A quality assurance model for an information system development life cycle

Mohamed H. Abdallah
Kuwait University, Kuwait

Introduction

The objectives of systems analysis and design have been identified in different terms according to the type and importance of the information system. The most important objective is to design a system which satisfies the user requirements and performs the required tasks. In addition, the system must possess some features such as efficiency, accuracy, compatibility, flexibility, portability and acceptability[1].

To achieve the objectives of systems analysis and design, the system life cycle stages have to be well identified. The different methodologies for system development have to be carefully selected according to the type of system and the level of quality to be achieved. Different types of system documentation have to be prepared, classified and imposed by the system manager.

The importance of documentation throughout the system life cycle stages cannot be underestimated. The design stage cannot be started without complete documentation of the analysis phase. Documentation of the system programmes is essential for programme testing and maintenance, and cannot be started without documentation of the system design phase.

The lack of documentation noticed in some organizations could be due to[2]:

- lack of knowledge and experience of documentation techniques for people working in systems analysis, design and programming;
- some old systems and packages were implemented without documentation and are still used with no desire to modify;
- the rush to obtain system results by modifying programmes without the modification of documents;
- the lack of management instructions about the importance and necessity of documentation.

The quality of information systems is becoming the concern of both users and systems analysts. The concept of total quality assurance developed and widely
used in manufacturing environments is now being introduced in the software processing[2].

Different techniques have been used for quality improvement in the manufacturing environment and can be applied in software processing. The techniques depend on building a model for the relationships between influential input variables and output quality characteristics[3]. When the nature of the relationship between the input variables and the process output model is examined, then a control function can be imposed to monitor the process output, and detect when changes in the inputs are required to bring the process back to an in control status. A capability maturity model was developed by the Software Engineering Institute in Pittsburgh which identified five levels of maturity for the software process[4]:

- Level 1: The stages of the process are not defined, and process control cannot be applied.
- Level 2: The software process can be viewed as a succession of black boxes and control is exerted only on the final software products.
- Level 3: The internal structure of the process is defined; and control can be exerted between the different processes.
- Level 4: The defined software processes are instrumented and controlled quantitatively, problems and progress can be measured.
- Level 5: The different processes of the software are continuously improved by quantitative feedback from the process.

The final level is considered to be the highest maturity level in which the process is under statistical quality control, where optimization of quality level can be achieved. The present work proposes a quality assurance model for information system development. It consists of different stages which work as a cycle, associated with the system development life cycle. A field investigation was conducted to assess the existence of the quality assurance function in computer-based information systems in governmental and private organizations in Kuwait.

The quality assurance life cycle model

The model is built in the light of total quality concepts, to satisfy the objectives of the system users, and the software products.

The model consists of the following stages (see also Figure 1):

- Set quality objectives.
- Design system development processes.
- Select methodologies and associated documentation techniques.
- Define activities and controls, estimate time and resources.
- Build a project management information system.
- Process implementation and record actual performance.
Set quality objectives

- Compare with standards and take action.
- Review and update project management information system.

Quality, in general, is a multifaceted entity having different dimensions such as performance, reliability, durability, serviceability, aesthetics, perceived quality and conformance to standards[3]. The characteristics of information systems imposed additional requirements such as timeliness, security, accuracy, implementability, maintainability, flexibility, etc.

A review of the literature regarding the quality of information systems reveals a lack of quality assurance functions during system development stages and also a lack of documentation standards supporting quality control practices[5,6].

The quality assurance department has the responsibility of defining and ensuring the type and level of quality required to be achieved, mostly with the risk of decreasing the level of other quality aspects and within the budget available. The functions of the quality assurance department can be summarized as follows:

- Define and ensure a specified level of quality for operations processes, personnel resources and final products.
- Prepare reliable tools and quality measurements.
• Apply different methods of evaluation during the system life cycle stages such as: strategic planning and feasibility studies; computer selections; system analysis; system design; system programming; system testing; system implementation; system monitoring and maintenance.

• Establish, measure and evaluate a number of performance indices for hardware (CPU utility, input/output utility, hard disk utility, magnetic tape utility, overall system utility) and software utilities (software utility, database utility, application packages utility).

• Establish, measure and evaluate indices for system productivity, responsiveness and reliability[7]. System productivity includes the number of programmes executed per unit of time, throughput rate transactions per second, volume of data processed per second, maximum system capacity. System responsiveness includes turnaround time, response time, reaction time. System reliability and availability includes breakdown rate and number of breakdowns per unit of time.

• Define the corrective actions to be taken when quality levels deviate from the standards.

Design system development processes
Although there is some sort of standardization with respect to system development processes, the type and characteristics of the information system could require different types of process.

The standard processes reported in the literature for system development are investigation, analysis, design, programming, testing, implementation, maintenance and evaluation[2,8-11].

Select methodologies and associated documentation techniques
A range of system analysis and design methodologies and related documentation techniques are available which satisfy different systems analysis and design requirements[12-18]. The software engineer, when designing the different processes for system development, has to study carefully the methodologies and select between them in the light of the following factors:

• The familiarity of systems analysts with the methodology and documentation techniques; however, managers have to encourage the learning of new methods and tools.

• The type of hardware and software available.

• The type and characteristics of the information system (input and output requirements).

• The type and level of quality.

• The practicability, reliability, and efficiency of the tools.
Organizational factors, such as information department experience, personnel experience and the rank and location of the department within the organization structure.

Automated approaches for system development such as CASE tools are also recommended to improve the productivity of the process and the quality of the software product[19]. CASE tools can be used primarily for documentation and data dictionaries[20]. Selection of suitable documentation for the system development life cycle can contribute to the success of system maintenance and increase user satisfaction[21,22].

Table I includes a summary of the methodologies, documentation and control tools reported in the literature, classified according to the system life cycle stages[1,4,8,23-26].

Define activities and control, estimate time and resources
The development processes are broken down into a number of defined activities. The time and resources required for each activity are estimated. The logical procedures of activities is represented as a project network to be analysed to estimate the completion time of the project and conduct related calculations of early start and finish of each activity.

The control activities are built within the project activities and also require time, resources and documentation. The objective of control within the quality assurance model is to provide accurate and timely information, needed by the quality assurance department to identify defects in the process before they become serious. Control is also needed to provide corrective actions to maintain the required level of quality. Successful control requires good planned and documented activities, and an accurate and timely communication system for reporting and taking action.

Apart from these types of control activities, project tracking is a continuous effort to ensure the required level of quality. Periodic meetings with team leaders of technical groups are essential to review technical specifications, to ensure the required level of quality at the start and the end of project stages. Meetings are also required in the case of unsolved serious problems and the need to take hard decisions, concerning either stopping or continuing the project activities. The system audit is the final review of the project and is conducted by an external team, and it has to be planned, implemented and documented for legal purposes. The need for documentation standards for control functions is identified in several studies[5,27].

Figure 2 illustrates a network for an inventory and sales system, and Figures 3 and 4 are examples of activity sheets.

Building a project management information system
The project management information system is considered to be complete documentation about how, when and who will conduct the system development
### Table 1. Summary of methodologies and documentation for system life cycle

<table>
<thead>
<tr>
<th>Stages</th>
<th>Problem definition of system objectives</th>
<th>System investigation</th>
<th>System analysis</th>
<th>System design</th>
<th>Programming</th>
<th>System testing</th>
<th>Implementation and evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feasibility study</td>
<td>Input/output analysis</td>
<td>Network analysis</td>
<td>Data structure design</td>
<td>Structural programming</td>
<td>Structured testing</td>
<td>Implementation strategy and approaches</td>
<td></td>
</tr>
<tr>
<td>Cost-benefit analysis</td>
<td>Statistical hypothesis</td>
<td>Questionnaire</td>
<td>Data analysis and modelling</td>
<td>Fourth generation languages</td>
<td>Link test</td>
<td>Hardware monitoring</td>
<td></td>
</tr>
<tr>
<td>Long-range plan report</td>
<td>Management objectives</td>
<td>Reports questionnaire</td>
<td>Program chart</td>
<td>Program specification</td>
<td>Testing procedures</td>
<td>Software monitoring</td>
<td></td>
</tr>
<tr>
<td>Management</td>
<td>System objectives</td>
<td>Existing system report</td>
<td>Data dictionary</td>
<td>Data dictionary</td>
<td>Test data</td>
<td>Quality monitoring</td>
<td></td>
</tr>
<tr>
<td>Documentation</td>
<td>Report</td>
<td>Input-output questionnaire</td>
<td>Data process specification</td>
<td>File specification</td>
<td>Test data</td>
<td>Convert system plan</td>
<td></td>
</tr>
<tr>
<td>Control</td>
<td>Review and evaluation of methodology</td>
<td>Review methodology</td>
<td>Review methodological</td>
<td>Structured walk through</td>
<td>Check with standards and initial system objectives</td>
<td>Software audit</td>
<td></td>
</tr>
<tr>
<td>Feasibility study review</td>
<td>Review and evaluation of methodology</td>
<td>Review methodology</td>
<td>Review methodological</td>
<td>Structured walk through</td>
<td>Check with standards and initial system objectives</td>
<td>Audit trial</td>
<td></td>
</tr>
</tbody>
</table>

### Notes
- **Table 1.** Summary of methodologies and documentation for system life cycle.
A quality assurance model

Figure 2.
A network for inventory and sales system development

activities. It is a powerful management tool for planning, implementing and controlling all aspects of the system development cycle. It contains the following data:

- Project planning data: activities of the project, duration, resources, responsibilities of personnel involved, complete analysis of project time and cost.
- Project monitoring data: updating of project status, activities start and finish time.

**ACTIVITY SHEET**

Activity: Writing a program

Start time: Finish Time:

**Procedure:**

1. Problem definition.
2. Problem analysis: input-output-procedure.
3. Program flowchart.
4. Program pseudo code.
5. Review flowchart and pseudo code.
6. Code program with the specified language.
7. Review program.
8. Compile program and correct errors.
9. Run program using test data.
10. Modify program according to the test results.
11. Run program using actual data.
Activity: Prepare a decision table

Start time: \hspace{1cm} Finish Time:

Procedure:
1. Problem definition.
2. Identify the number and status of the conditions involved.
3. Calculate the number of rules.
4. Identify the actions required for every rule.
5. Sketch the decision table, illustrate the conditions and related actions.
6. Make necessary simplification, removing identical rules.
7. Check for contradictions.
8. Make the necessary documentation.

- Project quality data: quality data are collected and distributed to facilitate corrective actions at the right time. Report or direct entry to the information system allows problems to be identified and can direct efforts to solve them.
- Budget control data: cost data concerning different activities of the project are controlled revealing deviation of cost if it occurred, and the use of allocated resources.
- Staff management data: personnel data regarding staff qualifications and training in new techniques and methods.

Process implementation and recording actual performance
The processes of system development are implemented according to a plan. Responsibilities related to different activities are allocated according to the complexity of the system and the size of the project. The quality assurance department controls activities using different measurement tools. Actual performance is recorded according to the planned specifications.

Compare with standards, take action
Actual performance is reported continuously throughout implementation and after finishing the activity. A ctual performance is compared with a standard; if there is deviation, action is taken to correct this deviation, if possible. If not, the activity is rejected (see Figure 5).

Review and update the project management information system
Reviewing and updating the project management system is a continuous process. Updating covers the activity's finish time and the start times of the following activities. Resources can also be reallocated according to the process requirements and in response to reports on the implementation of activities. Quality objectives at the process level can be evaluated and readjusted.
Evaluating computer-based information systems in Kuwait

A field investigation was conducted of computer-based information systems in a selected sample of 34 different governmental and private organizations in Kuwait. The objectives of the investigation were to assess the existence of the quality assurance department and its functions; and to evaluate the performance indices used for hardware and software, system productivity, system responsiveness, system reliability and availability.

Results

Of the sample institutions surveyed, 70 per cent were government organizations and 30 per cent were privately owned. As regards experience of working with computers, 9 per cent had used them for less than three years, 6 per cent for less than five years, 18 per cent for less than ten years, and 67 per cent of organizations had used computers for more than ten years. Only in 28.6 per cent of cases was there a quality assurance department or function. (See Tables II-VI for more information on the sample firms.)

Statistical analysis of the data obtained from the field investigation indicates the following:

- In the 28.6 per cent of organizations which did have a quality assurance department, the following duties were carried out: set up standards prior to the development phase, improve design of database files and programs, play an important role in the design and development phases, control the quality level during implementation, improve communication with users, and solve their problems with the system.

- System testing, implementation and maintenance were the most important functions of the quality assurance department.
On average, 21 per cent of the organizations investigated implement performance indices for computer system productivity, responsiveness, reliability and availability.

The overall average of hardware utility is 63.1 per cent and, for software, 55.1 per cent.

Breakdown rate is very low (1.88 per cent for hardware and 5.3 per cent for software); system productivity is high (1,058 transactions per hour, with a low response time of 1.6 seconds. However, indices can be changed from one environment to another, but they can always be used as a guide for enhancing system performance.

<table>
<thead>
<tr>
<th>Type</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Education</td>
<td>4</td>
</tr>
<tr>
<td>Commerce</td>
<td>24</td>
</tr>
<tr>
<td>Health</td>
<td>3</td>
</tr>
<tr>
<td>Finance</td>
<td>21</td>
</tr>
<tr>
<td>Research</td>
<td>3</td>
</tr>
<tr>
<td>Services</td>
<td>36</td>
</tr>
<tr>
<td>Transport</td>
<td>6</td>
</tr>
<tr>
<td>Others</td>
<td>3</td>
</tr>
</tbody>
</table>

**Table II.** Distribution of organizations according to type of activities

<table>
<thead>
<tr>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>High</td>
</tr>
<tr>
<td>Middle</td>
</tr>
<tr>
<td>Operational</td>
</tr>
</tbody>
</table>

**Table III.** Position of computer department in the organization structure

<table>
<thead>
<tr>
<th>Function</th>
<th>% of organizations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preliminary system investigation</td>
<td>37</td>
</tr>
<tr>
<td>(problem definition – objectives)</td>
<td></td>
</tr>
<tr>
<td>Feasibility studies</td>
<td>23</td>
</tr>
<tr>
<td>Computer selection</td>
<td>34</td>
</tr>
<tr>
<td>System analysis</td>
<td>31</td>
</tr>
<tr>
<td>System design</td>
<td>37</td>
</tr>
<tr>
<td>Programming</td>
<td>37</td>
</tr>
<tr>
<td>System testing</td>
<td>51</td>
</tr>
<tr>
<td>System implementation</td>
<td>46</td>
</tr>
<tr>
<td>System maintenance</td>
<td>43</td>
</tr>
</tbody>
</table>

**Table IV.** Distribution of the organizations according to the functions of the quality assurance department
Conclusion
This article outlined a quality assurance model for system development processes. It consists of successive stages, each one of which has to be under control to achieve the quality objectives. Documentation of every stage provides a tool for continuous evaluation between stages and for the final software product.

The model emphasizes the importance of suitable methodologies and documentation techniques for the different stages. The quality assurance department plays an important role in applying the model by establishing the quality standards for processes, activities, documentation and tools used. It also outlines standards for system utilities, productivity, responsiveness and reliability.
The study also revealed that a limited number use the quality assurance concepts, but some are beginning to establish quality assurance departments. System managers are becoming aware of the importance of quality assurance for successful information system development.

References


Further reading