**CHAPTER: -05**

**System Testing**

**INTRODUCTION**

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include, but are not limited to the process of executing a program or application with the intent of finding software bugs (errors or other defects). Software testing can be stated as the process of validating and verifying that a computer program/application/product:

* Meets the requirements that guided its design and development,
* Works as expected,
* Can be implemented with the same characteristics, and satisfies the needs of stakeholders.

Testing can never completely identify all the defects within software. Instead, it furnishes a criticism or comparison that compares the state and behavior of the product against oracles—principles or mechanisms by which someone might recognize a problem. These oracles may include (but are not limited to) specifications, contracts, comparable products, past versions of the same product, inferences about intended or expected purpose, user or customer expectations, relevant standards, applicable laws, or other criteria. A very fundamental problem with software testing is that testing under all combinations of inputs and preconditions (initial state) is not feasible, even with a simple product. This means that the number of defects in a software product can be very large and defects that occur infrequently are difficult to find in testing. More significantly, non-functional dimensions of quality (how it is supposed to be versus what it is supposed to do)— usability, scalability, performance, compatibility, reliability—can be highly subjective; something that constitutes sufficient value to one person may be intolerable to another.

**5.1 FUNCTIONAL TESTING**

In software development, functional testing is a type of software testing that validates the software system against the functional requirements/specifications. The purpose of functional testing is to ensure that the system performs and behaves as expected under given conditions. It primarily focuses on testing the functionality of the software application by providing appropriate input and verifying the output against the expected results.

**Characteristics of Functional Testing**

It verifies that the application performs as intended based on the defined functional requirements. It does not concern itself with the underlying code structure.

* It ensures that each function of the software application works in accordance with the requirement specification.
* It is typically conducted manually or using automated testing tools.

**Functional Testing Process**

1. Requirement Analysis: Identify and understand the functional requirements of the system.

2. Test Case Design: Develop test cases that outline input data, execution steps, and expected outcomes.

3. Test Execution: Run the designed test cases and document the results.

4. Defect Reporting: Log and track any defects or inconsistencies found during testing.

5. Retesting and Regression Testing: Verify fixed defects and ensure new changes have not introduced additional issues.

**Types of Functional Testing**

Functional testing includes various subtypes, such as:

* Smoke Testing: A preliminary test to check the basic functionality of the software.
* Sanity Testing: A quick, focused test to validate specific functionalities after minor changes.
* Regression Testing: Ensuring new code changes do not negatively impact existing functionalities.
* User Acceptance Testing (UAT): Validating the software from an end-user perspective before deployment.
* Integration Testing: Testing the interfaces and interactions between integrated components or systems.

**Benefits of Functional Testing**

• Ensures that the software meets business and user requirements.

• Identifies defects early in the development cycle.

• Improves software quality and reliability.

• Provides confidence in the product before its release.

Functional testing is an essential part of the software testing lifecycle, ensuring that an application meets its intended functional requirements. By systematically verifying the expected behavior of software, functional testing contributes to building robust, error-free, and high quality applications that align with business needs.

**5.2 User Interface Testing**

Not applicable

**5.3 Navigation Testing**

Not applicable

**5.4. Form testing:-**

Not applicable

**5.5. Database testing:-**

Not applicable