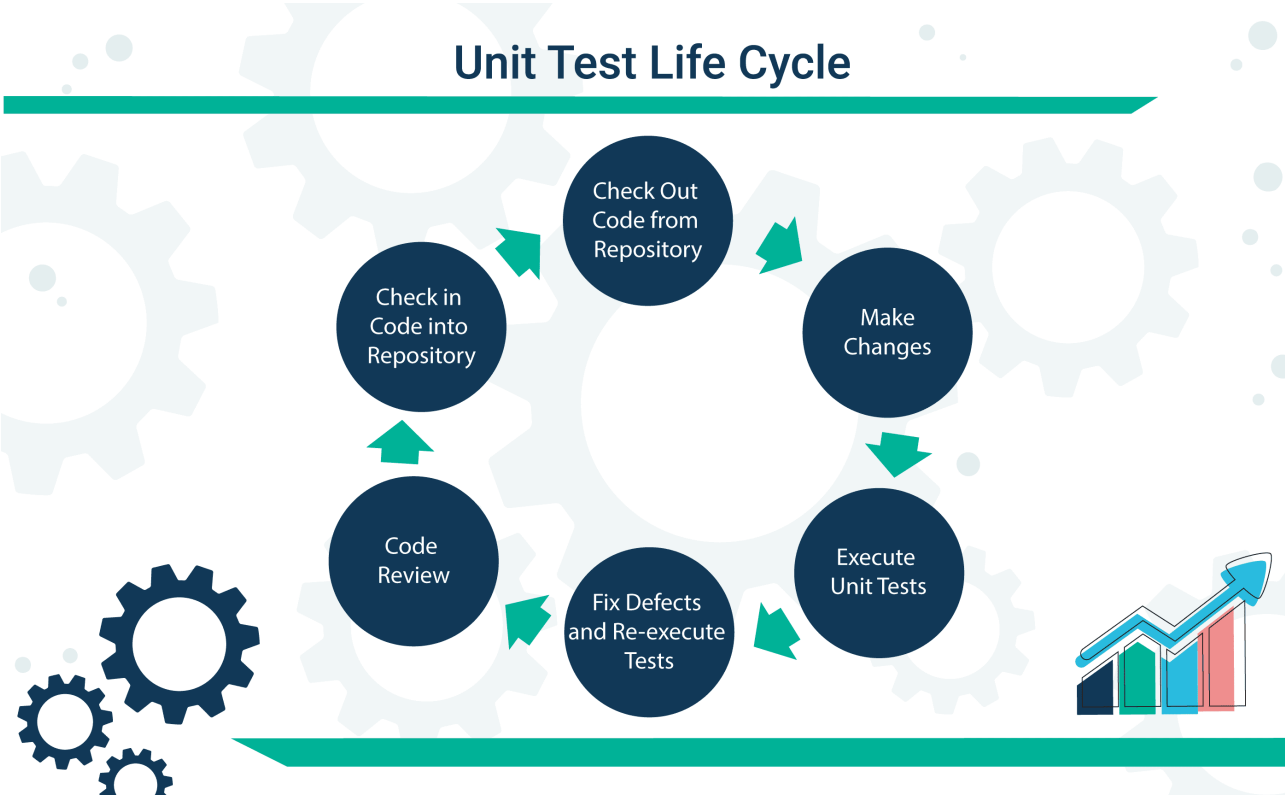
Unit Testing

Unit testing is very important in software development as it aims at delivering a better finished product by testing on the smallest portions of code: the units. Usually, this is a function, subroutine, a method or a property. This allows the tester to analyze the code, catch bugs faster (as opposed to wait till the completion of the program), and fix them quicker.. Unit tests can be (and usually are) conducted in an automated way.
Unit testing is an example of "white box testing", as it involves someone with in-depth knowledge of the code analyzed.

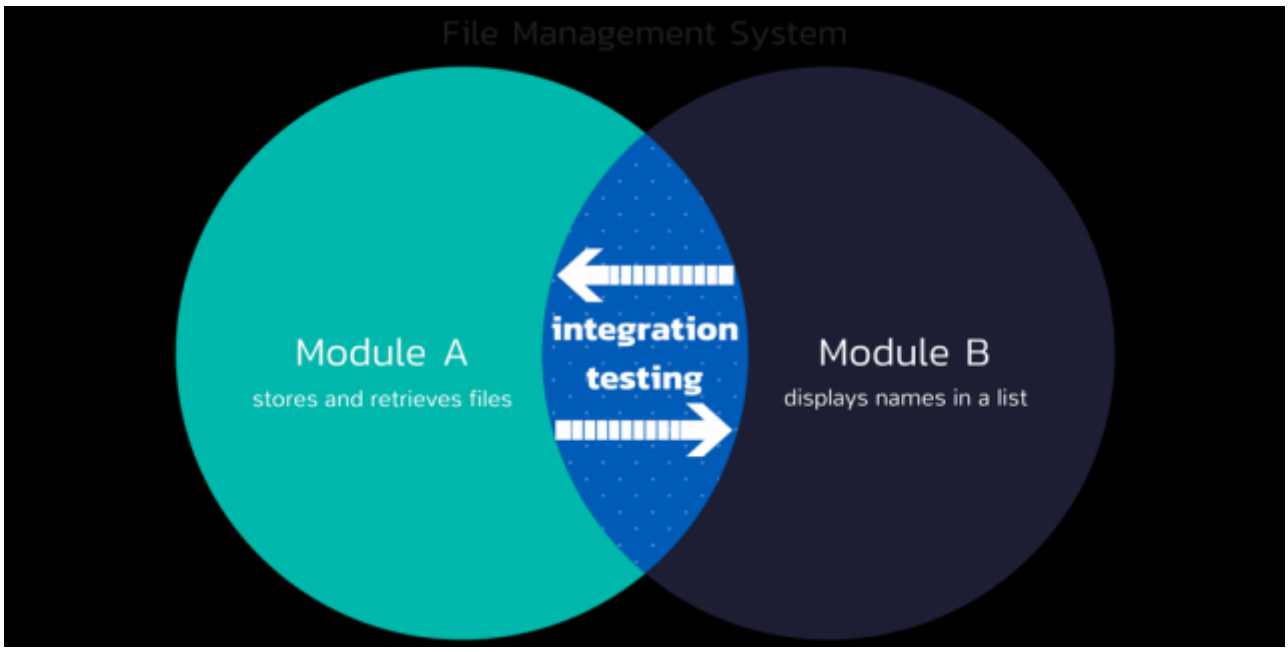


Read more about it on https://smartbear.com/learn/automated-testing/what-is-unit-testing/

Integration Testing

Integration testing is basically grouping a certain number of modules/components/units, following a precise logic, and then testing them, verifying that they work as a group as expected of them. More precisely, integration testing checks combinations, interactions and the way these groups fuse into one system.

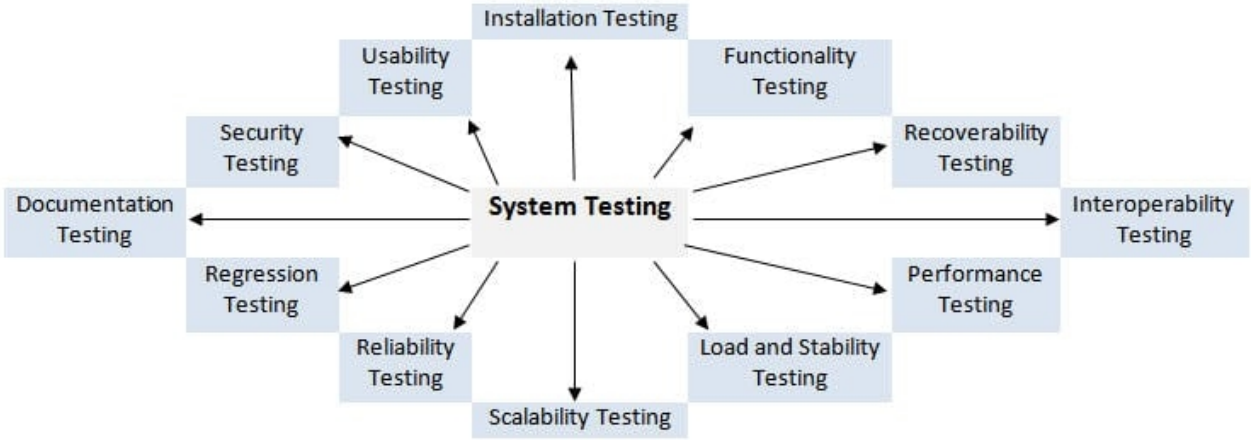
Integration testing is an example of "black box testing", as the tester interacts with the program through its units and graphical interface, without the need of inside knowledge of the code.



More informations on: https://www.guru99.com/integration-testing.html

System Testing

Another "black box testing", system testing is involves the testing of a complete program. It's the last test to check that the software meets the requirements and it tests both functional and non-functional areas.



System Testing - © www.SoftwareTestingHelp.com

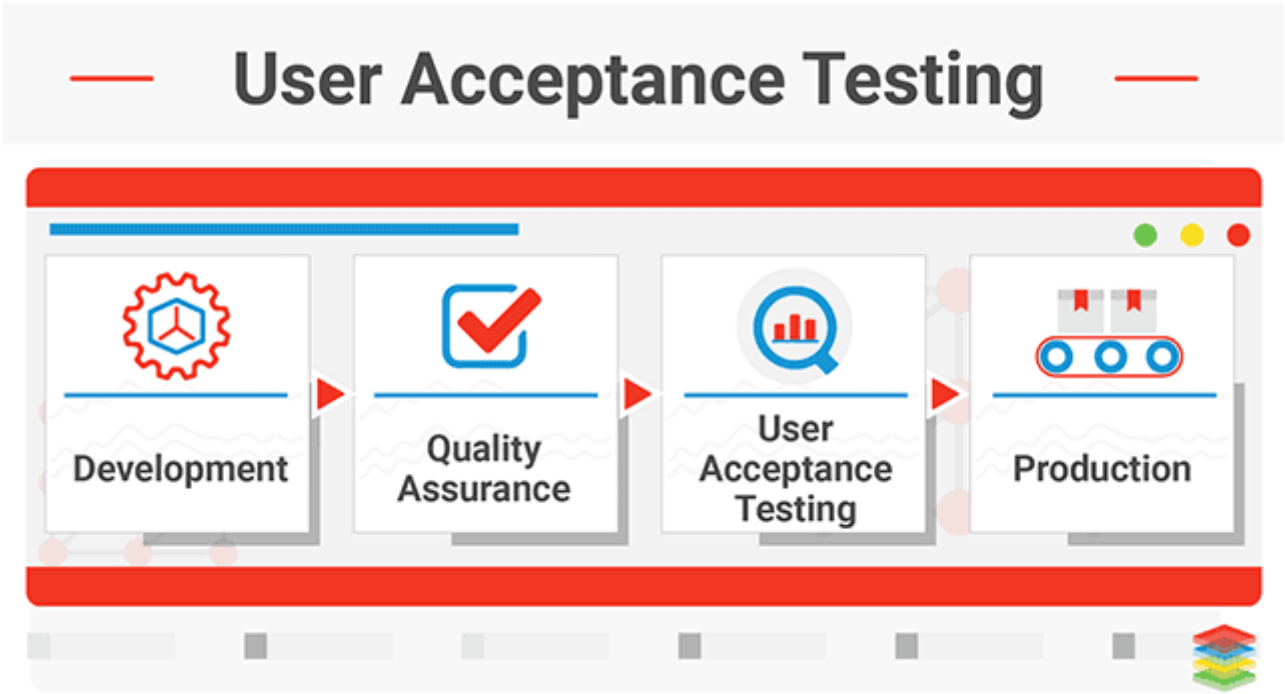
If you want to learn more: https://www.softwaretestinghelp.com/system-testing/

Acceptance Testing

Acceptance testing (again a type of "black box") is the first phase in the testing processes to involve users, to determine that there are no defects in the finished product. As such, the result of the test could only be a fail or a pass. During this testing, both user and business requirements are determined to be satisfied or not.

Sometimes, acceptance testing is divided in its sub-phases, such as user acceptance testing, business

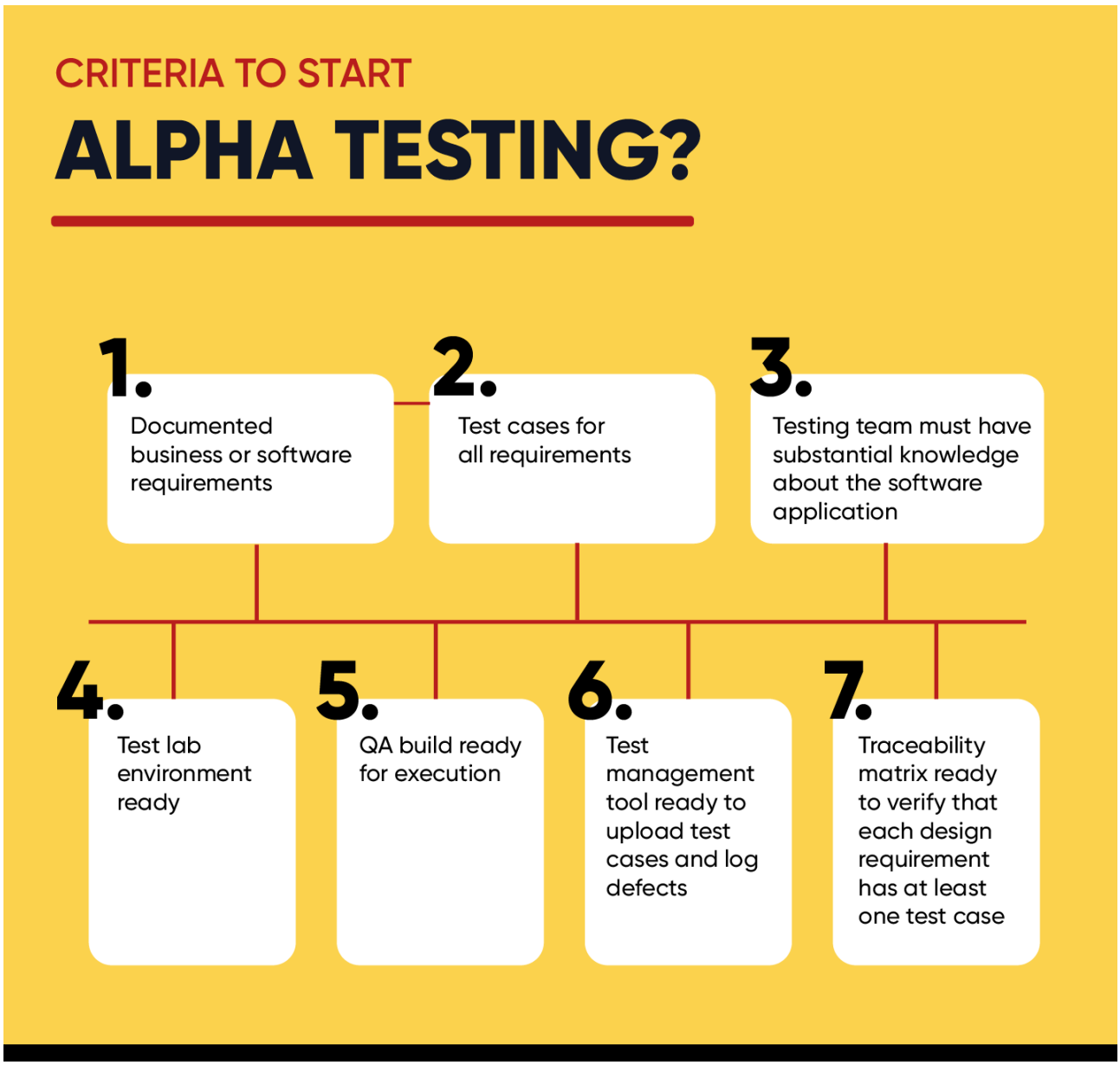
acceptance testing, alpha testing and beta testing (more on these below).



More on this: https://softwaretestingfundamentals.com/acceptance-testing/

Alpha Testing

A type of acceptance testing, alpha testing is performed by a group of end-users or sometimes by actual testers. It's usually “black box”, though it could rarely be “white box”. It's usually carried out internally, possibly by employees of the organization which created the software. The task is to simulate a normal user experience, as to find bugs and issues. This type of testing is done during the later stages of developement, whilst the program is not yet completed.

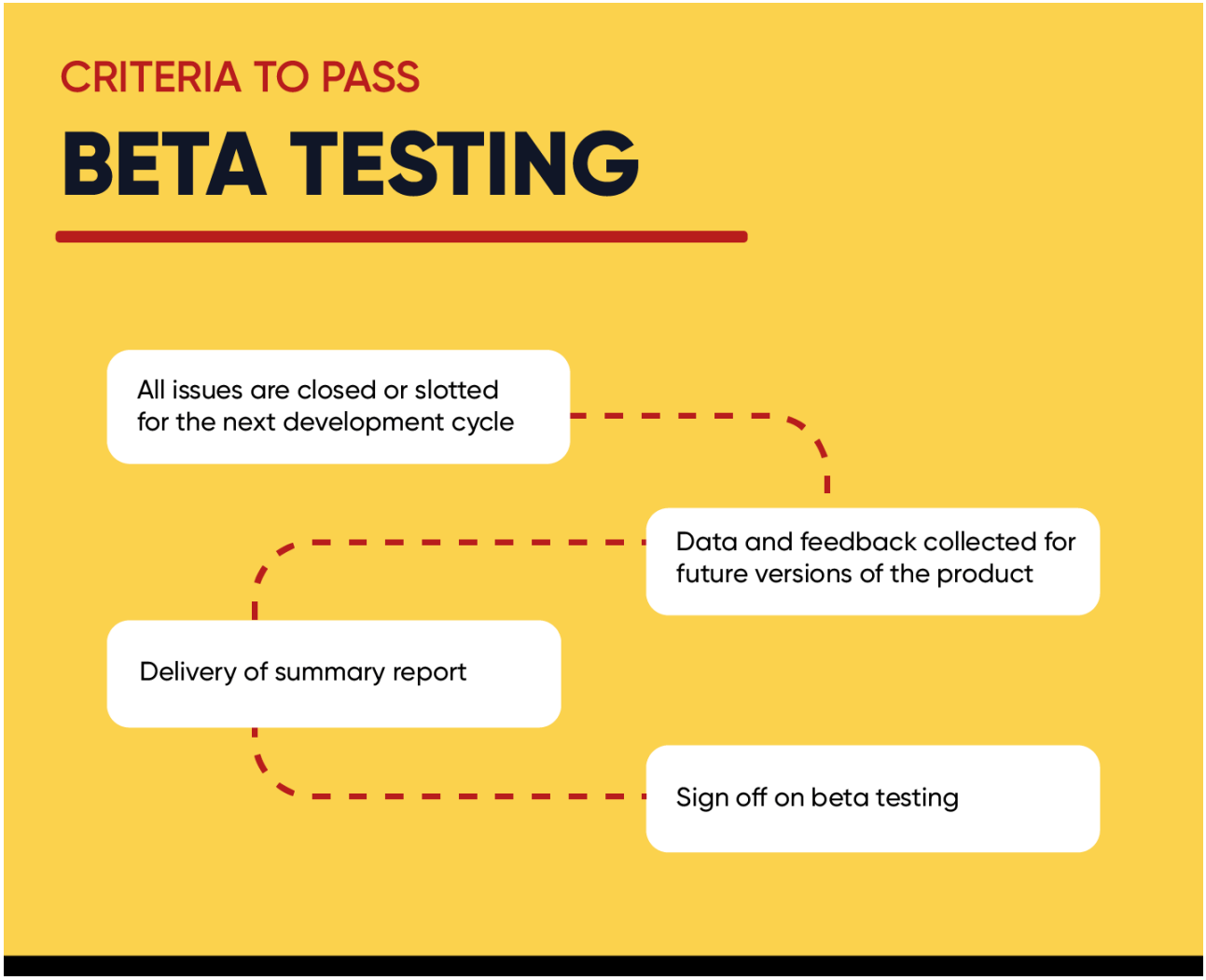


Find more informations here: https://www.javatpoint.com/alpha-testing

Beta Testing

The last phase of testing before publishing the end product, this “black box” method is done on a pratically completed software, with just minor re-touches to add, depending mostly on what comes out of the testing. The duty to find bugs and glitches is left completely on the users, with no testing left to actual tester whatsoever.

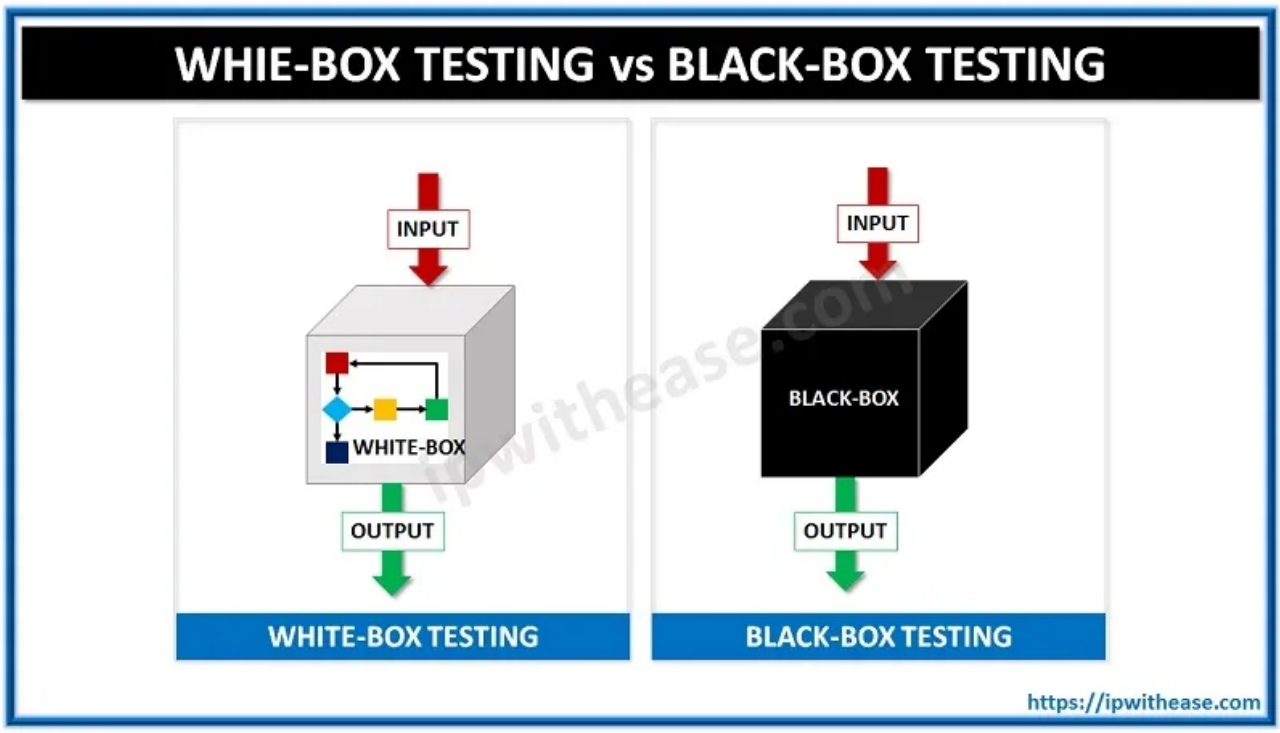
Beta testing could be either open (accessible to anyone) or closed (fixed number of beta-testers and maybe stricter requirements, like operative system, age range, etc.).



You can find more on: https://xd.adobe.com/ideas/process/user-testing/everything-you-need-to-know-about-beta-testing/

Black and White Box Testing

Let's delve a bit more into black and white box testing and what they exactly mean.



Black Box Testing:

In this method, the testers don't know the internal structure and workings of the software. Normally, this type of test isn't carried out by the coders who produced the program. As such, no programming knowledge is required. It's possible to apply this method to higher levels of testing, like system testing. The primary reason for black box testing is to check the functionality of the software.

White Box Testing:

White box testing, as an opposite to the black box one, requires for the tester to be familiar with the internal structure of the software. For this reason, it's usually done by the actual developers, who have the necessary programming knowledge. This type of testing is used for lower levels of testing, like unit testing. The focus of white box testing is the code.

More on these topics here: https://www.guru99.com/back-box-vs-white-box-testing.html