The software quality assurance

# Rémi MIKEL 13129095

The software quality assurance (SQA) is a set of planned and systematic activities of all actions necessary to provide adequate confidence that a software product or changed meets the requirements and expectations established. SQA practices vary depending on the business model and industry or software is employed.

Basic knowledge of SQA is divided into three main categories:

1. **The foundations of software quality**:

* The importance of culture and ethics in software engineering: It is expected at this latter stage that software specialists make a commitment in regard to software quality as a key element their daily activities. Ethics must play a significant role. Indeed, a culture and attitudes that promote ethics irremediably affect software quality.
* The value of the cost and quality: According to the business area and the specific quality issues, there are all kinds of quality characteristics which are expected by customers. In some cases, the quality characteristics may be required in a contract, or may be required to a greater or lesser degree. Of course we know that quality comes at a cost. To help managers better understand the cost of software quality , it is possible to calculate it by taking the sum of the costs in the following four perspectives: 1 ) the cost of defect prevention , 2 ) the cost of the quality assessment , 3 ) the cost of internal software failure , 4 ) the cost of external failure of software. The software quality assurance specialist may therefore properly present these issues to the organization in question and help to understand the return on invested capital.
* Improving quality: The quality of software products can be improved by an iterative process of continuous improvement that requires the management, control, coordination and feedback loops: 1) the process of the life cycle of the software, 2) the process of detection, correction and prevention of errors / defects, 3) the process of quality improvement. Theory and concepts of quality improvement, such as implementation of quality by preventing and detecting errors, continuous improvement and a focus on customer needs, are all applicable to the field of software.

1. **The process of managing software quality** : Management of software quality is defined in ISO 12207 [ISO 08a ] standard. This standard specifies the scope of the quality management software and describes the following processes:

* software quality assurance ( SQA );
* verification and validation (V & V);
* review and audit ;
* problem solving

These processes result in the crop quality and also enable the establishment of prevention of defects. The software quality assurance process offering a better quality software in the implementation of development projects , maintenance or proposed change in the IT infrastructure. They also provide, as a by- product of the synthesized information for decision-making, including an indication of the quality of the software process of the organism. The SQA process also can give an indication of the actual state of progress (eg management plans, development, configuration management, impact analysis). They aim to assess whether the intermediate and final products meet their initial requirements specified. The results of SQA activities can be summarized in dashboards (technique used to present key performance measures of a software project) for decision making and identifying corrective actions. Manager SQA must ensure that the results of these qualities are verified and relevant reports.

1. **The practical considerations**: The software engineer must also have practical knowledge regarding the use of quality principles and techniques common in the field. This section assesses the knowledge necessary to ascertain whether the quality target can be achieved.

The requirements of the application of quality: Factors influencing the planning , management and the choice of activities and SQA techniques when starting a project development, maintenance and operation, including: - field system in which the software will operate (safety - critical , safety - critical , Mission - critical , business - critical , critical - environment ) ; - System requirements and software ; - Components (internal ), commercial ( external) or standard to be used in the system ; - Specific engineering standards applicable to internal and imposed by the relevant field software ; - Software tools and methods to be used for the development and maintenance as well as evaluation and quality improvement ; - The budget , personnel, project organization , plans and establishing the schedule of all processes ; - Users and the intended use of the system; - The level of system integrity ; - The size of the organization; - The geographical distribution of development teams ; - Cultural diversity.