**Testing**

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.(client)

**Testing Methods**

* **The box approach:**

Software testing methods are traditionally divided into white- and black-box testing. These two approaches are used to describe the point of view that a test engineer takes when designing test cases.

#### White-Box testing:

**White-box testing** (also known as **clear box testing**, **glass box testing**, **transparent box testing** and **structural testing**) tests internal structures or workings of a program. In white-box testing an internal perspective(view) of the system, as well as programming skills, are used to design test cases.

While white-box testing can be applied at the [unit](http://en.wikipedia.org/wiki/Unit_testing), [integration](http://en.wikipedia.org/wiki/Integration_testing) and [system](http://en.wikipedia.org/wiki/System_testing) levels of the software testing process, it is usually done at the unit level.

#### Black-box testing:

**Black-box testing** treats the software as a "black box", examining functionality without any knowledge of internal implementation. The tester is only aware of what the software is supposed to do, not how it does it.

#### Grey-box testing:

**Grey-box testing** involves having knowledge of internal data structures and algorithms for purposes of designing tests, while executing those tests at the user, or black-box level. The tester is not required to have full access to the software's source code.

**Testing Levels**

### Unit testing:

### Unit testing, also known as component testing, it verify the functionality of a specific section of code, usually at the function level. In an object-oriented environment, this is usually at the class level, and the minimal unit tests include the constructors and destructors.

### Integration testing:

### Integration testing works to expose defects in the interfaces and interaction between integrated components (modules).

### System testing:

### System testing, or end-to-end testing, tests a completely integrated system to verify that it meets its requirements.

### Acceptance testing:

### At last the system is delivered to the user for Acceptance testing.