**Integration Testing --**

It is a level of software testing where individual units are combined and tested as a group.

Testing performed to expose the defects in the interactions between integrated components or system is called integration testing.

The purpose of this testing is to expose defects or faults in the interactions between integrated units.

Integration testing is performed by developer (wrt connectivity) or by the independent testers (wrt integrity).

There are different approaches in Integration Testing

1.      Top Down Approach

2.      Bottom Up Approach

3.      Hybrid Approach

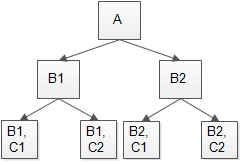
4.      System Approach (Big bang Approach)

Top down Approach-

This technique starts from the top-most module & gradually progresses towards lower modules.

Only the top module is tested in isolation, after this lower modules are integrated one by one.

The process is repeated until all modules are integrated and tested.



From above diagram testing starts from module A & lower modules B1 & B2 are integrated one by one.

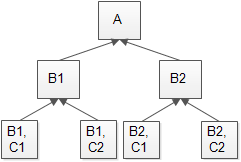
At this stage lower modules B1 & B2 are actually under development so in order to test topmost module A completely, we need to develop a temporary/dummy program called **Stub** to simulate functionality of the module which is under development.

Bottom up Approach –

As the name suggests, testing starts from the lowest or innermost units of the application & gradually moves up.

This Integration Testing starts from the lowest module & gradually progresses forward to upper modules of the application.

This integration continues until all modules are integrated & the entire application is tested as a single system.



From above diagram module B1C1 & B1C2 are lowest modules which will be unit tested.

Modules B1 & B2 are not yet developed & we need some program that will simulate functionality to call B1C1, B1C2, B2C1, & B2C2 modules.

The simulator program (dummy/temporary) is called Driver.

Driver is a dummy program which is used to call the function of the lowest module in case the calling function does not exist.

Hybrid Approach –

This is a combination of both top down & bottom up approach in order to check or test huge applications.

Integration starts with the middle layer & moves forward up and down.

System Approach –

The testing takes place only when all modules are developed & unit tested.

Steps for Integration Testing (Process)

1.      Understand architecture of application.

2.      Identify the modules.

3.      Understand what each module does.

4.      Understand how data is transferred from one module to another.

5.      Understand how data is entered & received into the system.

6.      Identify & create test conditions & test cases to execute.

7.      Test the conditions one by one.

Sample Test Condition for Integration Testing

1.      Test Condition 1:- Verify the interface link between login page & home page i.e. When a user enters valid credentials to login then he or she should be directed to the home page.

2.      Test Condition 2:- Verify interface link between home page and profile page.

3.      Test Condition 3:- Verify interface link between network page & connection page i.e. clicking accept button on network page, user should be able to accept the connection request & number of connections should update on connection page.

4. Test Condition 4:- Verify the interface between Amazon cart to payment gateway to check data (amount and notification (pass/fail)) exchange between both the application.

Note – Integration Testing can be performed manually as well as using tools like Rational integration testing, Protractor, Tessy.

Interview Questions:

1.      What is integration testing?

2.      Who performs integration testing?

3.      When it is performed?

4.      What do you check in integration testing?

5.      What are approaches to integration testing?

6.      Give examples of integration testing.