The Quality Assurance

The Quality Assurance is part of the executing process and is concerned with making sure that the quality objectives are met. It is focused on process improvement.

The Quality Assurance Phases:

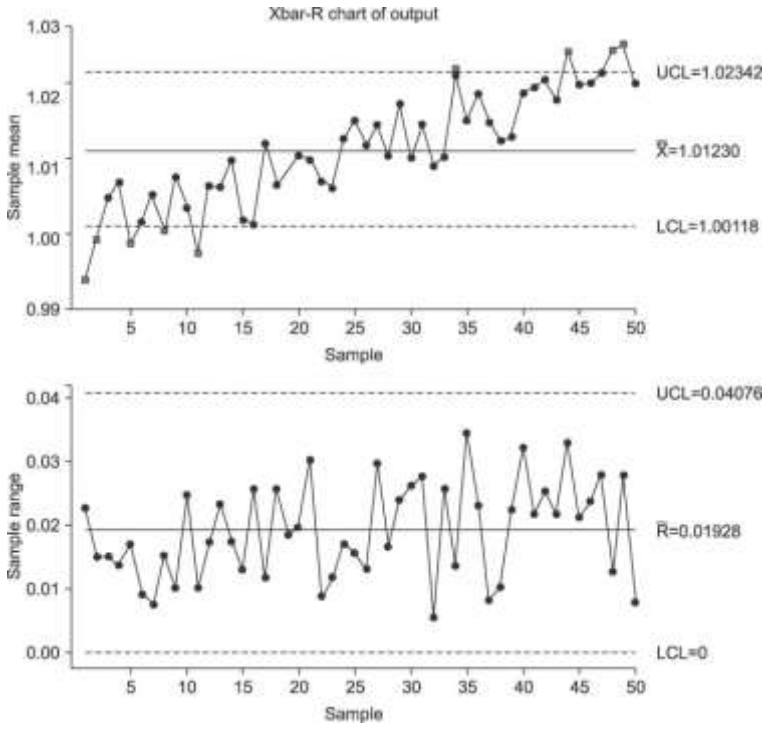
- Failure Testing:

  The failure test or the endurance test for a complete consumer product. Where the product is run until it fails, it is often done by entering many images and making sure that the program has identified them correctly and then makes a simplified explanation for the visitor to understand

everything about this image. This may reveal many unexpected weaknesses in the product, and the data is used to drive improvements in the programming process. Very simple changes often greatly improve product service.

- Statistical Control:

Statistical control is based on analyzes of objective and subjective data. Many organizations use statistical process control as a tool in any quality improvement effort to track quality data. Product quality data is plotted statistically to distinguish common cause variance or special cause variance.



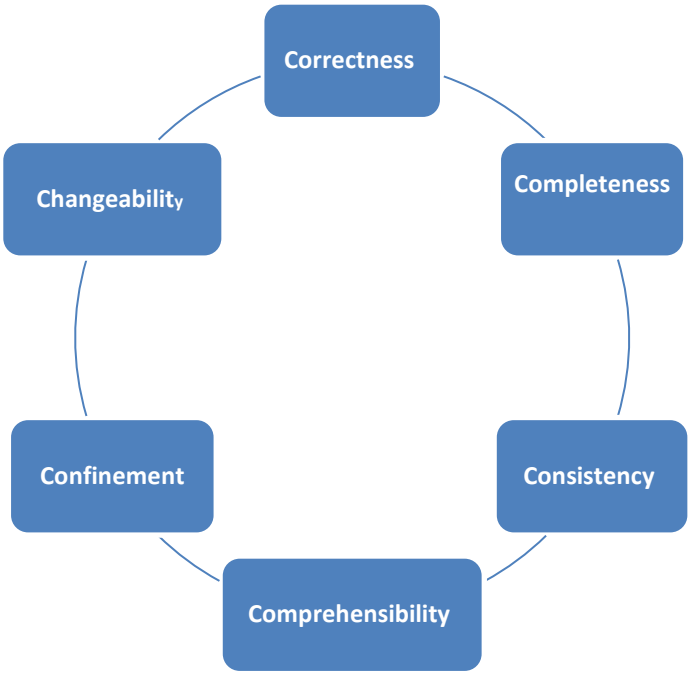
Data (images) can be taken from areas examined from a sample, a batch of a portion, and then statistical differences are analyzed and plotted. Control can then be carried out on the part in the form of rework or scrap, or control can be performed on the process that made the part, ideally eliminating the defect before more parts are made like it.

- Total Quality Management:

  The quality of the products depends on the quality of the ingredients used and in the application we use advanced technologies in AI and NLP. The process(s) are managed with quality assurance related to TQM.

The 6C model quality goals presented by Mohagheghi et al:

1. Correctness: A model is correct if it includes the right elements and correct relations between them, and if it includes correct statements about the domain. Furthermore, a model must not violate rules and conventions. This definition includes syntactic correctness relative to the modeling language as well as semantic correctness related to the understanding of the domain.



2. Completeness: A model is complete if it contains all relevant information, and if it is detailed enough according to the purpose of modeling. For example, requirement models are said to be complete when they specify all the black-box behavior of the modeled entity, and when they do not include anything that is not in the real world.

3. Consistency: A model is consistent if does not contain contradictions. This definition covers horizontal consistency concerning models/diagrams on the same level of abstraction, vertical consistency concerning modeled aspects on different levels of abstraction as well as semantic consistency concerning the meaning of the same element in different models or diagrams.
4. Comprehensibility: A model is comprehensible if it is understandable by the intended users, being humans or tools. In most of the literature, the focus is on comprehensibility by humans including aspects like aesthetics of a diagram, model simplicity or complexity, and the use of the correct type of diagram for the intended audience. Several authors also call this goal pragmatic quality.
5. Confinement: A model is confined if it suits to the modeling purpose and the type of system. This definition also includes relevant diagrams on the right abstraction level. Furthermore, a confined model does not have unnecessary information and is not more complex or detailed than necessary. Developing the right model for a system or purpose of a given kind also depends on selecting an adequate modeling language. This means that the modeler uses language concepts that are suitable for the intended purpose of the modeling activity. Further concepts should be used very sparsely or even omitted deliberately.
6. Changeability: A model is changeable if it can be evolved rapidly and continuously. This is important since both the domain and its understanding as well as system requirements evolve over time. Furthermore, changeability should be supported by modeling languages and modeling tools as well.