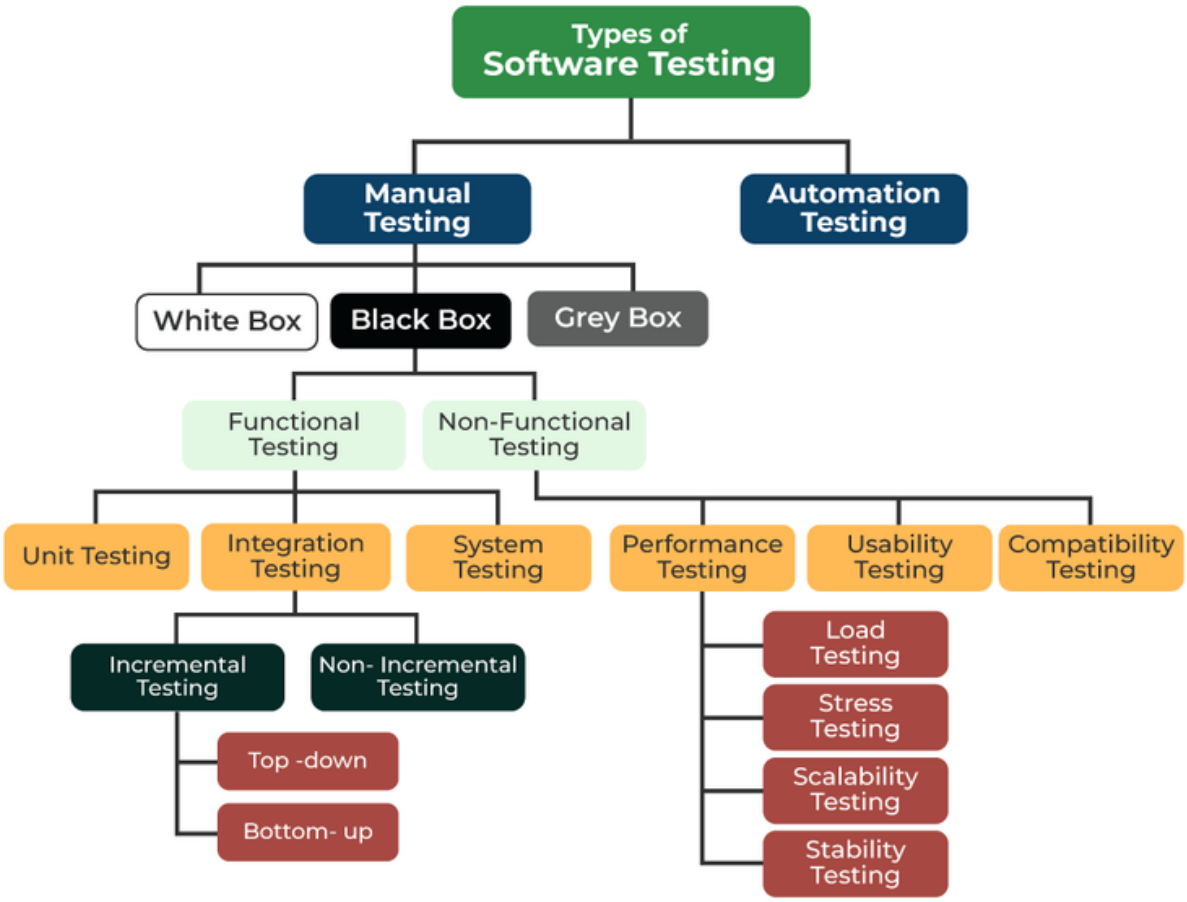
# Third Class Assignment of Software testing

**Type of Software Testing**




**Unit Testing:**

Unit testing is a crucial method in software development where individual units or components of an application are tested in isolation. Usually carried out by developers, these tests are automated and focus on specific sections of the code, such as functions or methods. The advantages of unit testing include early bug identification, prevention of new bugs during code changes, enhanced code modularity, and improved overall software quality and reliability. Notable frameworks like JUnit, NUnit, and xUnit facilitate unit testing.

Unit testing concentrates on the smallest unit of software design. It involves testing individual units or groups of interrelated units, often performed by programmers using sample inputs and observing corresponding outputs. Examples include checking loops, methods, or functions for proper functionality, confirming correct arithmetic precedence, and validating appropriate initialization.

### Integration Testing:

Integration testing assesses how different units or components of a software application interact. Conducted after unit testing and before functional testing, it identifies and resolves issues that may arise during the combination of different software units. Various approaches, including top-down, bottom-up, big-bang, and incremental integration testing, ensure a systematic integration process. Integration testing advantages include issue resolution, assurance of units working together as intended, and overall improvement in software reliability and stability.

Integration testing can be either black-box, focusing on validation and ignoring internal mechanisms, or white-box, emphasizing verification and understanding of internal workings.

### Regression Testing:

Regression testing ensures that changes made to the software, such as bug fixes or new features, do not introduce new bugs or break existing functionality. Typically performed after code changes, regression testing involves retesting the entire application or specific affected functionality, re-execution of previously executed test suites, or comparison with previous software versions. Its benefits include preventing the introduction of new bugs, ensuring the software continues to function as intended, and improving overall reliability and stability.

For example, in a school records application, regression testing might involve combining modules related to staff, students, and finance to ensure proper integration after changes.

### Smoke Testing:

Smoke testing is conducted to verify that the software under testing is stable and ready for further testing. Serving as an initial pass, it checks if the software exhibits issues during the initial switch-on. An example is ensuring that module 1 of a project with two modules works correctly before proceeding to additional testing.

### Alpha Testing and Beta Testing:

Alpha testing, a type of validation testing, is performed internally within an organization before the product release to customers. Beta testing involves end-users testing a version of the software at customer sites. Both types are crucial for identifying issues in a real-time environment, with alpha testing serving as internal validation and beta testing involving a limited number of external users.

### System Testing:

System testing evaluates the entire system based on system or functional requirements, covering both functional and nonfunctional aspects. It includes security testing, recovery testing, stress testing, and performance testing, focusing on required input and output.

### Stress Testing and Performance Testing:

Stress testing involves subjecting the system to unfavorable conditions to assess its performance under duress. Performance testing, also known as load testing, evaluates the run-time performance of the software within an integrated system. Examples include executing test cases requiring maximum memory or resources and checking processor cycle performance.

### Object-Oriented Testing:

Object-oriented testing combines various techniques to verify and validate object-oriented software. This includes testing of requirements, design and analysis, code testing, integration testing, system testing, and user testing.

### Acceptance Testing:

Acceptance testing, conducted by customers, ensures that the delivered products perform as desired, aligning with specified requirements. Object-oriented testing is employed for test plans and project execution during acceptance testing.

**10 automation testing tools with uses:**


**Selenium:**

Use: Selenium is a widely used open-source framework for automating web applications. It supports cross-browser and cross-platform testing, allowing testers to write scripts in various programming languages such as Java, C#, and Python. Selenium WebDriver provides a convenient API for interacting with web elements and automating user actions like clicks, form submissions, and data input.

**Appium**:

Use: Appium is an open-source tool designed for automating mobile applications on Android, iOS, and Windows platforms. It supports native, hybrid, and mobile web applications, making it versatile for mobile app testing. Appium allows testers to write scripts using popular programming languages like Java, C#, Python, and more.

**Cucumber**:

Use: Cucumber is a behavior-driven development (BDD) testing framework that promotes collaboration between developers, testers, and non-technical stakeholders. It uses a Gherkin language syntax for writing human-readable scenarios, making it easy to understand. Cucumber allows the creation of executable specifications, aiding in the validation of application behavior against user-defined scenarios.

**JUnit**:

Use: JUnit is a widely used testing framework for Java applications. It is primarily employed for unit testing, ensuring that individual units or components of Java code behave as expected. Developers use annotations to define test methods and assertions to verify expected outcomes, making it an essential tool for test-driven development (TDD) in Java.

**TestNG**:

Use: TestNG is a testing framework for Java that extends and enhances JUnit functionalities. It supports a variety of test configurations, such as parallel execution, parameterization, and grouping. TestNG is commonly used for unit testing as well as higher-level testing, offering features like data-driven testing and test parallelization.

**Robot Framework:**

Use: Robot Framework is a versatile, open-source, keyword-driven testing framework. It supports acceptance testing, acceptance test-driven development (ATDD), and robotic process automation (RPA). With its simple syntax and easy-to-read keyword format, Robot Framework allows testers to create test cases for various platforms, including web, mobile, and desktop applications.

**Postman**:

Use: Postman is a popular API testing tool that simplifies the process of testing RESTful and SOAP web services. Testers can create and execute HTTP requests, analyze responses, and validate API behavior. Postman also provides collaboration features, allowing teams to share and manage API-related tasks efficiently.

**JMeter:**

Use: Apache JMeter is an open-source tool for load testing and performance testing. It is commonly used to simulate multiple users accessing a web application concurrently, helping to identify performance bottlenecks and assess the system's scalability under different loads. JMeter supports various protocols, making it suitable for testing different types of applications.

**SoapUI**:

Use: SoapUI is a comprehensive API testing tool that supports both SOAP and RESTful web services. It allows testers to create and execute functional, security, and performance tests for APIs. With features like automated test generation and data-driven testing, SoapUI streamlines the testing of web services across different protocols.

**Katalon Studio:**

Use: Katalon Studio is an all-in-one automation testing solution that supports web, mobile, API, and desktop application testing. It is built on top of Selenium and Appium, making it versatile for testing various application types. Katalon Studio provides a user-friendly interface for scripless automation as well as support for scripting using Groovy and Java. It is suitable for both beginners and experienced testers seeking a comprehensive testing tool