

ISO 9126 Software Quality

The ISO 9126 software is an international standard software quality model that helps in creating a solid framework for assessing software. This standard way of assessing software can be segregated in four different ways. These are used to address subjects of different nature. This software is profoundly used in a widespread way to embrace various models and metrics. The recommended features describe externally when software is found to be a result of attributes of internal attributes of software. The following ways by which a standard software quality model can be calculated are as follows:

1. Quality Model.
2. External Metrics.
3. Internal Metrics.
4. Quality in use Metrics.

The part one of this software quality model is an extension to the previously work done by the other quality enhancing models. The other defined set of software quality models are as follows:

- Mc Call 1977.
- Boehm_1978.
- FURPS

This model is used to represent the cutting-edge research. It moves into figuring software characteristics for the few main purpose of checking software quality control, software quality assurance and software performance improvement. The model can be acquired by purchasing the model from official ISO 926 documentation. With the SQA.NET, this model can be only obtained by the basic structure along with structures, commentary, or guidance. Like every software, ISO 926 software model has distinct qualities. These are laid on following basis:

Functionality: It is a key aspect of any product or service. It is due to this the software is able to fulfill a task and keep to its purpose. It is defined as a software product that helps to meet the needs

of the clients. A functionality of software is dependent on its complexity. For example: an ATM machine. This is further divided in other categories are as follows:

- Suitability.
- Accuracy.
- Interoperability.
- Security.
- Functional compliance.

Reliability: This characteristic determines the capability of software to sustain its use when put under different circumstances.

Usability: The usability of software is highly dependent on the functional uses of software. For example: ATM machine is used to withdraw cash. According to the usability of an ATM; the ATM is not affected or influenced by any amounts entered by the user. This is further divided into other sub-categories and these are as follows:

- Maturity.
- Fault Tolerance.
- Recoverability.
- Reliability Compliance.

Efficiency: This feature of the model is more concerned by resources of the system when used for providing a desired functionality. This type of feature is defined by amount of disk space, memory and network. This is further divided into other sub-categories and these are as follows:

- Understandability.
- Learner ability.
- Operability.
- Attractiveness.
- Usability Compliance.

Maintainability: This property of maintainability of the software model is used to recognize and fix a defect accordingly. The model is inspected for the faults and these can be identified easily. In accordance to this the cause and effect of maintainability of software is a concern. This is further divided into other sub-categories and these are as follows:

- Analyzability.
- Resource Utilization.
- Stability.
- Testability.
- Changeability.

Portability: According to this feature, capable software should easily adapt to the environmental changes frequently as possible. The designing of an object and the practices of its implementation are highly dependent on this feature. This standard method is further divided in few categories:

- Adaptability.
- Install ability.
- Co-existence.
- Replaceability.
- Portability compliance.