Staging Environment *(aka Pre-production Env)*

A "production like" environment to run performance tests and test usability and features.term-48;

Most of the companies perform UAT testing in this env

After UAT passes, the application will be deployed to production.

Example:
http://**stg**.amazon.com

Production Environment *(aka Operational Env)*

The environment for the actual system operation.

Also called the operational environment.

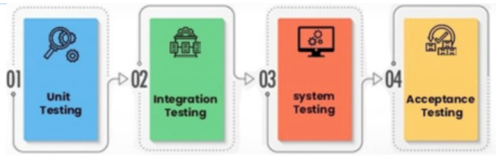
End-users use this Env.

Example:
http://amazon.com

Testing Hierarchy / Level

There are generally **4 recognized levels** of testing:

1. Unit testing
2. Integration testing
3. System testing
4. User Acceptance testing **(UAT)**



Unit Testing

Unit testing is software testing where Developers **test the individual unit of software** *(from a unit code perspective)*.

WHITE BOX TTESTING
WHO: Developers
ENV: Dev Env

```
public class UnitTest1 {
    [TestMethod]
    public void Test_AddMethod() {
            BasicMaths bm = new BasicMaths();
            double res = bm.Add(10, 10);
            Assert.AreEqual(res, 20);
```

Integration Testing

After unit testing, integration testing is done **to see whether the app's several modules work together** smoothly or **not**.

WHITE BOX TESTING
WHO: Developers
ENV: Dev Env

Example: Modules of an app: File, View, Help, Share
Integration test scenario: User login -> upload a file -> share the file. {file upload & share modules are tested to see if they work together}

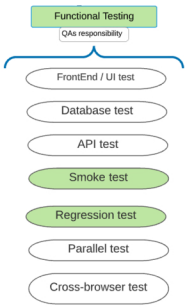
System Testing

It **tests the entire system** as one entity to ensure that it is working correctly.

System test is divided to:
Functional Testing & Non-functional Testings

GREY/BLACK BOX TESTING
WHO: *Functional Test* - QA/SDET
Non-functional Test - Performance Testers
ENV: QA Env

Functional Testing *(type of System Testing)*

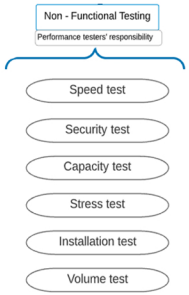


Functional testing is **to validate every software function** as per the functional requirements.

Testers test all **3 layers** of the application functionality daily. *(FrontEnd / UI, DataBase, and API).*

WHO: QA Autom. Egineers / SDET
ENV: QA Env

Non-Functional Testing *(type of System Testing)*



Non-Functional testing focuses on a software's **performance, security, volume**, load, etc., and parts not related to functionality.

WHO: QA Autom. Egineers / SDET
ENV: QA Env

Smoke Testing

Smoke testing is **performed every day** to determine whether the application is stable or not, and the QA Environment is up and running.

The **main goal** is validating the **critical** and **major** functionalities of the application.

Example:
70 scenarios to run as Smoke test every day at 6 AM
It takes 15 minutes to run.Testers receive emails to see the Smoke test **Cucumber** report from **Jenkins**

SOFT SKILLS PAGE 2 - BACK

Staging Environment	**Production Environment**	**Testing Hierarchy / Level**
Unit Testing	**Integration Testing**	**System Testing**
Functional Testing	**Non-Functional Testing**	**Smoke Testing**

SOFT SKILLS PAGE 2 - FRONT