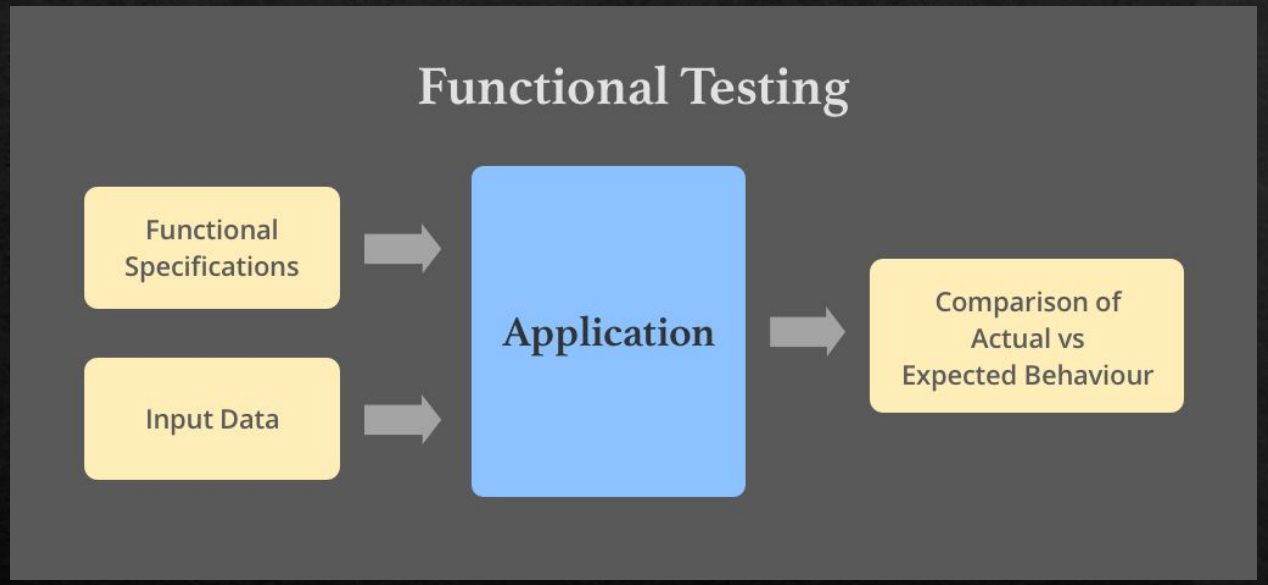
Functional Testing

What is functional testing?

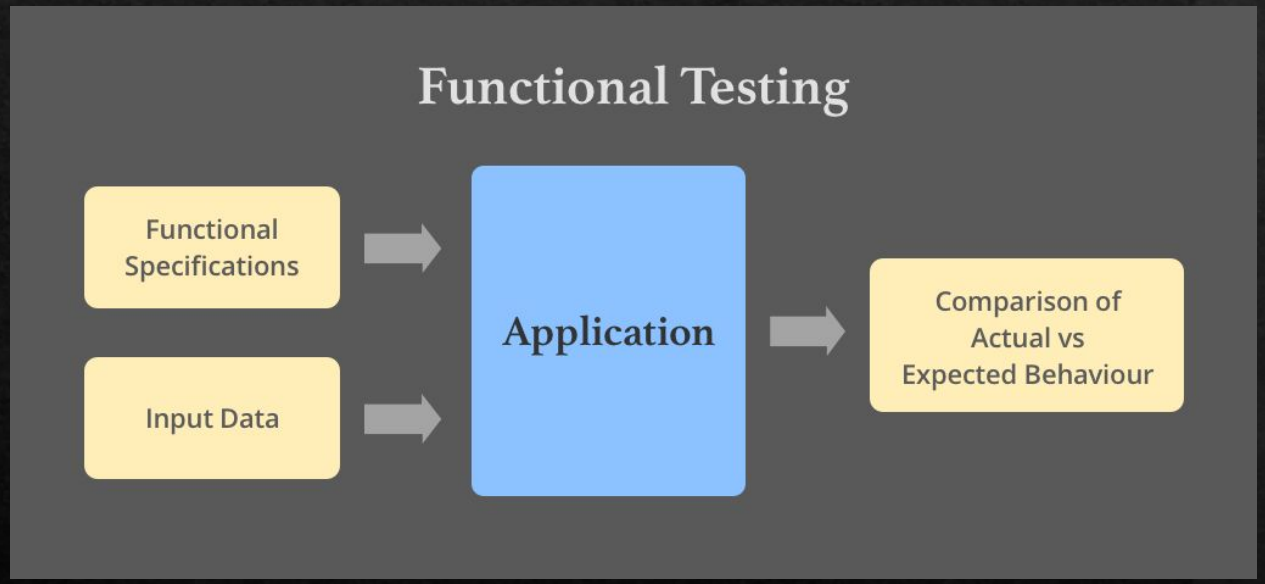


The task of functional testing is to verify that the software features meet the functional requirements.

Simply put, expectations and reality should match.

Only if every feature of a software system works correctly, it can pass a functional test.

What is functional testing?



Mainly involves Blackbox testing

It is not concerned about the source code of the application

Each and every functionality of the application is tested by giving a proper input . And verifying the output

Comparing the actual results with the expected result.

When To Perform Functional Testing?

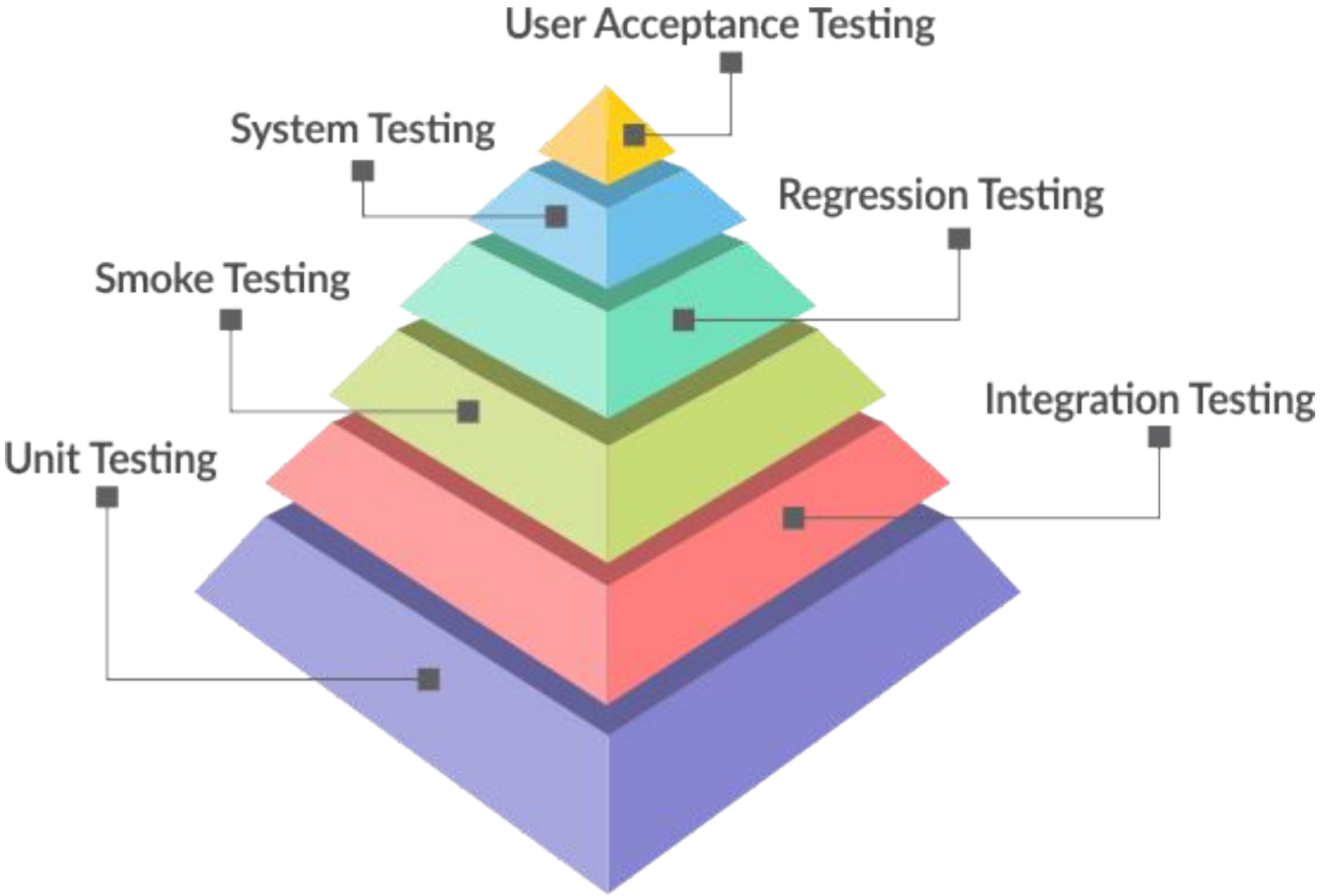
- When you need to verify the requirements specified in the documentation.

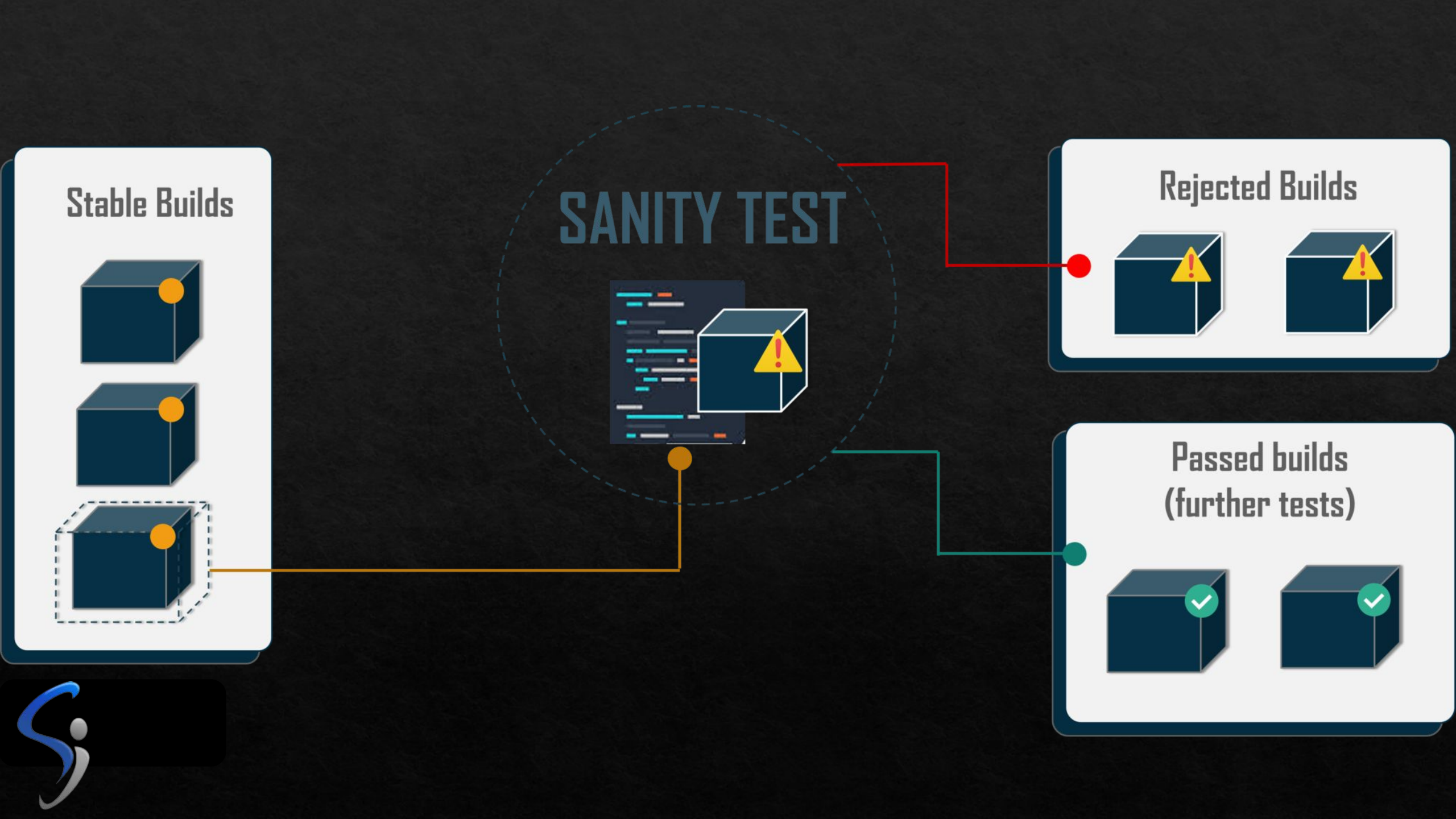
- When you need to verify that the app provides expected business processes.

Advantages Of Functional Testing

- It ensures that the customer or end- user is satisfied.
- It produces a defect-free product/software.
- It ensures that all the requirements should be met.
- It ensures the proper working of all the functionalities of an application/software/product.
- It ensures that the software/product works as expected.
- It improves the quality of the product.

Types Of Functional Testing




Stable Builds
SANITY TEST
Rejected Builds
Passed builds
(further tests)

Sanity Testing

- *Is a part of functional testing*
- *Which is conducted after receiving a software build, with minor changes in the code, or functionality.*
- *The aim is to make sure that the bugs have been fixed and to confirm that there are no further issues introduced due to the new changes.*

Contd...

- Used to verify the correctness and rationality of the software.
- Testing the major functionalities in detail in the last stage before giving it to the client
- To ensure previous functionalities are working properly.

Sanity Testing Example

Web Application

Multiple Modules

New Feature Implemented

Sanity Test is Performed

Smoke Testing

- **Smoke Testing** is a software testing process that determines whether the deployed software build is stable or not. Smoke testing is a confirmation for QA team to proceed with further software testing. It consists of a minimal set of tests run on each build to test software functionalities. Smoke testing is also known as “Build Verification Testing” or “Confidence Testing.”

OR

Testing the major functionalities in each release to ensure the stability of the application.

Smoke Testing

Crucial Functionality

Vs

Added Functionality

Sanity Testing

Smoke Testing	Sanity Testing
Smoke testing helps in determining if further rigorous testing is needed or not and hence saves time.	Sanity testing is only conducted if there is enough time. It is mainly used to check the functionality of a program once minor edits are done.
Smoke testing mainly includes testing of all the major parts though not deeply.	Mainly focusing on narrow testing, in Sanity testing only checks few areas of functionality.
It ensures whether the most crucial functions are working properly or not. However, this test does not bother finding the fine details.	This test is performed to check or prove that the application is a functioning considering all its specifications.

Adhoc

Testing

- Buddy Testing
- Pair Testing
- Monkey Testing

ADHOC TESTING

Buddy Testing :

- Conducted with minimum 2 people
- Done after unit testing is completed
- Identifying the bugs and report in the same time in the same module
- Team - One software developer and one tester

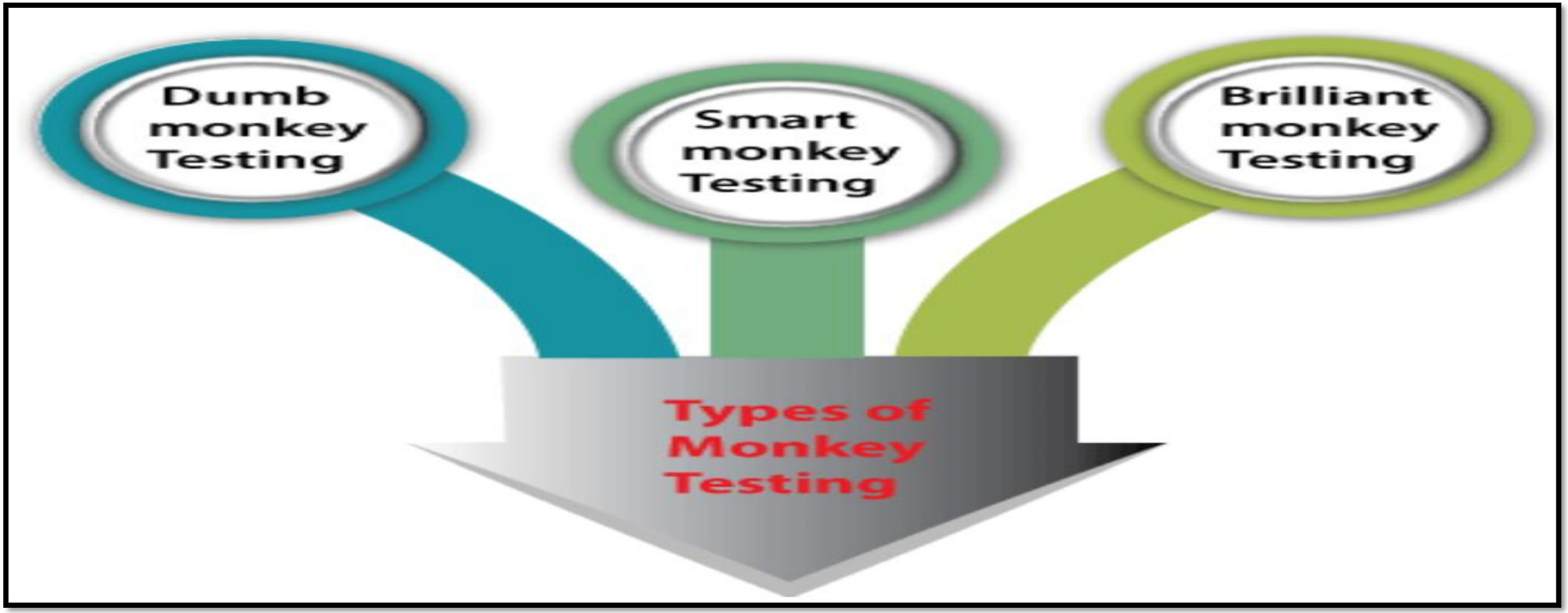
Monkey Testing

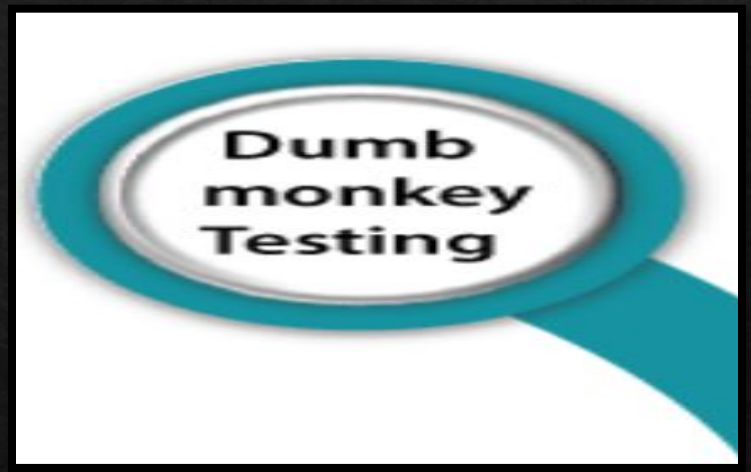
- Testing the application in random manner like a monkey jumping without any order or procedure
- Also known as Random Testing

Features of Monkey Testing

- Subsequently, **Monkey Testing** includes testing the software or application by giving some random data and detecting whether the system fails or not.

- The key intent of executing the monkey testing is to identify the defects and errors in the software and make sure that the system does not crash once the entire development on the software product is done.


Dumb monkey Testing
Smart monkey Testing
Brilliant monkey Testing
Types of Monkey Testing



The first type of monkey testing is **Dumb Monkey testing**. It is the easiest and direct type of monkey testing that does not know their or system's abilities or the application's flow.

Eg: In this, the **test Manager or lead** appoints a test engineer who does not have the knowledge of particular module of an application in order test the product.

The next type of monkey testing is **Smart Monkey testing**, wherein the **test engineer is entirely attentive of the system or the application.** The smart monkey tests are aware of where the pages of the application will redirect to.

The last and third type of monkey testing is Brilliant Monkey Testing. In this type of monkey testing, the test engineer has a **good knowledge of the system**. The test engineer knows about the pattern of using the product, and henceforth, they can perform testing from the user's viewpoint.

Advantages and Disadvantages of Monkey Testing

- New kinds of bugs
- Easy to execute
- Less skilled people
- Less costly

- No bugs can be reproduced
- Time consuming
- Fewer bugs
- Less accuracy

Pair Testing

- Conducted by 2 people
- 2 testers are paired based on the experience and knowledge
- So different idea on a particular problem
- Helps one tester to do the test case execution and other to report it

Variations of Adhoc	Who conducts it?	How?
Buddy testing	developer + QA	QA engineer has to perform the necessary checks, and the developer has to fix all the defects in the early stages.
Pair testing	QA + QA	QA engineers check one module and help each other simultaneously. For example, one can look for defects, and the other one to document them.
Monkey testing	QA	QA engineer works alone to find all the bugs and inconsistencies with the planned behavior.

GORILLA
TESTING

Gorilla Testing

- Gorilla Testing is a Software testing technique where a module of the program is repeatedly tested to ensure that it is working correctly and there is no bug in that module.
- A module can be tested over a hundred times, and in the same manner. So, Gorilla Testing is also known as “Frustrating Testing”.

Monkey Testing

VS

Gorilla Testing

Monkey Testing	Gorilla Testing
Monkey testing is a type of software testing which is performed based on some random inputs without any test cases and checks the behavior of the system and confirms whether it crashes or not.	Gorilla Testing is a type of software testing which is performed on a module based on some random inputs repeatedly and checks the module's functionalities and confirms no bugs in that module.
This testing is performed on entire system.	While this testing is performed on few selective modules of the system.
The main objective of Monkey Testing is to check whether system crashes or not.	The main objective of Gorilla testing is to check whether the module is working properly or not.
There are three types of Monkey Testing i.e. Dumb Monkey Testing, Smart Monkey Testing and Brilliant Monkey Testing.	While there is no such different types of Gorilla Testing available.
Monkey testing is also known as Random Testing, Fuzz Testing or Stochastic Testing.	Gorilla Testing is also known as Torture Testing, Fault Tolerance Testing or Frustrating Testing.

NEGATIVE TESTING

Negative Testing

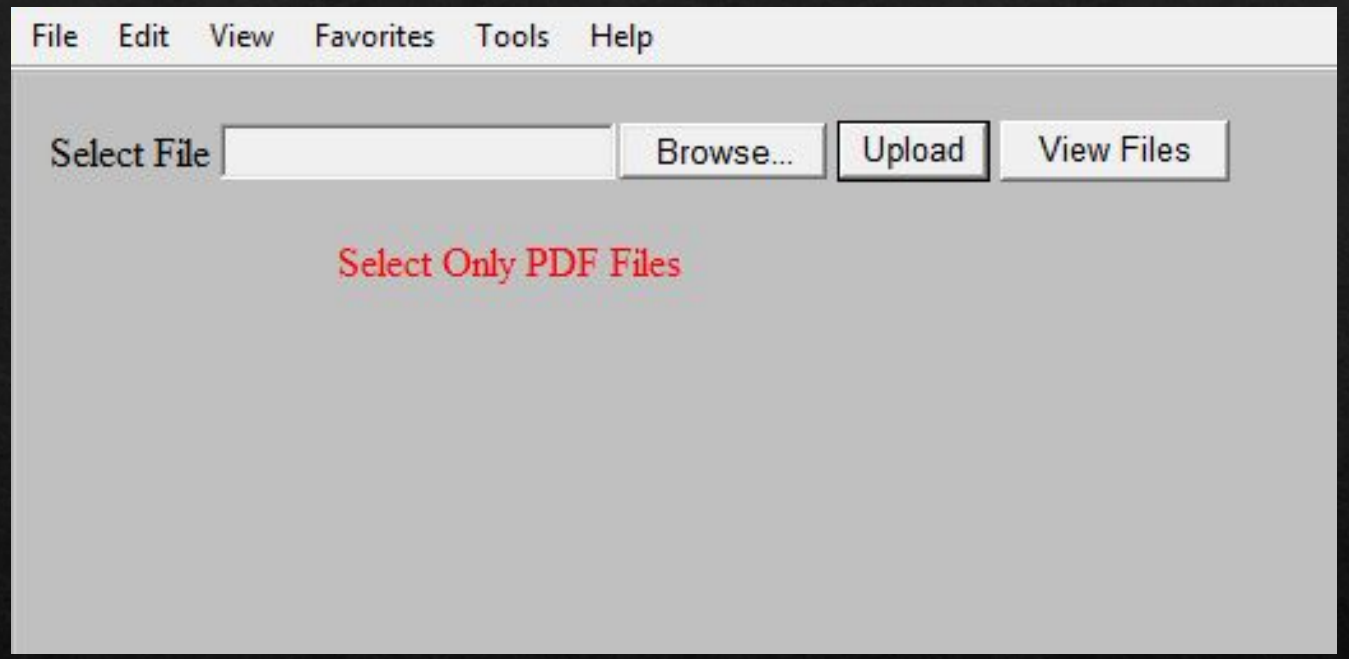
- **Negative Testing** is a software testing type used to check the software application for unexpected input data and conditions. Unexpected data or conditions can be anything from wrong data type to strong hacking attack. The purpose of negative testing is to prevent the software application from crashing due to negative inputs and improve the quality and stability.
- By just doing positive testing we can only make sure our system is working in normal conditions. We have to make sure that our system can handle unexpected conditions to ensure a 100% fault-free system.


1

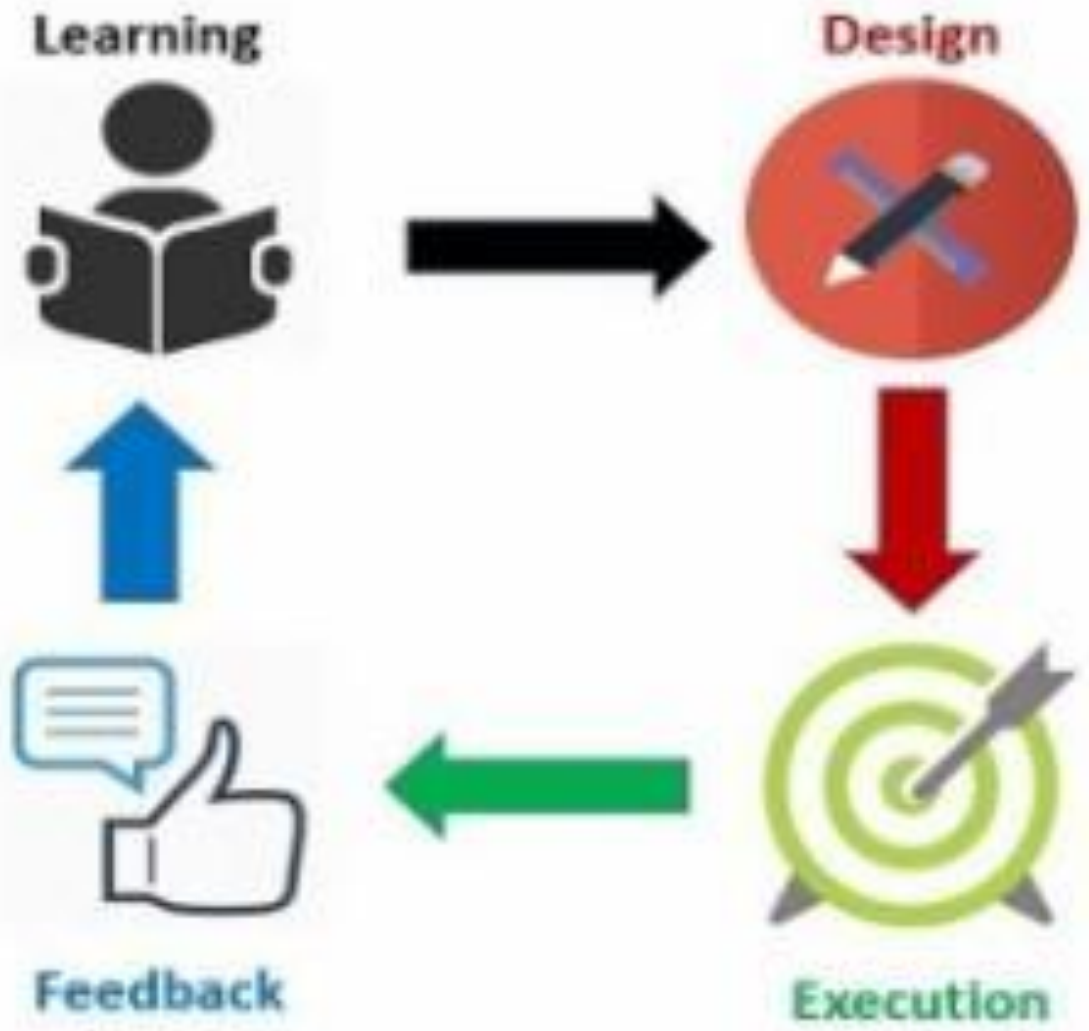
Enter Only Numbers

abcdef

Negative Testing



Exploratory Testing



Exploratory Testing

- It is the simultaneous process of test planning, test case preparation, test case execution and bug reporting.
- All done at the same time

What is Retesting?

Retesting

Testing the fixed bug or testing the failed test case after a bug fix


REGRESSION
TESTING