

# Introduction to Software Testing

Himanshu Arya

# Agenda

1. Introduction to Software Testing
2. Manual vs. Automation Testing
3. SDLC & STLC, Validation & Verification, QA vs. QC
4. TestOps & Future Trends in Testing
5. Types of Testing (Functional & Non-Functional)
6. Performance Testing & Its Types
7. Agile, Waterfall, and V-Model
8. Defect Tracking & Bug Lifecycle
9. Test Plan, Test Cases, and Test Scenarios
10. Practical Task – Xenonstack Website Testing
11. Conclusion

# Introduction to Software Testing

## **Definition of Software Testing:**

*Software Testing is a systematic process aimed at evaluating the functionality of software application to ensure it meets the specified requirements and works as intended.*

## **Purpose: Identify Bugs, Ensure Quality, Improve User Experience:**

*The primary objective of software testing include detection defects early, ensuring quality standards, and enhancing the end-user experience, which collectively contribute to customer satisfaction and product reliability.*

## **Types: Manual & Automation Testing:**

*Testing can be broadly categorized into Manual Testing, Where human testers execute test cases without automation tools, and Automation Testing, Where scripts are used to execute tests, thus increasing efficiency and repeatability.*

# Manual vs. Automation Testing

## **Manual Testing:**

*Conducted by Human testers, this approach allows for exploratory testing, adaptability, and human intuition in scenarios where automation might struggle to replicate complex user interactions.*

## **Automation Testing:**

*Utilizes software scripts to run tests automatically, making it a preferred choice for repetitive tasks, as it dramatically increase test execution speed and accuracy over time.*

## **Tools: Selenium, JUnit, TestNG:**

*Various tools facilitate both manual and automated testing, choices for automation, while tools like TestNG serve to enhance the testing process by offering flexible test configuration and execution features.*

# SDLC & STLC

## Software Development Life Cycle:

*SDLC consists of a precise plan that describes how to develop, maintain, replace, and enhance specific software.*

*SDLC consist of 6 Phases:*

- *Requirement Analysis*
- *Software Design*
- *Software Build*
- *Testing*
- *Deployment*
- *Maintenance*

## Software Testing Life Cycle:

*STLC is a systematic approach to testing a software application to ensure that it meets the requirements and is free of defects.*

*SDLC consist of 5 Phases:*

- *Test Planning*
- *Test Case Development*
- *Test Environment Setup*
- *Test Execution*
- *Test Closure*

# Validation & Verification, QA Vs. QV

## Validation & Verification:

*Validation involves assessing whether the right product has been built, focusing on user needs and requirements, while Verification entails ensuring that the product is built correctly through rigorous testing.*

## QA Vs. QV:

*Quality Assurance emphasizes proactive measures to prevent defects in software, while Quality Control focuses on identifying and fixing defects through testing, representing two complementary but distinct philosophies in maintaining software quality.*

# TestOps & Types of Testing

## What is TestOps?

*TestOps is an evolving methodology that integrates testing processes into DevOps lifecycle, fostering collaboration, continuous testing and automation to enhance overall software delivery speed and quality.*

## Functional Testing :

*Encompasses various testing methods such as Unit testing, Integration Testing, System Testing, and User Acceptance Testing (UAT) to validate that the software functions according to specified requirements.*

## Non-Functional Testing :

*Focuses on aspects like performance, security, usability, and compatibility, ensuring that software meets outlined quality standards beyond just functional requirements.*

# Types of Testing

## Smoke Testing:

*It quickly verify that the core functionalities of the application are working as expected after a build or release, acting as a quick sanity check.*

## Alpha Testing:

*Alpha testing is internal, conducted by developers and QA within a controlled environment, focusing on functionality and usability.*

## Beta Testing:

*Beta testing is external, involving real users in real-world conditions to gather feedback and identify issues before release.*



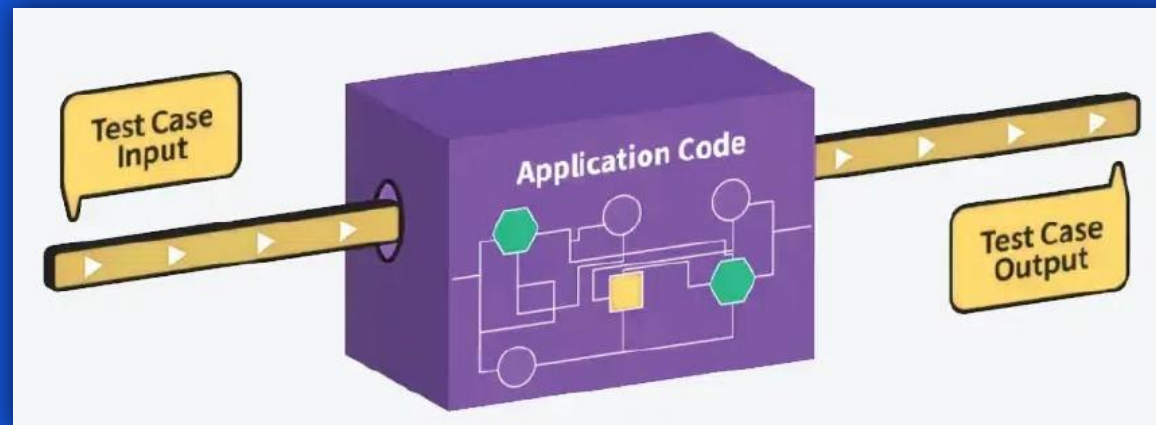
# White Box vs Black Box

## White Box

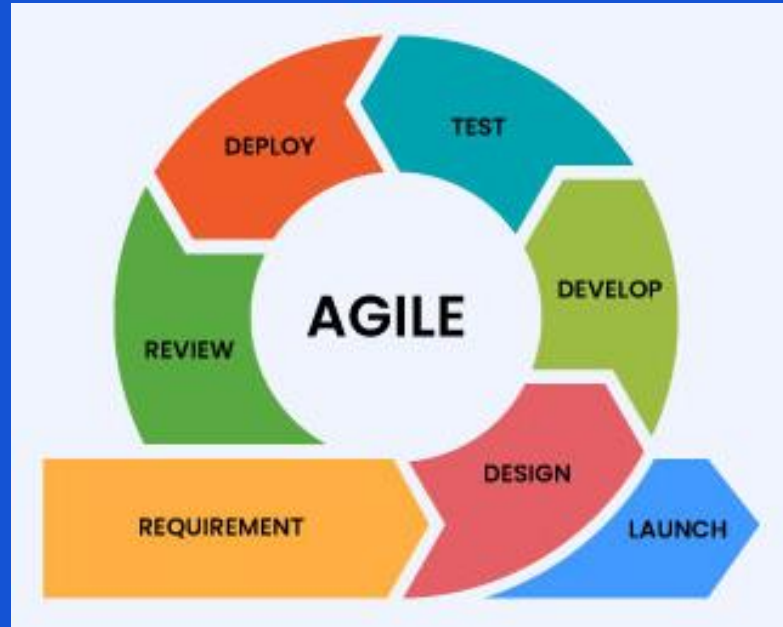
- Performed after knowing the internal structure of the software.
- Can be executed by developers or testers.
- Easy to automate.

## Black Box

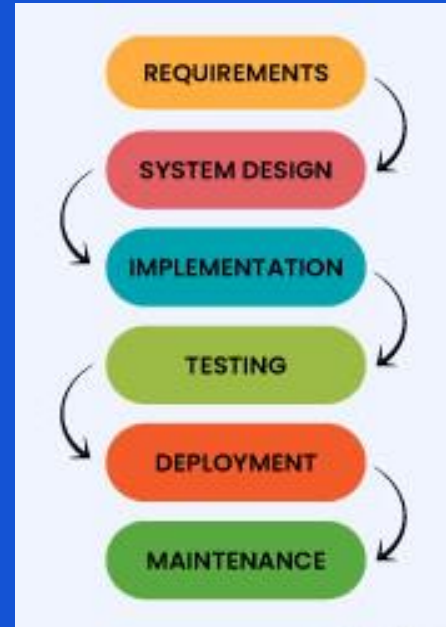
- Performed without any idea about the internal structure.
- Mostly executed by testers.
- It's tough to automate.



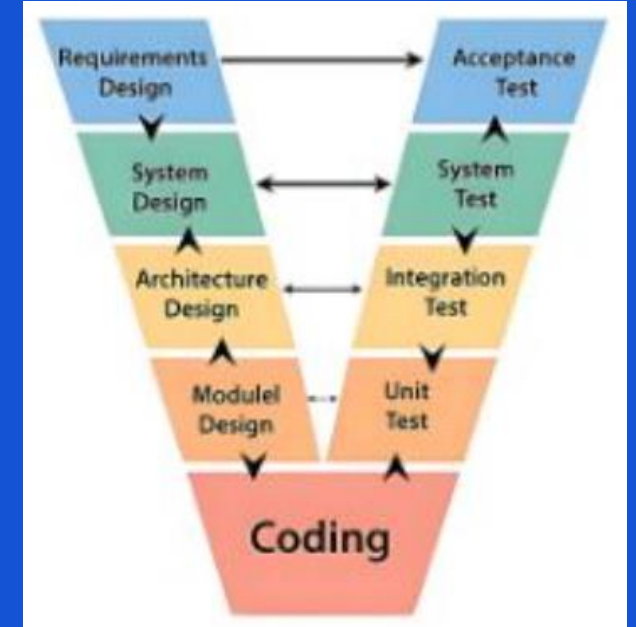
# Agile , Waterfall & V-Model



Characterized by its iterative and flexible approach, Agile supports continuous testing and integration, enabling rapid adaptations to changing project requirements and fostering collaborative team environments.



This linear and structured methodology is traditionally structured, with each phase dependent on the completion of the previous one, making it well-suited for smaller, less complex projects.



The V-model showcases the testing phase occurring parallel to development, emphasizing validation and verification at each stage, which enhances quality assurance throughout the dev life cycle.

# Test Plan, Test Cases, & Test Scenarios

## Test Plan

A Test Plan outlines the strategy, scope, resources, and schedule for the testing process, serving as the foundation for the testing efforts and aligning team goals with project objectives.

## Test Cases

Test Cases provide a detailed list of steps required to validate functionalities of the software, specifying inputs and expected outcomes for checking correctness.

## Test Scenarios

High-level descriptions of what to validate within the software, outlining various paths a user may take through the application, ensuring coverage of diverse usage cases.

**Thank You**