**Different Quality Aspects According to the Researchers**

**Five Models Based:** McCall, Boehm, FURPS, Dromey, and ISO

**McCall Model:** McCall’s model was developed by the US air-force electronic system decision (ESD), the Rome air development centre (RADC), and general electric, to improve the quality of software products. This model was developed to assess the relationships between external factors and product quality criteria. Therefore, the quality characteristics were classified in three major types, **11 factors** which describe the external view of the software (user view), 23 quality criteria which describe the internal view of the software (developer view), and metrics which defined and used to provide a scale and method for measurement**. The number of the factors was reduced to eleven in order to simplify it. These factors are Correctness, Reliability, Efficiency, Integrity, Usability, Maintainability, Testability, Flexibility, Portability, Reusability, and Interoperability.** The major contribution of this model is the relationship between the quality characteristics and metrics. However, the model not consider directly on the functionality of software products.

**Boehm Model:** Boehm added new factors to McCall’s model and **emphasis on the maintainability of software product**. The aim of this model is to address the contemporary shortcomings of models that automatically and quantitatively evaluate the quality of software. Therefore, Boehm model represents the characteristics of the software product hierarchically in order to get contribute in the total quality. Furthermore, the software product evaluation considered with respect to the utility of the program. However, this model contains only a diagram without any suggestion about measuring the quality characteristics.

**FURPS Model:** FURPS model was proposed by Robert Grady and Hewlett-Packard Co. The characteristics were classified into two categories according to the **user’s requirements, functional and non-functional requirements**. Functional requirements (F): Defined by input and expected output. Non-functional requirements (URPS): Usability, reliability, performance, supportability. And then, the model was extended by IBM Rational Software – into FURPS+. Therefore, this model considered only the user’s requirements and disregards the developer consideration. However, the model fails to take into account the software some of the product characteristics, such as portability and maintainability.

**Dromey Model:** Dromey (1995) states that the evaluation is different for each product, hence a dynamic idea for process modelling is required. Therefore, the main idea of the proposed model was to obtain a **model broad enough to work for different systems.** The model seeks **to increase understanding of the relationship between the attributes (characteristics) and the sub-attributes (sub-characteristics) of quality**. This model defined two layers, high-level attributes and subordinate attributes. Therefore, this model suffers from lack of criteria for measurement of software quality.

**ISO IEC 9126 Model:**

Since, the number of the software quality models were proposed, the confusion  
happened and new standard model was required. Therefore, ISO/IEC JTC1 began to develop the required consensus and encourage standardization world-wide. The ISO 9126 is part of the ISO 9000 standard, which is the most important standard for  
quality assurance. First considerations originated in 1978, and in 1985 the  
development of ISO/IEC 9126 was started.  
In this model, the totality of software product quality attributes were classified in a  
hierarchical tree structure of characteristics and sub characteristics**. The highest level  
of this structure consists of the quality characteristics and the lowest level consists of  
the software quality criteria**. The model specified six characteristics including  
**Functionality, Reliability, Usability, Efficiency, Maintainability and Portability;**  
which are further divided into 21 sub characteristics. These sub characteristics are  
manifested externally when the software is used as part of a computer system, and are  
the result of internal software attributes. The defined characteristics are applicable to  
every kind of software, including computer programs and data contained in firmware  
and provide consistent terminology for software product quality. They also provide a  
framework for making trade-offs between software product capabilities.

**The Comparison Method**In order to show the clear differences between software quality models, mathematical comparison method is proposed. The method aims to show the clear and accurate differences between quality models, which consider the sub factors in addition to the factors.

It consists of four main tasks: model selection, assigning values, factors comparison, and models comparison.

**Models Selection:** The process of models selection is depends on the scope  
intended to be evaluated, usually the well-known software quality models are  
considered in developing a new model.  
  
**Factors Selection:** The factors of the selected models are collected and combined  
in one structural tree (Fa, Fb…Fn)  
Different sub-factors are considered in each factor from different model, the sub  
factors are combined under their factors (S1, S2, Sn). According to the aim and  
definitions of each factor and sub factor, the repeated are excluded.

**Factors Weighting:** After analyze the scope that needed to be evaluated, the  
experts in this field are required to assign the weight of these factors (W1,  
W2…….Wn) and sub factor (Wa, Wb……Wm) are assigned.

**Factors Values:** the value of each factor in the original models is calculated, based  
on the weight that assigned in the previous step.

o **First,** the value of the same factor within the selected models is calculated  
(Formula 1).  
o **Second,** the total value of each model is calculated (Formula 2), based on  
the calculated values of their factors.

**The Comparison**: the total value for each factor is compared between the selected  
models. That shows the comprehensiveness differences between these factors in  
different models.