

Particularities of Quality Management in Software Industry

Liviu ILIEȘ

Babeș-Bolyai University, Cluj-Napoca, Romania

E-mail: iliesliviu@yahoo.com

Cătălin AFRĂSINEI-ZEVOIANU

Babeș-Bolyai University, Cluj-Napoca, Romania

Phone: +4 0264/512568

Abstract

Very often IT domain, with its outcomes, through its multidisciplinary orientation, is an essential contributor to quality assurance of economic bodies and not only. It is difficult nowadays to find out an activity sector or even a sub-sector where software applications, regardless their nature, hadn't marked out their place and contribution to its good economic and social development. In order to contribute as a tool toward economic and qualitative increasing of performance, the tool itself (the software product) must be of high quality.

Therefore, it's useful to turn back to a less visible thing, placed behind or aside of the other front stage things. It's about the quality of the quality drivers, the quality of software, the quality into the software industry. The last thing is treated more carefully in this study and the intention is nothing else than a closer look and reveal of those elements which create some quality particularities coming up from those „specialties” of the so much particular software domain.

Of course, quality principles are truly the same but their interpretation and implementation are slightly particular. The authors' opinion is that the more we know about these particularities the easier is quality management improvement in software industry but with the remark it isn't enough and the subject still remains not totally covered.

Keywords: *quality, quality management, IT, software, plus quality*

JEL classification: M15

1. Perception of quality in software industry

Software industry, generally, and the one of Romania, especially, with its particularities, keeps looking the ways to guaranty a sufficient quality level to their outcomes, software applications, so as to generate quality or important quality bound to those who choose to use them.

Nowadays, the software product is seen as a working instrument strictly necessary in almost all economic and social activities (Roncea, 2000). It is well known that software applications have a high level of diversity and complexity (Gillies, 1997), starting from the simplest programs and ending with the most complex databases or reports processors and analysis capable to provide results impossible to be obtained in the past.

Regardless the complexity, we could state that, from the quality point of view, the aim is one and the same: software products, with the vital role they are playing, have to prove a superior qualitative level. Naturally, we ask ourselves which are the elements that could generate quality or quality perception in the case of software products. We provide some examples:

- a total completing of the purpose it was created for;
- easiness of installing or implementation;
- acceptable hardware requirements;
- a reasonable level of required technical knowledge or an easy to understand information to be assimilated;
- facilities regarding integration with other applications;
- an easy to make initial data loading;
- a very low risk level of losing data;
- a high level of data security;
- an easy to use application and minimum attached costs;
- intuitively regarding program operating;
- adaptability to the business particularities;
- easy and costless maintenance;
- portability;
- high quality of attached services: installation, trainings, post installation service, updates.

All of these elements, and specific others, are the quality requirements (Ilies, 2003) in fact. Their cumulative completing becomes a core condition of meeting a high level of quality of the software product. Practically, it is almost impossible this think to become real if the entire activity with its entire processes involved into creation, implementation, maintenance of the product software, whose quality evaluation is made by these elements partially mentioned above, is not carried out in the context of and respecting the principles of quality.

Assurance of the quality has to be not the responsibility of a single person, even if he can have a top position, but a concern of every employee within organization. The concept of quality has to be embedded in the organizational culture and the leaders have to pay continue attention (Năstase, 2009) for building a climate of trust and fostering the innovation in different parts of the organization.

Having an analysis of the above mentioned elements we are able to assert that they have different quality impact. When we state this, we refer the fact that the quality perception of the user is influenced harder by some of the elements and lower by others. We can even say that the last ones, having low perception impact,

could be ignored at the start till the first ones, with visual or other types of impact, are felt as being complied with the quality criteria imposed by the users.

For exemplification purposes, we enumerate some elements with high and respectively low impact on quality perception of the software application users:

1. **with high impact:** here we include those elements with instant visual impact or of other nature:
 - easy to install
 - stability on operating
 - well structured functionalities so as to ease their later finding and identification
 - correctness of results
 - different and suggestive charts based on the generated data
 - highly adaptable software based on a possible initial configuration and parameterization
 - high operating speed followed by immediate results
 - friendly and intuitive user interface
 - permanent technical and informational support at the place of working in the initial implementation phase and in the next stages of the application running (consulting, documentation, training etc)
 - etc
2. **with low impact:** it is mainly about those details which don't essentially affect the main functionalities, operability, results correctness and they could be avoided through the operating mode, parameterization or their reparation could be done with low comparable effort. We give the following examples:
 - efficiency of data filtration
 - interoperability of application's windows
 - the existence of defining/editing possibilities based on wizards
 - information about who, what and when something has happened into the software application
 - etc.

A rigorous approach of the quality management in the case of software product means, from our opinion, identification and categorization of these quality criteria on impact grades regarding clients perception on product's quality (Ilies, 2003). Beside the previous example where the split of quality criterions was made just on two impact grades, in the real life, based on necessity and specifics, it would become justified to structure the quality requirements on several impact levels with the specification that this classification has to be very well defined and delimited in order to avoid confusions.

2. „Plus quality” concept applied on software industry

When we talk about „quality impact”, with different grades, we in fact refer to the way those quality generating elements are influencing users’

satisfaction. Quality is measured through the satisfaction revealed by the product/service consumers and satisfaction is determined by how much the product/service meets their expectations. This idea is the background of what it is called „innovative quality”. Traditionally, quality used to be based on how much the final product or service complied with some specifications and this concept determined the development of „quality inspection” and then of „quality assurance”. Today, what matters the most is not the delivering of a quality product or performing of a quality service. It is, in fact, the delivery of satisfactions. If the client is fully satisfied, surely the products or the service he or she owns is exceeding his/her expectations regarding the quality.

In a more competitive environment as the today’s one, the organizations have to take into consideration not only the present needs of their stakeholders, but they have to be able to predict their evolution, the changes in their preferences, in order to be perceived as a reliable partner and an organization with a strong customer focused organizational culture (Năstase, 2007).

Quality today is no more considered as being a simple look for conformity with a specification. Keep looking just for diminishing the gap between what the client asks and what the client receives it’s not the way to success anymore. What satisfies the client has to be guessed or discovered. It’s mandatory to innovate¹.

Everything starts from the needs and their satisfying. Technical studies made with the purpose of finding out the optimum solutions of reaching the clients satisfaction have revealed the fact that the needs to be satisfied might be split out on three categories based on the owner’s reaction:

- *default needs*: the functions which satisfy this type of needs are considered as a natural right. Satisfying of these needs is mandatory but it doesn’t create satisfaction and, on the opposite situation, their missing generates a high level of dissatisfaction. It’s like trying to ride a bicycle without wheels. In the market economy any organization cannot exist if the finality of its activity doesn’t meet the default needs. Therefore, a telephony service cannot exist if it doesn’t allow communication between sender and receiver. Going deeply with this example, the continuous increasing of subscribers’ number has determined the technologies of the domain to be accordingly improved but this thing is not perceived at the subscriber’s level and consequently it doesn’t create satisfaction despite the price of investments, which is often high;
- *specified needs*: as these needs are fulfilled, they generate a corresponding, proportional satisfaction. It’s about the additional services, attached to the basic product or service: detailed phone invoice, „wake up” service, centralized locking or air conditioning for cars etc. In this case, the additional investment is justified for a proportional increase on satisfaction;

¹ H. Mitonneau, *O nouă orientare în managementul calității*, București, Editura Tehnică, 1998

- potential needs*: are those needs which are not perceived by the customers at the moment, but, if they are fulfilled, they generate an unexpected satisfaction. This type of satisfaction is crucial for the client. This is, in fact, „plus quality” or „innovative quality” which differentiates the products or services on a competitive market. Well known companies are trying hard to create plus quality to their products and services in order to gain competitive advantage. Examples about this can be found in domains like mobile telephony (accessing the Internet directly from the phone, phones with digital photo cameras etc) or IT (USB and FireWire technologies, infrared communication between devices, compact disks with bigger and bigger capacities etc) and, generally, in domains having a high level of development dynamic. When events like those exemplified appeared, nobody has specified. The innovative designers had the intuition of those potential needs. It’s about the ability of satisfying potential needs which generates a high level of satisfaction for a relatively low amount of money invested in the most of the cases. However, it must be stated that the potential needs, once satisfied, will become, by the time, specified needs (second category) and the companies have no other possibility unless to innovate new functions for their products and services so as to keep satisfying the potential needs. We realize that these needs to have an accelerate dynamic which impose a corresponding attitude to the companies in order to access a reasonable position on the market.

By sustaining the ideas already mentioned, the bellow image presents the three categories of needs and their impact on the client if they are satisfied.

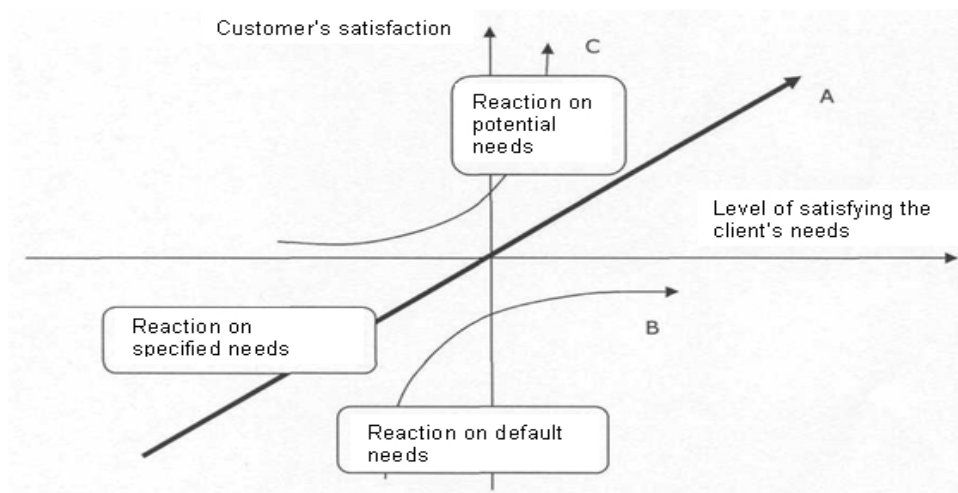


Figure 1 The three categories of needs and their impact on the client’s quality perception

(Adaptation: Mitonneau, H., *O nouă orientare în managementul calității*, București, Editura Tehnică, 1998)

By analyzing the above figure, we can resume with the following conclusions:

- *default needs* (curve B) generates satisfaction only in the initial phases when the client get intouch with the product or service. Later on, satisfaction remains at the same level;
- *specified needs* (curveA) must be fulfilled because the market ask for them. Satisfaction linked to them evoluate proportionaly with the effort paid by the provider in this case;
- *potential needs* (curve C) must be discovered through innovation. These give plus quality to the product or service and they make the difference on quality between the competing companies; it is the critical element for differentiation. Therefore, satisfying the potential needs determines the highest level of satisfaction, the ratio between effect and effort being certainly in the favor of the effect.

Therefore, from the user's point of view, the quality of the product is perceived according to the satisfaction created because of its utilization. The more the producer is capable to find out the most efficient ways to reach the highest level of consumers' satisfaction, the higher its changes of full success will be. All of these things are valid in all domains and especially in software industry where the contact between the software product and the user is a very direct one, the last one being hardly influenced by the quality of the first one. The guarantee of success in software domain, and not only, seems to be easy assured if we are taking into account the following elements:

- first, the quality of the product has to be approached from the user's/consumer's perspective
- a product is qualitative if and only if it produces satisfaction, it impresses the user
- the producer has to be capable to identify the qualitative elements with high impact on the user and to implement them with priority onto the offered product. Very useful for this case is to structure the needs in the three categories mentioned above. Specified and potential needs are critical for subsequent product attractiveness.
- searching for potential needs has to be a continuous one because of the fact that a potential need is rapidly changing to a specified need which has a lower emotional impact.

3. Quality from producer's perspective vs. from consumer's perspective

Even if the quality approach has to be taken from user's/consumer's perspective, quality from *producer's perspective* cannot be neglected. In most of the cases quality from user's perspective is conditioned by the quality from producer's perspective. For instance, the user is way too little interested by the technology used for the software application design, or by the programming

language involved, or by its architecture but all of these things will influence later on elements which determine the quality as it is seen by the user: easy to use, adaptability, portability, hardware requirements, operating speed, complexity etc. Therefore, „unseen” quality is necessary but not sufficient. On the other hand, it’s important producer’s perspective not to be in conflict with the consumer’s perspective. If this thing is happening, quality from consumer’s perspective will always be the winner. It’s worthless the software producer to be willing, for instance, to use a framework or a programming language which are facile, stable, costless but obsolete (out-of-date) if all of these have negative impact on the product-user relationship and on the users’ or potential users’ perception on product quality.

Obviously, there is a strong relation between the quality at producer’s level and the quality as it is seen by the consumer/user, in the sense that the first one determines the second one. The first is the „invisible” quality but very important for the second one, which is the „visible” quality. Starting from the idea that apparently the same product can be made using several different ways, the selection of the way it is produced and its materialization are critical elements which determine its technical, functional quality, its reliability. For example two vehicles could be apparently the same but the parts or pieces they are made of could be different from quality perspective and the consequences are going to be visible sooner or later. Many of the elements which influence quality are not visible for the user but they are very needful. There are also quality elements which are not considered as qualitative requirements by the user, even if they are included or related to that product. Rather more, they are elements which are not generating quality from user’s perspective but their absence will surely determine dissatisfaction and negative perception of quality. For instance, the possibility to use a software application without the help of the mouse, using just the keyboard, is a less distinguishable element and surely not an element for qualitative differentiation but if this feature is missing, this will generate disappointment on users when they observe this operating ergonomic problem.

All of these are elements linked to the quality from producer’s perspective, which are sensitive for the one sensed by the client. Besides, they are elements which influence, in the same time and during the time the utilization parameters of the software application.

Concluding, we emphasize on the fact the quality, as a final output, is the outcome of some inputs that we could give them a simplified name - *qualitative elements* – meaning those elements which are included into the production recipe of desired quality. The majority of quality elements, invisible for the user, have low impact on his or hers perception but, as we argued above, they are necessary for the second category of qualitative elements – *visible elements* – with impact and very important for the quality perception from consumer’s/user’s perspective. The image of an iceberg is eloquent for what we defined as being the visible and invisible sides of the quality.

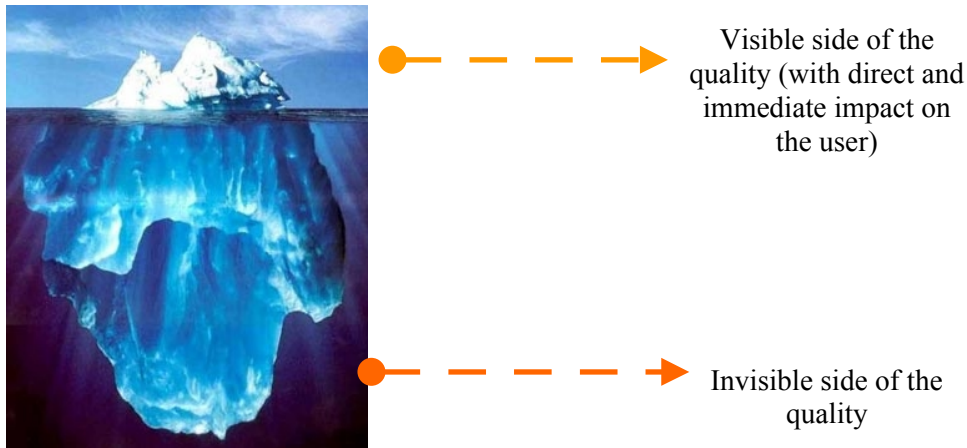


Figure 2 Visible and invisible sides of quality from customers' perception

It is well known the fact that the quality level into initial phase of the software product is strictly dependent on its complexity level. There is a reverse proportionality between them: the more complex the software product, the less the initial quality level is. This thing is not a desired one and that's why there is a period into the implementation phase allocated for testing in real environment and functional validation, named "*pilot phase*". Moreover, it is much more important as we talk about a software application of "project type" which is differentiating through authentication, originality and high level of novelty both for the application provider and its buyer.

Because it is almost impossible an application to function with no error, gaps, disparities into the initial phase, it is natural to try to find out some solutions to decrease the negative impact generated because of the lack of quality and avoiding the apparition and profound manifestation of user's dissatisfaction. There are some arguments sustaining the idea that 100% of quality or "zero" defects is extremely difficult to be assured and even inefficient and not desired. We mention some of them:

- a. time constraints coming from contract deadlines. Not keeping the delivery terms could drive to the situation of paying important penalties which become more important than the qualitative aspect. Surely, going deeply, it is necessary to avoid this situation through a better design and temporal allocation of the involved activities into the IT product creation;
- b. a benchmarking of the competing products could drive to the conclusion that it's just enough to keep the product quality at a slightly higher level but not the maximum one. Could be a market strategy not to offer the maximum possible quality level of the product from the very beginning keeping, in this way, the possibility to have subsequent and gradual appartitions of quality improved versions of the same

- software application, with positive impact on the image the client has regarding his/her provider;
- c. 100% quality is too expensive in many cases. This thing attracts corresponding costs with direct impact on the selling price of the product. Therefore, the product is positioned on a different level and it might become unaffordable for some categories of potential buyers. It is an extremely important decision making issue also in the case of software products. When a product is intended to be designed so as to be accessible for a large amount of consumers the company has to face a strong competition. It is the case when the price at higher quality comparing with the competition is a very important selection key for the buyers. The quality level in this case (when price is the first selection criteria) is not at the highest possible level. Another decisional alternative is to create a „perfect” product. The cost of perfection in this case marks out very hard on the selling price. The product is perceived as being a luxury product designated to a both limited and severe number of customers.

4. Intervals of quality

Considering the things we have already talked about, we encourage *the approach of quality by intervals*, in an evolutional and tempered approach, where the starting decisional element is the one of choosing/selecting the operational quality level for quality management deployment. In order to better understand the previous idea, we propose/suggest the next figure. It is an example where are visible three possible quality intervals. The organization could be positioned on any of quality intervals (it is its strategic decision) and the purpose is the same: to try to reach the maximum level of quality on that quality interval. The principle is to reach the maximum quality inside a quality interval.

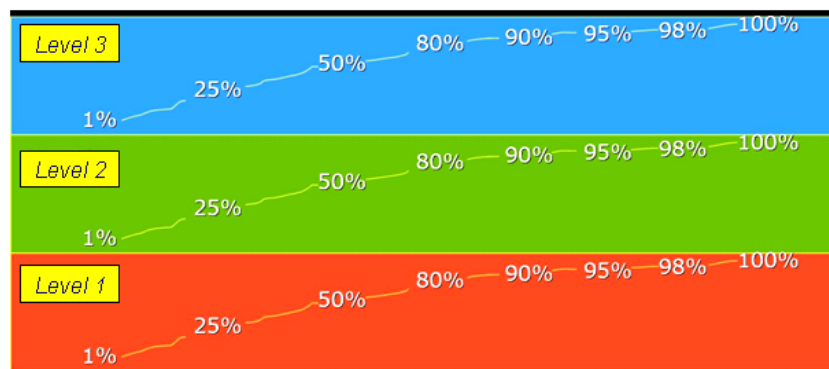


Figure 3 Intervals of quality and different levels of accomplishment (exemplification)

A relevant example comes from tourism sector where a hotel, for instance, has „3 stars”. This means a certain specific comfort standards and a guaranteed set of services. It is truly possible the hotel management not to need a different quality interval (which means 4, 5 stars in terms of tourism) because its current tourists are of that type and they couldn't afford higher prices (in accordance to higher quality standard) and because the financial results are great for the hotel. Inside this quality interval (“3 stars”) there is possible to make a lot of quality improvements including: better trained personnel, serving on bar and restaurant better organized, much more cleaning, more hospitality, better information about accommodation, services, recreation, supplementary service packs, disfunctionalities elimination through a closer look to the activities and processes carried out currently and all the problems which affect the comfort of the tourists etc.

In our opinion there are cases when quality is „imposed” by the direct competitors, besides the quality requested by the existing and potential clients. In order to keep the market position of the company, it is very important to pay attention to the actions of the direct competitors and to react accordingly. If a direct competitor starts to offer services with visible superior quality, the lack of reaction will determine a less or bigger part of customers to migrate to that competitor. In order to avoid such a situation, the involved company has to take measures such as quality improvement through economic efficiency.

However, we don't eliminate the apparition of situations which justify a jump from an inferior quality interval to superior ones if this can be possible without selling price adjustment. Theoretically, this thing is possible as a consequence of improvements with technical and organizing nature applied to the whole activity:

- the coming up of new technologies in that field: new design frameworks, new programming languages or updates of the existing ones, computers with higher performance, new improved communication technologies
- a better management of the labor force in accordance to the quality management principles: ISO standards implementation, elaboration of new work procedures
- automation of processes or activities in order to generate higher efficiency and/or human errors diminishing.

Ideally, switching from a quality interval to another one (higher) would be best to be done without sacrificing any category of the consumers in the favor of earning other categories, generally much numerous and more profitable for the organization. Such a transformation would be better to be carried out and finalized with adding of new customers and keeping the existing ones, whose fidelity cannot be betrayed, being an element of stability on the long run.

Generally speaking, migrating to a superior quality interval is an important strategic decision which has to be very well justified. Referring strictly to the specific of a standard software company, it might offer a various portfolio of software applications, some of „product” type, others of „project” type. The clients' categories the applications are referring to are different. Considering all of these, the approach of quality by intervals is justified to be applied and considered

for each application individually from the portfolio. Therefore, it is possible for some products the quality interval to be rationally lower comparing to the other products of the same company. Generally speaking, software applications of “project” type are positioned on superior quality intervals because such products are exclusively dedicated to a single client and they must rely totally on its requirements. Once the requirements are carried out, the products correspond from quality point of view. Considering all of these, we can conclude that it is useful to apply a differentiated quality management on the existing portfolio of products, especially when we talk about software domain.

5. Quality generated by the testing activity of the software application

From the software domain we give the example of testing activity applied on software applications. It is an activity with a crucial role on defects identification and elimination even in the first stages of the application’s lifecycle. It contributes substantially to the final quality, requested or desired, of the software product. A well done management of work into such a department could be and has to be completed through automation of some sub-processes taking the advantages offered by the IT domain itself. Without going into details, the testing activity could be improved a lot by designing *software utilities* or achieving them from the market, which record and deploy automatically repetitive testing sequences assuring the stability test of the software product on different phases. We talk about simple but necessary operations which are repeating identically at each improved versions of the application. For instance, on each testing version of the application open/close operations and data loading of windows must be repeated over again. Such operations are boring for the employee and they have negative effects on his/her attention and are increasing the chances of making mistakes, with negative impact on quality. Additionally, there is also the case when a sequence of operations which has generated an error is automatically “memorized” and replayed any time it is necessary on next application’s versions. It is about developing “supporting” applications for the main applications designated to the market. The costs of developing or purchasing of such instruments are, from our opinion, returned later on through the savings related to other costs and through the additional guaranteed quality. It is in another way a development of own instruments which automate some processes and sub-processes. The consequence is an increase on passing speed of some stages and, in the same time, a decreasing of the total production cycle of the software application. Another possible consequence is much more time allocated for those activities which directly contributes to the final level of quality.

As a general principle, we consider the *identification of those processes or sub-processes to which automation solutions do exist* and searching of the necessary instruments represents a major step toward reaching in a certain conditions and with acceptable costs the quality aim. Of course, at least in software

domain, not all the processes could be automated but it is economically justifiable to guide the effort toward identification of those processes proper to be automated, totally or partially, as a supplementary step toward final goal accomplishment: quality proved through clients' satisfaction.

6. Quality cost and the impact level on quality elements

We cannot talk about quality without taking into account a very important aspect of it: the cost of quality. Even if we are in the situation of making minor changes in order to improve the level of quality, these actions generate new costs. The idea is to compensate them with the savings that will come out from other cost categories as a consequence of the quality improvement, such as costs of delay, costs of removing technical, functional problems or errors etc. All of these costs are generally named "costs of non-quality". The purpose is to have a decrease on non-quality costs greater than the amount of costs generated by the actions taken to improve the quality.

Going back to the paragraph where we talked about the quality elements and their higher or lower impact on the customer's perception (paragraph: Perception of quality in software industry), we concluded that it's important to focus on those elements with high impact, which directly and strongly influence customers' satisfaction. Of course, now we ask ourselves how much these elements will cost the company to have them completed. All we can state at this moment is that all of these elements will cost more or less.

The approach of quality based on customers' perception means to discover and focus on the quality elements with important impact on them. The elements having high perception impact could be categorized based on the cost involved on their accomplishment. Because the cost is a critical decision element, it's important to try to estimate the amount of money necessary to be expensed in order to implement those important quality elements.

However, it's important to realize that not all the time the quality elements with high perception impact cost a lot. There are examples that prove the fact that elements with high quality impact could be completed quite easy, with rather low expenses.

From efficiency perspective, the idea is to have a comprehensive approach regarding the quality elements and its impact on the customers' perception. It's important to be able to make two classifications of the quality elements. One is the one which structures the quality elements by the impact on customers' perception. It's the classification which helps the individuals involved to select first the elements with high impact. The second classification helps the decision makers to structure the quality elements by the costs necessary to materialize them. Such classification offers the necessary information to identify those elements which cost a lot and those which are not so costly.

Making a combination between the both classifications, it generates some kind of matrix with two dimensions where each quality element is positioned on it

by the two characteristics already mentioned: impact and cost. Therefore a diagram could be graphically represented. Its simplest form of representation can be seen on the next figure:

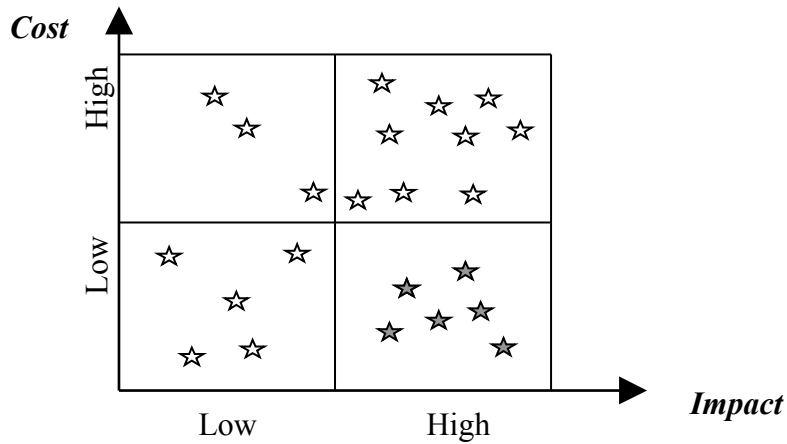


Figure 4 Quality elements structure by impact and cost characteristics

By analyzing the above figure we conclude that we talk about four resulting frames by combining impact and cost characteristics of each quality element. The frame which corresponds to “high impact-low cost” combination is the most desirable one. If there are quality elements inside this frame, they come on the first place, they have the highest priority on the quality implementation process. Of course we talk about the implementation order of quality elements. This categorization doesn’t have the role to eliminate quality elements by analyzing the diagram. Could be situations when quality elements with low impact and high cost are still necessary to be implemented. But, if it is possible, the diagram gives a justified order based on these characteristics of efficiency.

As a general conclusion, it is very hard to depict quality through simple rules of how it could be applied or implemented. Universal principles of quality implementation with total success aren’t possible. The specific of each domain or industry marks the way the quality is seen, approached, implemented. The software industry is one of the most particular domains of activity. Quality is essential and it has to be treated with great attention minute by minute, day by day. All we intended to do was to describe a systematic approach regarding the quality in the first phases of implementation, applicable on the software industry, which generates a rational and logical framework based on the principles of economic efficiency. It eases the path to the final goal: an efficient quality system based on the customers’ satisfaction.

Bibliography

1. Gillies, A., *Software Quality – Theory and Management*, International Thomson Computer Press, London, 1997
2. Goetsch, D., L., Davis, D., *Introduction to Total Quality – Quality Management for Production, Processing and Services*, Prentice Hall Inc, New Jersey, 1997
3. Ilieș, L., *Managementul calității totale*, Editura Dacia, Cluj-Napoca, 2003
4. Juran, J., M., Godfrey, B., A., *Manualul calității Juran*, Mc Graw Hill, 1999
5. Mitonneau, H., *O nouă orientare în managementul calității*, Editura Tehnică, București, 1998
6. Năstase, M., “Importance of Trust in the Knowledge Based Leadership”, *Revista de Management Comparat International / Review of International Comparative Management*, Vol. 10, București, Editura ASE, 2009, pp. 518-526
7. Năstase, M., *Lideri, leadership și organizația bazată pe cunoștințe*, București, Editura ASE, 2007
8. Năstase, M., *Cultura organizațională și cultura managerială*, București, Editura ASE, 2004
9. Parker, G.W., *Costurile calității*, Editura Codecs, București, 1998
10. Roncea, C., “Principiile asigurării calității în domeniul tehnologiei informațiilor”, în „Q Media”, nr.2/2000
11. Walmüller, E., *Software Quality Assurance – A Practical approach*, Prentice Hall International, London, 1994
12. *** *Revista de Management Comparat International / Review of International Comparative Management*, (2008) Vol. 9, București, Editura ASE
13. *** *Revista de Management Comparat International / Review of International Comparative Management*, (2009) Vol. 10, București, Editura ASE