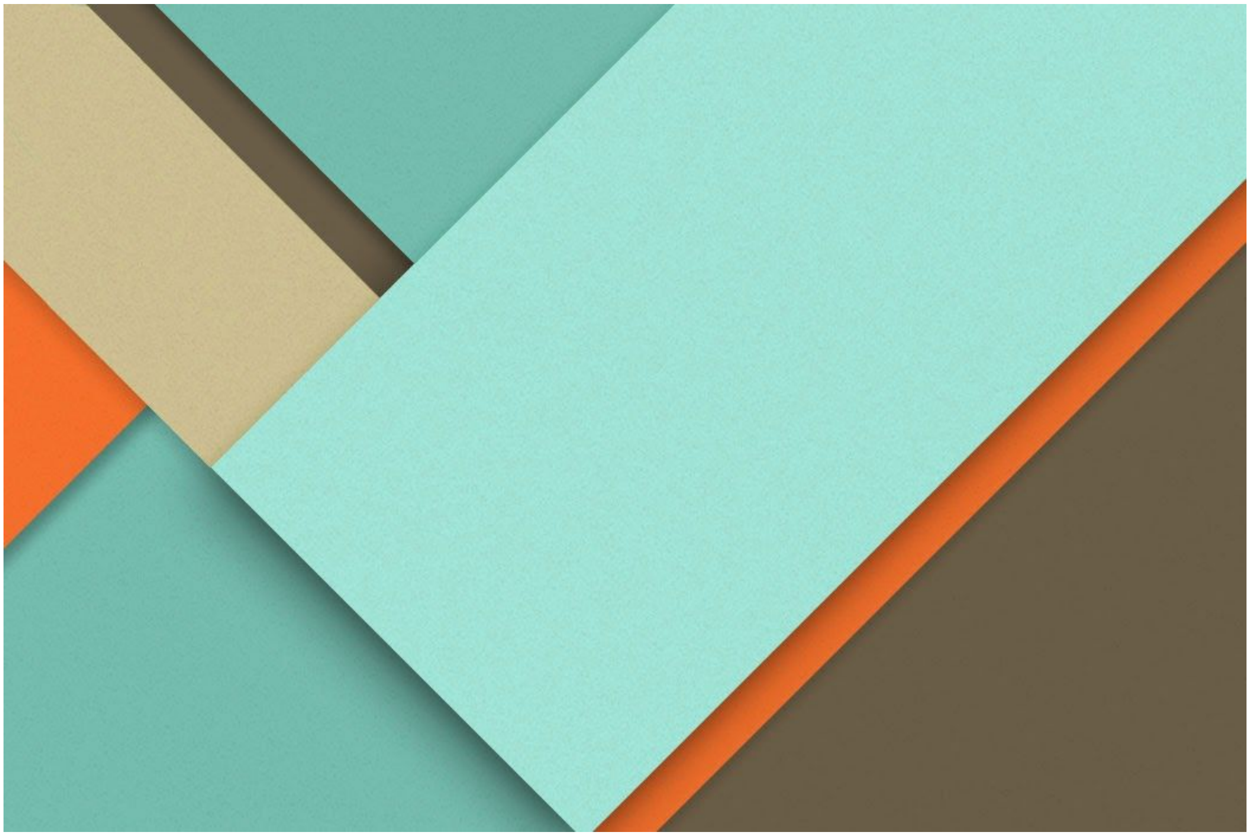


SEPM week 1 assignment

02.04.2020

RA1811003010303

RAGHU.B

CSE E1

1. Define Software Testing and types of testing.

Ans. Testing is the process of exercising a program with the specific intent of finding errors prior to delivery to the end user.

Types of testing are:-

Unit testing [white box]

Concentrates on each component/function of the software as implemented in the source code

Integration testing

Focuses on the design and construction of the software architecture

Validation testing

Requirements are validated against the constructed software

System testing

The software and other system elements are tested as a whole

There are other sublevel testing are performed like

Functional Testing (after Unit testing) – Testing technique used – Black box

Regression Testing, Smoke Testing

2.What is Validation and Verification?

Ans. Software testing is part of a broader group of activities called verification and validation that are involved in software quality assurance

Verification :The set of activities that ensure that software correctly implements a specific function or algorithm. It refers to the set of tasks that ensure that software correctly implements a specific function

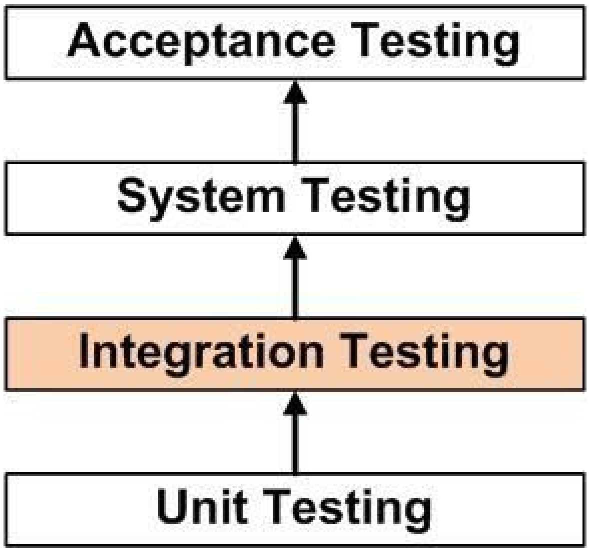
Validation:-

The set of activities that ensure that the software that has been built is traceable to customer requirements.

Verification and validation includes a wide array of SQA activities: technical reviews, quality and configuration audits, performance monitoring, simulation, feasibility study, documentation review, database review, algorithm analysis, development testing, usability testing, qualification testing,acceptance testing, and installation testing.

3.Discuss about Integration test and its types.

Ans.

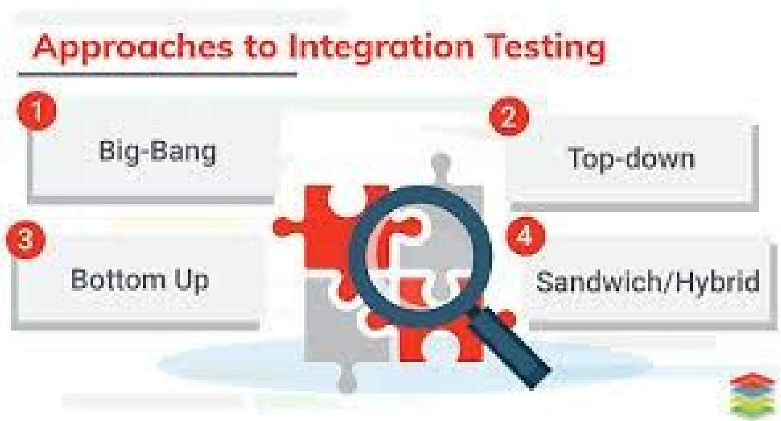


Acceptance Testing

System Testing

Integration Testing

Unit Testing



Approaches to Integration Testing

Defined as a systematic technique for constructing the software architecture. At the same time integration is occurring, conduct tests to uncover errors associated with interfaces Objective is to take unit tested modules and build a program structure based on the prescribed design

Two Approaches

Non-incremental Integration Testing

Incremental Integration Testing

Non Incremental Testing:-

Commonly called the “Big Bang” approach.

All components are combined in advance.

The entire program is tested as a whole Chaos results many seemingly-unrelated errors are encountered Correction is difficult because isolation of causes is complicated. Once a set of errors are corrected, more errors occur, and testing appears to enter an endless loop

Incremental Testing:-

Three kinds:-

Top-down integration

Bottom-up integration

Sandwich integration

The program is constructed and tested in small increments. Errors are easier to isolate and correct.Interfaces are more likely to be tested completely systematic test approach is applied.

Top-down integration:-

Modules are integrated by moving downward through the control hierarchy, beginning with the main module .

The control program is tested first. Modules are integrated one at a time. Emphasize on interface testing. Subordinate modules are incorporated in either a depth-first or breadth-first fashion.

DF: All modules on a major control path are integrated

BF: All modules directly subordinate at each level are integrated

Bottom-Up Integration:-

Integration and testing starts with the most atomic modules in the control hierarchy

Allow early testing aimed at proving feasibility and emphasize on module functionality and performance

Sandwich Integration:-

Consists of a combination of both top-down and bottom-up integration.

Occurs both at the highest level modules and also at the lowest level modules.

Proceeds using functional groups of modules, with each group completed before the next High and low-level modules are grouped based on the control and data processing they provide for a specific program feature.

Integration within the group progresses in alternating steps between the high and low level modules of the group.

When integration for a certain functional group is complete, integration and testing moves onto the next group

Reaps the advantages of both types of integration while minimizing the need for drivers and stubs.

Requires a disciplined approach so that integration doesn’t tend towards the “big bang” scenario.