

Classification of Testing - Testing Types, Testing Approaches, Testing Levels



Coherent Solutions

Better Teams. Better Results.

Summary

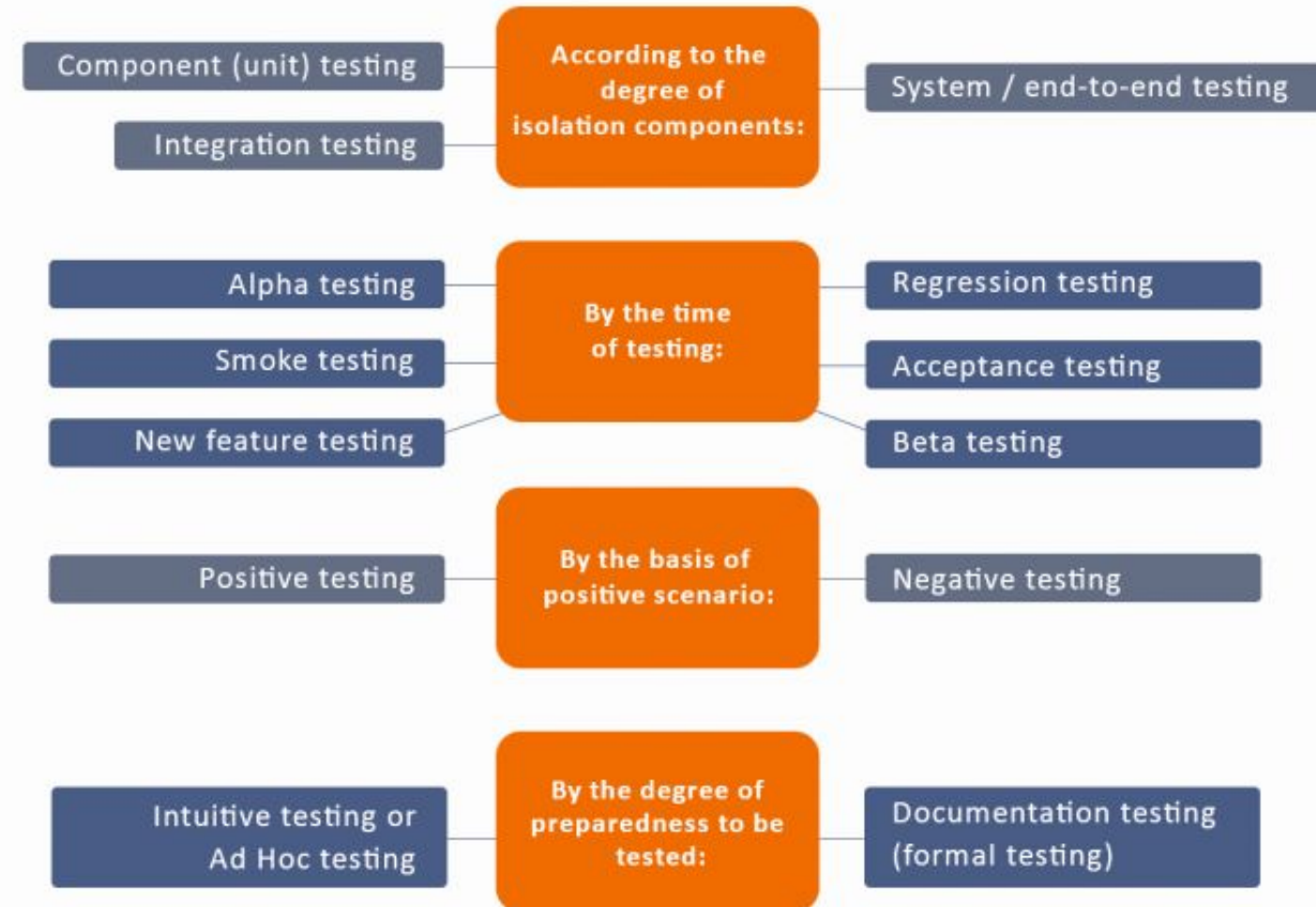
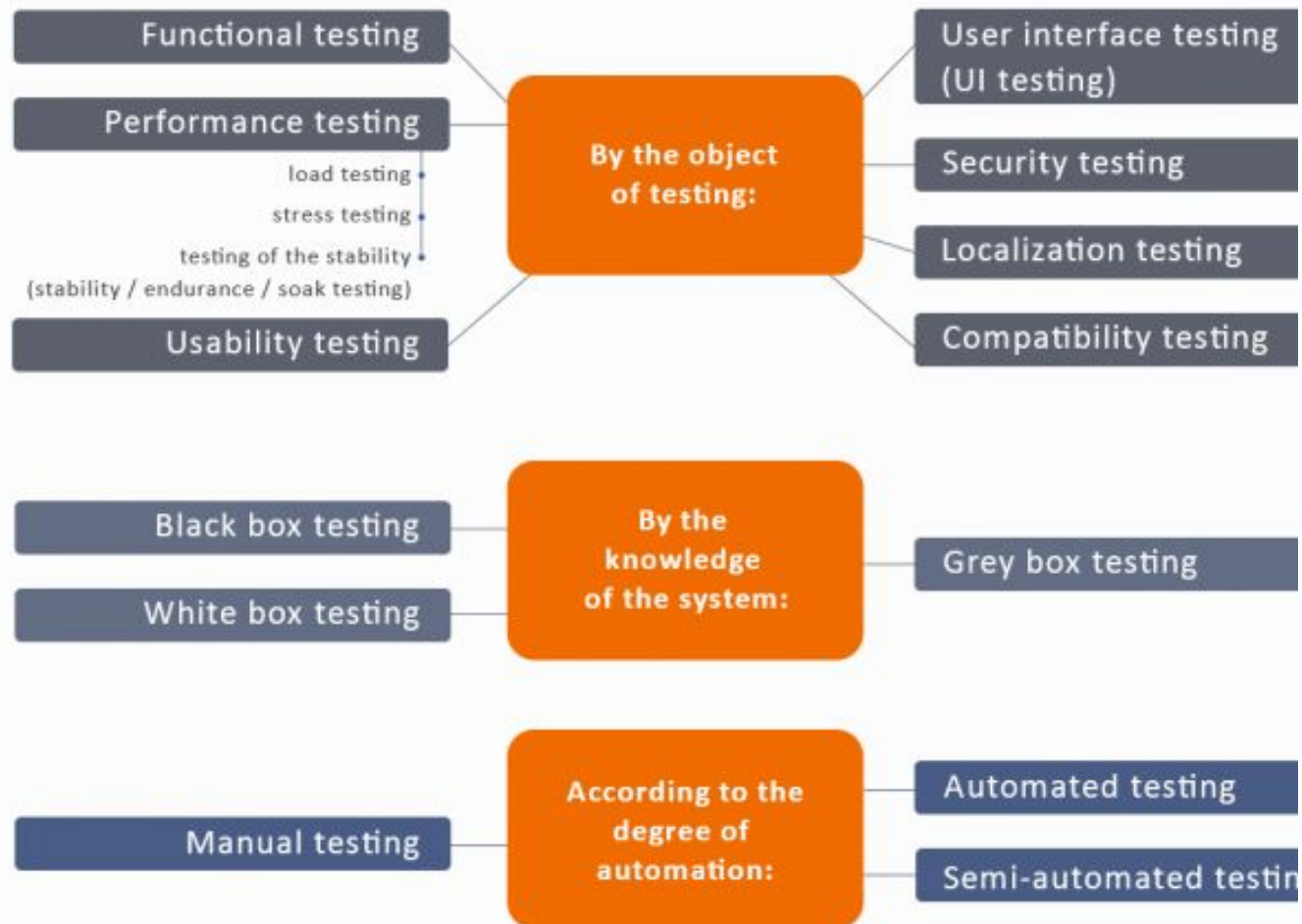
- What is testing?
- Testing types
- Test levels

What is Testing?

Definition of software testing: *“Software testing is a way to assess the quality of the software and to reduce the risk of software failure in operation.”*

Testing is the **process** consisting of all lifecycle activities, both static and dynamic, concerned with *planning, preparation and evaluation* of software products and related work products to determine that they satisfy specified requirements, to demonstrate that they are fit for purpose and to detect defects.

Types of Testing



Functional VS Non-Functional

Functional Testing vs Non-Functional Testing

Functional Testing

testing against functional requirements that describe the **functions** of a system and its components.

Non-Functional Testing

testing of a software application for its non-functional requirements: the way a system operates, rather than specific behaviours of that system.

GUI (Graphical User Interface) Testing

testing an application's visual elements, such as images, texts, buttons, etc., to validate their view and functional accuracy.

Usability (UX) Testing

focuses on the user's ease to use the application, flexibility in handling controls and the ability of the system to meet its objectives.

Configuration Testing

checks whether the software is capable of running on different hardware, operating systems, **browsers**, network environments or mobile devices.

Performance Testing

testing performed to determine the system parameters in terms of responsiveness and stability under various workload.

Performance Testing

Load Testing

determines a system's performance under **real-life load** conditions.

Stress Testing

determines the system on its robustness and error handling under **extremely heavy load** conditions.

Volume Testing

is carried out to find the response of the software with different sizes of the **data** being received or to be processed by the software.

Internationalization Testing

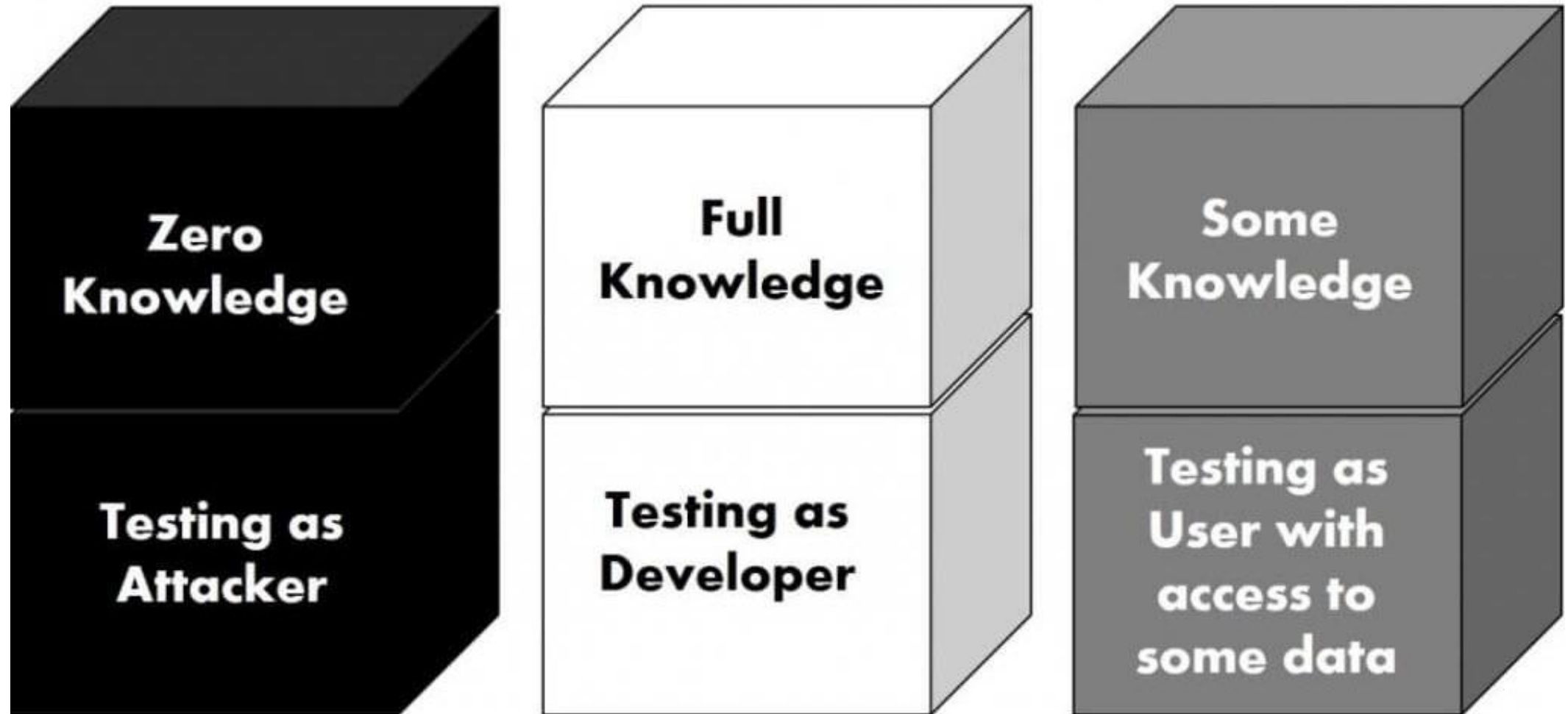
testing to ensure that application **can** function in any culture or locale (language, territory and code page).

Localization Testing

checks that software application **behaves** according to the local culture or settings.
The major area affected by localization testing includes **content and UI**.

Testing Types by approach (methods)

- Black Box
- White Box
- Gray Box



Testing Types by Testing Levels

Unit Testing/Component Testing

Component testing is a method where testing of each component in an application is done separately.

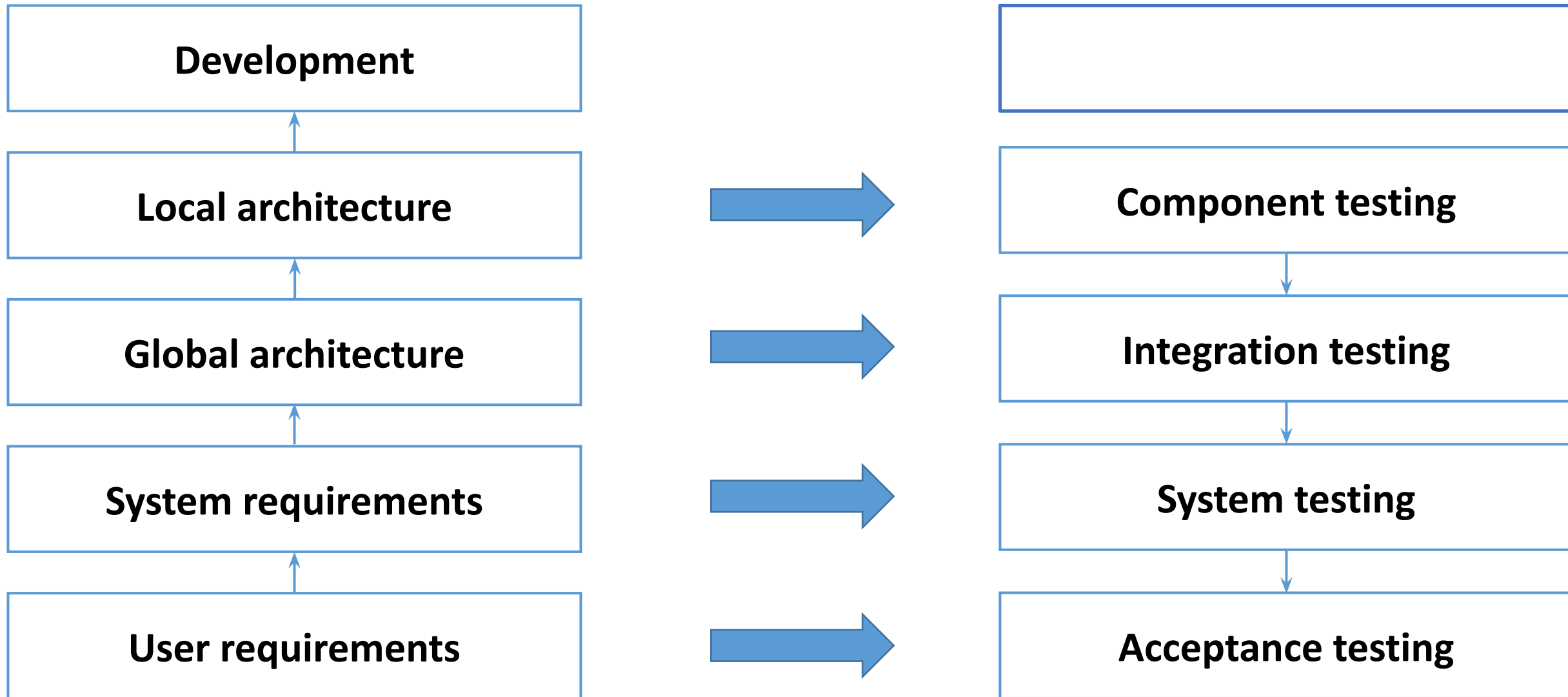
Integration Testing

Integration testing is one of the most common and important types of software testing. Tests integration between components, interactions to different parts of the system such as an operating system, file system and hardware or interfaces between systems. There are different approaches for Integration testing namely, Top Down, Bottom Up and Big Bang.

System Testing

System testing includes multiple software testing types that will enable to validate the software as a whole (software, hardware, and network) against the requirements for which it was built. Both functional and non-functional types of testing are carried out to complete system testing.

Testing Levels



Testing Types by Testing Levels

End-to-end Testing

End-to-end testing tests main flows across application e.g., right from order creation till reporting or order creation till item return etc. End-to-end testing is usually focused on mimicking **real life scenarios** and usage.

Acceptance Testing

Acceptance testing is a formal type of software testing that is performed by end user when the features have been delivered by developers. The aim of this testing is to check if the software confirms to their **business needs** and to the requirements provided earlier.

Testing Types by person who test

Alpha Testing

is done at the developers site. It is done at the end of the development process. Alpha testing is typically performed by a group that is independent of the design team, but still within the company.

Beta Testing

is done at the customers site. This is a formal type of software testing that is carried out by end customers before releasing or handing over software to end users. Successful completion of Beta testing means customer acceptance of the software.

Alpha VS Beta Testing

Main purpose	Test every user journey and ensure the application is working as expected.	Understand how real users interact with the application and test how it works in real-world conditions.
Done by	Internal team , potentially also end-customer.	Joe Public. Either via invite or via controlled release.
Visibility	White box. The testers know what is happening and what they are testing.	Black box. The testers just see the application and any associated help or tool-tips.
Structure	Rigorously structured. Every flow is tested and all results recorded and carefully analyzed.	Completely unstructured. Users are free to do what they want. Feedback is requested, but not required.
Performance		
Not interested in how the application or backend performs, simply in functionality.		Reliability and performance are key aspects of beta testing. Along with application security and stability.
Post-test-actions		
Any bugs identified are fixed. Any suggestions for small changes are analyzed and may be implemented. E.g. minor UI changes.		Show-stopper bugs are fixed. A/B testing may be conducted and result in a decision. But, in general, the results of feedback are used to influence the next version.
Duration		
This can be a long process , however, you should aim to keep it as short as possible (to avoid delays with your release).		Usually, this is a short process lasting a few weeks. However, some companies run betas that last for many years.

Testing Types by phase of testing

Retesting (Confirmation testing)

is a type of retesting that is carried out by software testers as a part of defect fix verification. For e.g. a tester is verifying a defect fix and let us say that there are 3 test cases failed due to this defect. Once a tester verifies the defect fix as resolved, the tester will then retest or test the same functionality again by executing the test cases that were failed earlier.

Regression Testing

is a type of software testing that is carried out by software testers as functional regression tests. The objective of regression tests is to find defects that got introduced to defect fix(es) or introduction of new feature(s). Regression tests are ideal candidates for automation.

Testing Types by phase of testing

Smoke testing

is a type of testing that is carried out by software testers to check if the new build provided by the development team is stable enough i.e., **major functionality** is working as expected in order to carry out further or detailed testing. Smoke testing is intended to find “show stopper” defects that can prevent testers from testing the application in detail. Smoke testing carried out for a build is also known as build verification test.

Sanity Testing

is brief or quick testing in order to ensure that the changes are working as expected and as per the specification documents. Sanity testing is focused on **one or two functionalities** whereas smoke testing is done to ensure that all the major features of the project is working fine. Sanity tests are narrow and most of the time sanity tests are not documented.

Testing Types by structure

Ad-hoc testing

Ad-hoc testing is performed on-the-fly without (clear) planning or the goal (e.g. Click here and there and see what's going on...).

The person performing Ad-hoc testing has a good understanding of the domain and workflows of the application to try to find defects and break the software.

Ad-Hoc Testing



Exploratory Testing

Exploratory testing is an informal type of testing conducted to learn the software at the same time looking for errors or application behavior that seems non-obvious. Exploratory testing is performed with a clear plan and structure in mind (e.g. Session-based testing is a good practice of exploratory testing)

