**LEVELS OF TESTING :**

1. Unit testing
2. Component testing
3. Integration testing
4. System testing
5. Acceptance testing

**Unit testing** – done by developers

* Can test each module separately.
* Smallest testable portion- which can be compiled, loaded, executed.
* These are individual independent units.

How Is unit testing done:

* Section of code is written to test a functionality separately, helps identify **unnecessary** dependencies.
* Automated / manual

Workflow for unit testing:

* Create a test case.(explain each small functions)
* Rework
* Baseline
* Execute test case.

Automated tools: Junit, Nunit, Jmockit, emma, PHPUnit etc.

Advantages:

* What functionality is being provided and hoe to use it.
* Useful during regression testing
* Refactor code at later date.
* Whenever a change causes fault in other functionalities

Disadvantages:

* Ant be used for checking every error in the program.
* Focuse on a unit of code cant reach integration errors.

**Module testing:**

* Tests the **software without** the third-party codes and services.
* Simulators **are replaced with real objects**.
* But real system is available only in the system testing

**Integration testing:**

* Modules are integrated and tested as a group.
* Data flow from one module to a another is checked.
* For e.g : formats of data a etc.
* Drivers or stubs are used to simulate the missing functions.

**LEVELS OF TESTING:**

* SUB UNIT – COMPONENT

**Why integration:**

* Different in programming logic.
* Verify that the software modules work in unity.
* change in requirements by clients.
* Interfaces my be enormous with software modules.

Strategies:

* Top down approach
* Bottom up approach

**Big bang testing**

All the component modules are integrated together at once and then tested as the unit.

If all components are not completed then the , the integration system will not execute.

Adv : for small systems(LTI-SMALL, FIIP- there are other systems integrated so large)

DISADV :

* FAULT LOCALIZATIO IN DIFFICULT.
* Less time in testing execution phase.
* No priority for high risk modules.

**Incremental testing:**

* Integrating two or more modules that are logically related.

**Botton -up:**

Lower-level modules are tested first.

Which are then used to facilitate the working of higher-level modules.

**Top-down:**

Stubs are used for testing if the modules are not ready.

Order cancellation etc. are tested first and then it is checked to see if the home page is working or not.

Level of testing:

ENTRY CRITERIA: - UNIT TESTED COMPONENT AND MODUELS.

EXIT CRITERIA-Integrated application is tested.

All high prioritized bugs are fixed.

Technical document to be submitted followed by release notes.

**System testing – make sure functional testing is done**

* Done on a complete integrated system.
* Allows systems compliance per the gvt. regulations.
* Nonfunctional testing’s.
* Most of the time is the final test to confirm system behaves as needed.
* **Done in an environment that is very close to the production environment.**

**Why is it needed?**

* **End to End testing scenarios.**
* **Testing the user experience with the system.**

**Different types of system testing:**

* Usability testing- how easily user understand s and can use the system.
* Load testing: real life load testing.
* Regresion testing.
* Recovery testing.
* Migration testing – android system updates
* Functional testing.
* Hw/Sw testing : interaction bwm hardware and the software

**UAT: - All the scenarios are not tested. (-happy path)**

* Goal is not destructive(doesn’t enter random data.)
* Male sure that the system meets the user requirement.
* Types:

1. Apha – BY DEVELOPER OR IN THE PRESENCE OF CORE DEVELOPER OR CLIENT.
2. Beta – A ltd number of end users carry out the bets testing before the delivery.