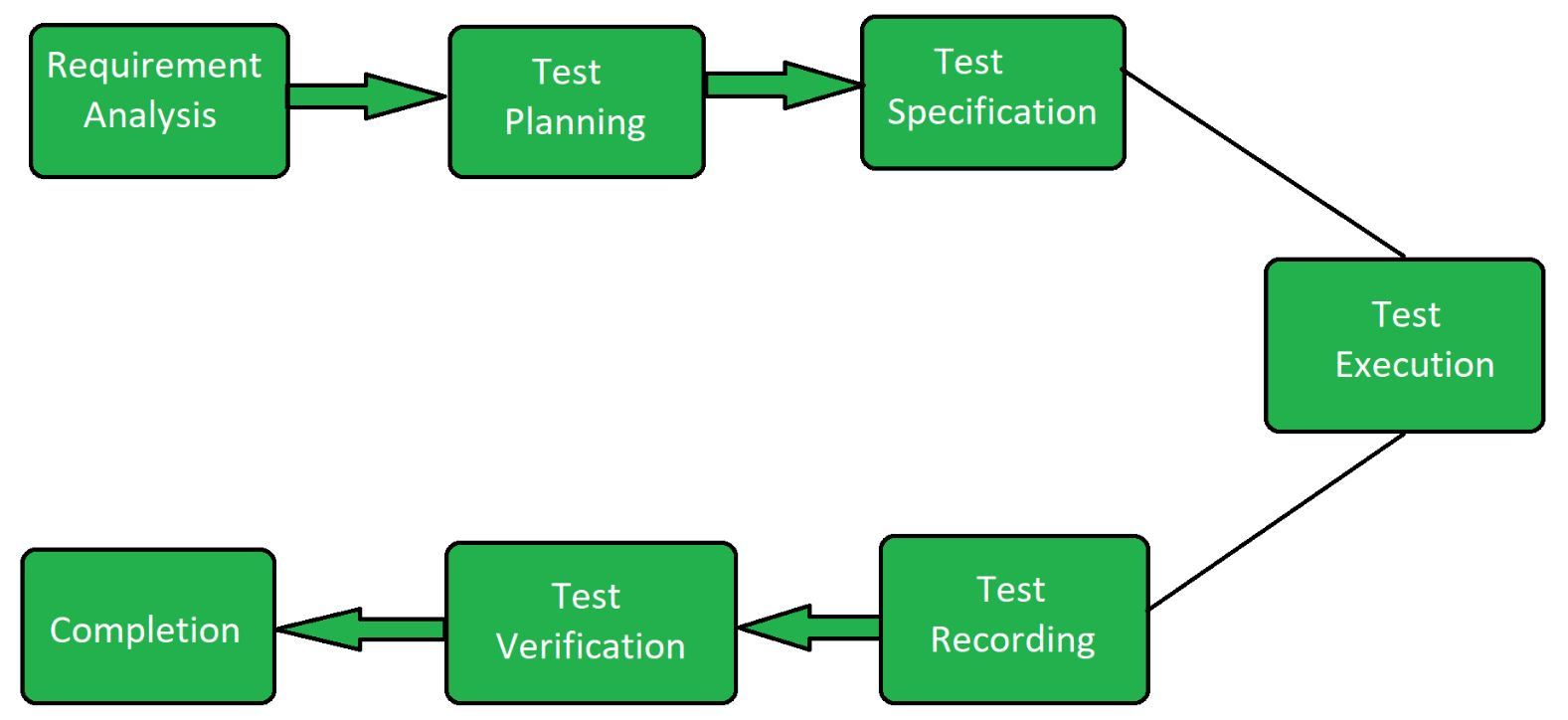
**StarLabs 2022 - Documentation**

***Component Testing***

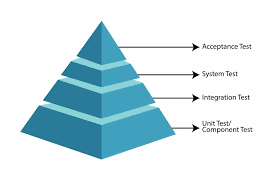
Component testing is a sub-category of software testing in which each individual component of the software is separately tested without integrating with other components. It is also known as module testing, when viewed from an architecture perspective. A software, in general, is made up of several components. Component testing addresses these components individually. It is one of the most frequent type of black box testing. Component testing requires a test strategy and test plan, where we consider each part of the software individually. We define a test scenario for each component, which is further divided into high level test cases, low level detailed test cases with prerequisites.



***Objective of Component Testing***

The primary objective of performing component testing is to check whether the input/output behaviour of the test object or component is working as expected. Component testing ensures that the functionality of the component is working properly and as specified.

1. **To reduce risks** − Component testing validates each component of the software and helps identify bugs in the software code and fix them.
2. **Discover bugs in the component** − Component testing helps identify bugs in the source code of the software. Also, it validates control flow, functions, data structure, etc., used in the program.
3. **Validate the functional as well as non-functional performance of the component** − Component testing ensures that the system design and specifications are performing as expected and specified.
4. **To improve quality of the component** − Component testing is essential in developing and maintaining confidence in the components; fewer bugs in the additional testing; thus, improving the quality of the software.
5. **To prevent bugs from escaping to higher testing levels** − Developers identify the coding errors and fix them as early as possible. Thus, it reduce the existence of bugs in the higher levels of testing.



***The most common reason for different perception of Component testing are:***

1. Type of Development Life Cycle Model Chosen
2. Complexity of the software or application under test
3. Testing with or without isolation from rest of other component in software or application.