
- System testing should investigate both functional and non-functional requirements of the testing.

User Testing :

(UAT) is the last phase of the software **testing** process.
It's a testing methodology, clients/end users involved in testing the product to check or validate the software against their requirements. It is performed at client location at developer's site.

UAT - Diagram


Unit Testing
Integration Testing
System Testing
Contract and regulation acceptance
Alpha Testing
Operational Acceptance Testing (OAT)
Beta Testing
User Acceptance Testing

Scenario Base testing :-

- Scenario testing is a software testing technique by which we test the software activity that uses scenarios.
- If the scenario runs from start to finish, then it passes.
- Scenario is just a story which explains the usage of the software by any end user.

e.g.

- Name must be in capital letters
- Length of letters is 20 charaters
- CNIC have 13 numbers with special letter of (-)
- Now user must log in by accepting its data

THANK YOU ☺

Any Question????