**Software Quality**

Software quality is undeniably has become an increasingly essential topic in software engineering. Software requirements have traditionally been characterized as either functional or non - functional, with quality considerations buried in the latter. As the industry shifts its attention from functionality to increasing quality, a new set of quality-related standards is emerging. It is necessary to define quality to specify these new quality standards. A quality model provides the basis for defining quality. Software developers have long understood the importance of properly defining and specifying something before being used in a product. Consequently, the industry's current push for software quality lacks a strong background in a kind of an agreed-upon quality paradigm that can be used to evaluate and design software quality.

Jim McCall presented the first quality model in 1977, also known as the General Electronic Model. There was no actual model to assess or check the product's quality before the presentation of this model. This model gives a collection of fundamental characteristics for determining product quality (McCall et al., 1977). Quality is primarily measured in McCall's quality model based on the opinion of the individual(s) or users by responding to questions presented in a questionnaire. The following are three key perspectives of the McCall approach for proclaiming and acknowledging product quality (McCall et al., 1977).

1. Product Revision: this is concerned with the product's debugging, testability, flexibility, adaptability, and maintainability.
2. Product Transition: this is concerned with the product's environmental adaptation, portability, interoperability, and reusability.
3. Product operation: this refers to the degree to which a product can be comprehended quickly and readily and encompasses factors such as correctness, reliability, efficiency, usability, and integrity.

In comparison, the ISO 9126 Model stresses external traits, while the McCall Model also addresses internal qualities (T.E.A.M.Q.U.E.S, 2020). Reusability, for example, is a product's inherent attribute. Customers are ignorant of the efforts made by product developers to design reusable components. Second, high-quality factor-like testability is a poor sub-feature of maintainability in the ISO 9126 Model. Most of the quality elements in the McCall Model, like interoperability and reusability, are crucial to developers. The ISO 9126 Model, on the other hand, primarily assesses the product (T.E.A.M.Q.U.E.S, 2020).

Another difference is that, in McCall's Model, a single quality criterion might affect numerous quality factors, while in the ISO 9126 model, each sub-feature affects just one quality criterion. McCall proposes 11 high-level quality parameters; however, the ISO 9126 standard only specifies six quality characteristics. Compared to McCall's Quality Paradigm, interoperability is not an inherent, top-level quality criterion in the ISO 9126 model.

**References**

McCall, J. A., Richards, P. K., & Walters, G. F. (1977). *Factors in software quality. volume i. concepts and definitions of software quality*. GENERAL ELECTRIC CO SUNNYVALE CA.

T.E.A.M.Q.U.E.S. (2020). Compare McCall's quality model with ISO 9126 quality model. *Ques10*.