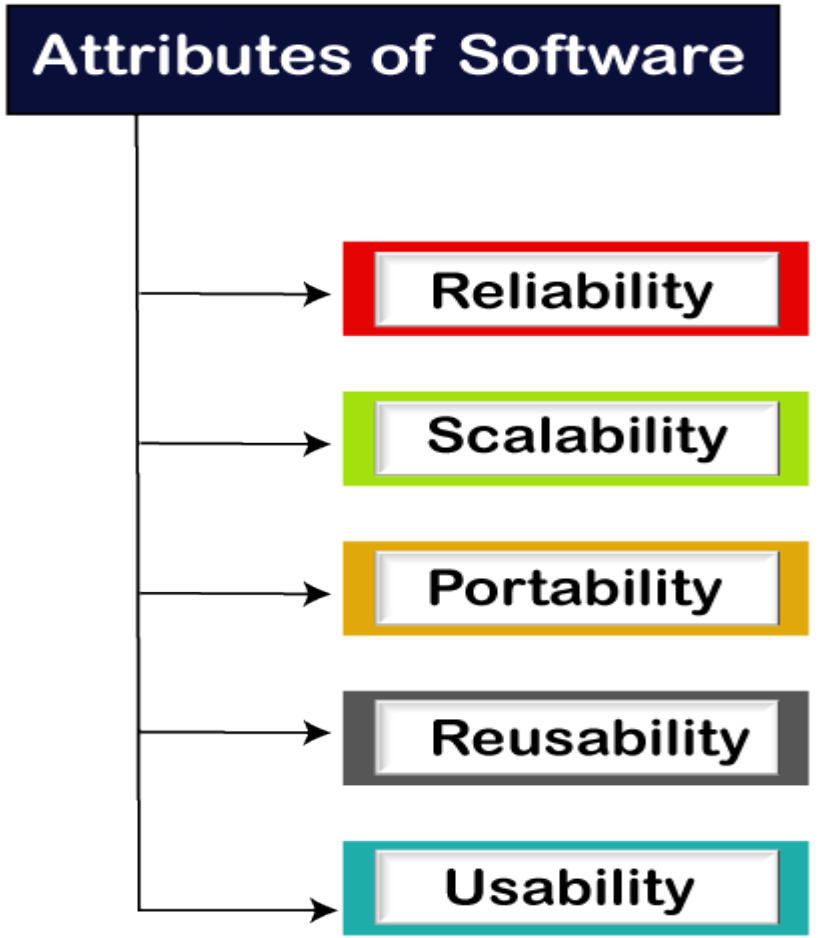
Software Testing

By

Madumathi

What is Software Testing?

- Software testing is a process of identifying the correctness of software by considering its all attributes (Reliability, Scalability, Portability, Re-usability, Usability) and evaluating the execution of software components to find the software bugs or errors or defects.



Significance of Testing

Some of the reasons why testing become a very significant and integral part of the field of information technology are as follows.

1.Cost-effectiveness

2.Customer Satisfaction

3.Security

4.Product Quality

Principles of software testing

There are seven principles in software testing:

1. Testing shows the presence of defects
2. Exhaustive testing is not possible
3. Early testing
4. Defect clustering
5. Pesticide paradox
6. Testing is context-dependent
7. Absence of errors fallacy

Types of Testing

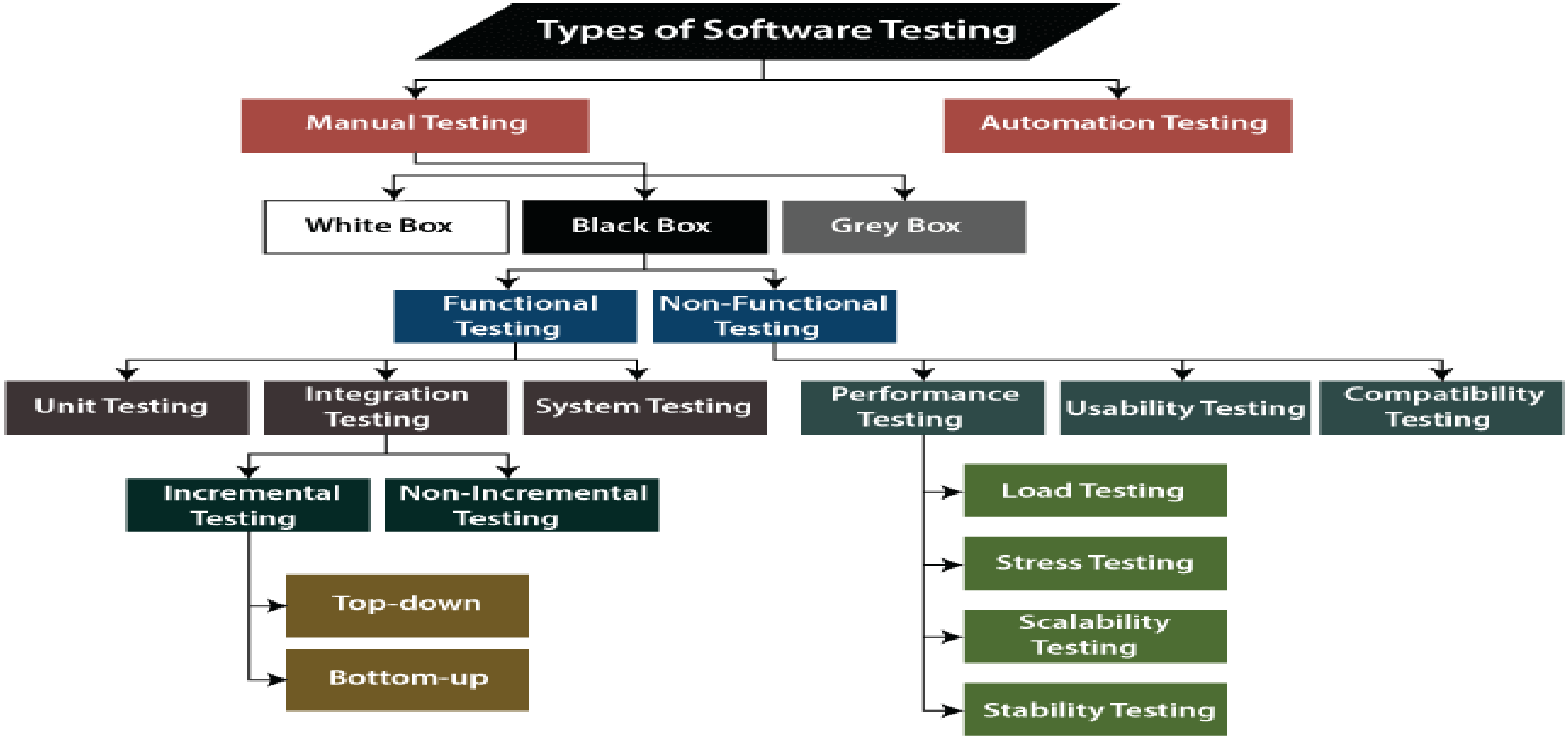
Software Testing can be broadly classified into two types:

1. **Manual Testing:** Manual testing includes testing software manually, i.e., without using any automation tool or any script. In this type, the tester takes over the role of an end-user and tests the software to identify any unexpected behaviour or bug. There are different stages for manual testing such as unit testing, integration testing, system testing, and user acceptance testing.

Testers use test plans, test cases, or test scenarios to test software to ensure the completeness of testing. Manual testing also includes exploratory testing, as testers explore the software to identify errors in it.

2. **Automation Testing:** Automation testing, which is also known as Test Automation, is when the tester writes scripts and uses another software to test the product. This process involves the automation of a manual process. Automation Testing is used to re-run the test scenarios quickly and repeatedly, that were performed manually in manual testing.

Apart from regression testing, automation testing is also used to test the application from a load, performance, and stress point of view. It increases the test coverage, improves accuracy, and saves time and money when compared to manual testing.


Types of Software Testing
Manual Testing
Automation Testing
White Box
Black Box
Grey Box
Functional Testing
Non-Functional Testing
Unit Testing
Integration Testing
System Testing
Performance Testing
Usability Testing
Compatibility Testing
Incremental Testing
Non-Incremental Testing
Top-down
Bottom-up
Load Testing
Stress Testing
Scalability Testing
Stability Testing

Testing Approaches

There are three types of software testing approaches.

1. White Box Testing:

It is also called Glass Box, Clear Box, Structural Testing. White Box Testing is based on the application's internal code structure. In white-box testing, an internal perspective of the system, as well as programming skills, are used to design test cases. This testing is usually done at the unit level.

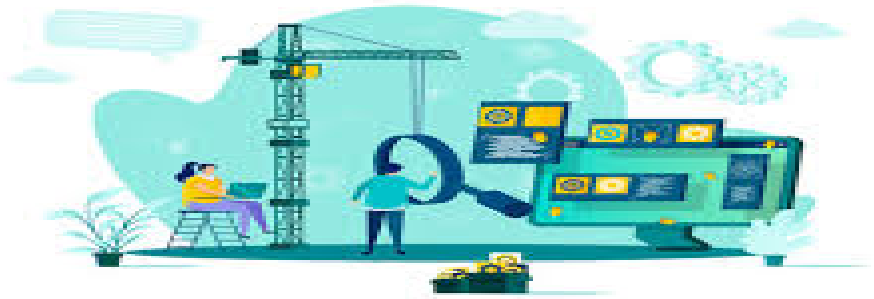
2. Black Box Testing:

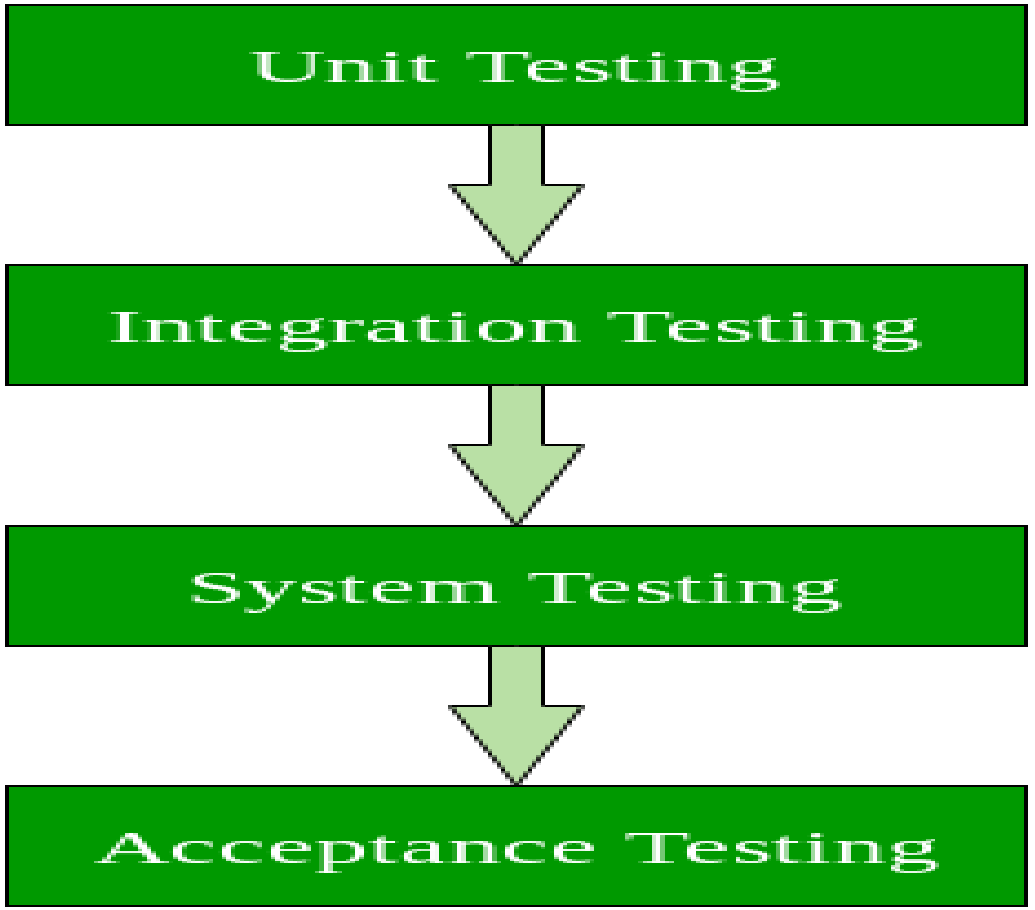
It is also called Behavioural/Specification-Based/Input-Output Testing. Black Box Testing is a software testing method in which testers evaluate the functionality of the software under test without looking at the internal code structure.

3. Grey Box Testing:

Grey box is the combination of both White Box and Black Box Testing. The tester who works on this type of testing needs to have access to design documents. This helps to create better test cases in this process.

Testing levels





Some examples of Functional and Non-functional Testing

Examples of Functional testing are

- Unit Testing
- Smoke Testing
- Sanity Testing
- Integration Testing
- White box testing
- Black Box testing
- User Acceptance testing
- Regression Testing

Examples of Non-functional testing

- Performance Testing
- Load Testing
- Volume Testing
- Stress Testing
- Security Testing
- Installation Testing
- Penetration Testing
- Compatibility Testing
- Migration Testing

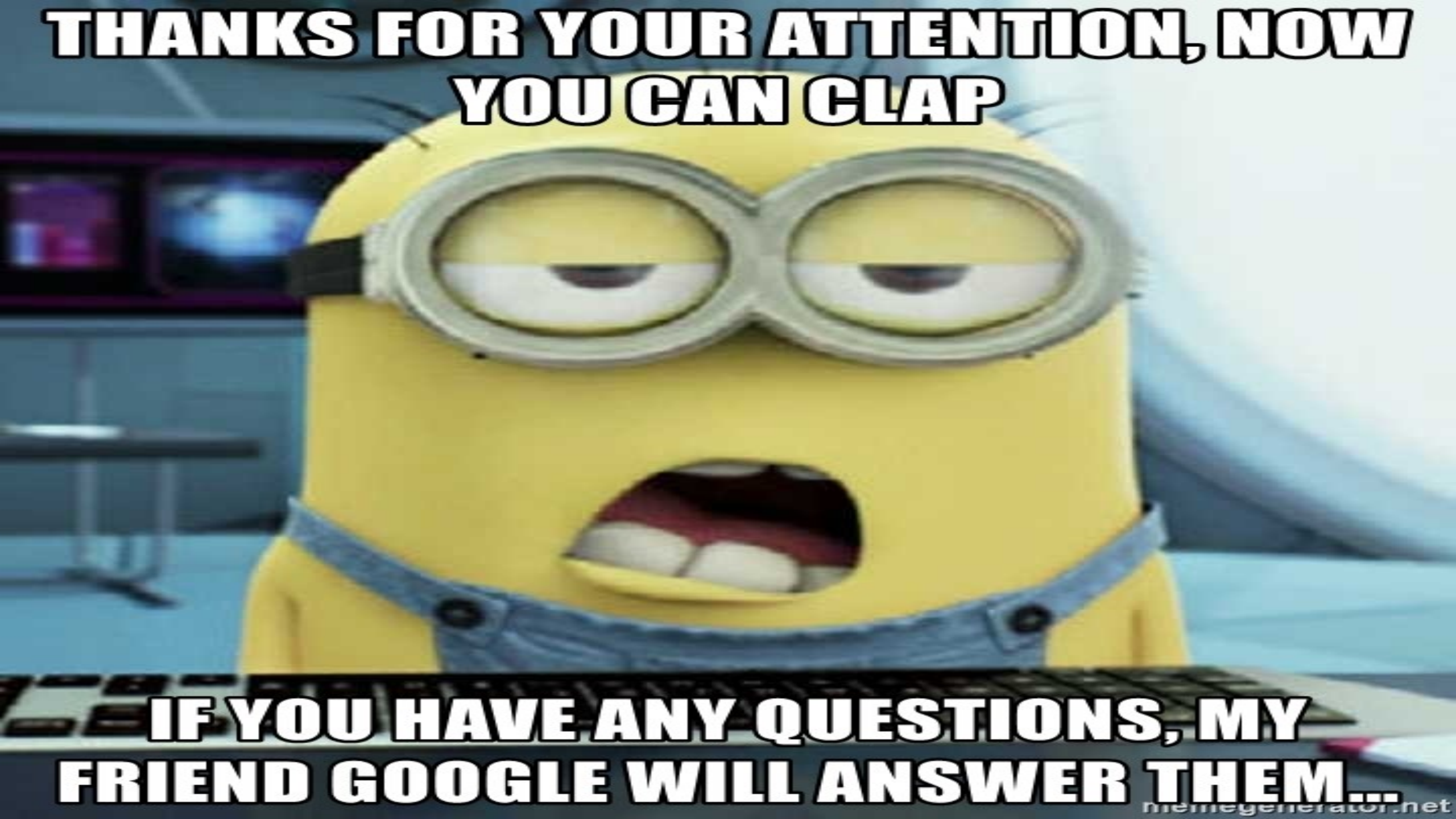
Testing Strategies

Here are important strategies

- Unit Testing: This software testing basic approach is followed by the programmer to test the unit of the program. It helps developers to know whether the individual unit of the code is working properly or not.
- Integration testing: It focuses on the construction and design of the software. You need to see that the integrated units are working without errors or not.
- System testing: In this method, your software is compiled as a whole and then tested as a whole. This testing strategy checks the functionality, security, portability, amongst others.

Resources

- https://www.softwaretestingmaterial.com/software-testing/
- https://www.geeksforgeeks.org/software-engineering-testing-guidelines/?ref=lbp
- https://www.guru99.com/software-testing-introduction-importance.html
- https://www.guru99.com/testing-tools.html
- https://www.guru99.com/functional-testing.html
- https://www.javatpoint.com/types-of-software-testing


THANKS FOR YOUR ATTENTION, NOW YOU CAN CLAP
IF YOU HAVE ANY QUESTIONS, MY FRIEND GOOGLE WILL ANSWER THEM...
memegenerator.net