Quality Model for M-Learning Applications

ADRIAN VISOIU, LORENA BATAGAN, CATALIN BOJA
Economic Informatics Department
Academy of Economic Studies
Romana Square No. 6, Bucharest
ROMANIA
adrian.visoiu@csie.ase.ro; lorena.batagan@ie.ase.ro; catalin.boja@ie.ase.ro

Abstract: A high level of quality is an important objective for any software application because it has a major impact on the user-software relation. The concept is a complex one and a representative analysis must take into consideration particular characteristics and properties of a specific category of software products. This paper defines a quality model to be used to assess characteristics levels for M-learning applications. The model is represented by an aggregate indicator that quantifies the effect of all the factors taken into consideration. In order to validate the model are proposed two approaches to be used on a real M-learning product. The role of the model is to evaluate quality levels of M-learning applications and to allow developers to improve it or to allow users to rank different applications. The factors included in the quality model have been identified in other researches, [2], from a user point of view.

Key-Words: quality, software, m-learning, characteristic, mobile devices, mobile technologies.

1. Introduction
In our age the evolution depends on the access to more and more knowledge and information.
Today evolution is based by produce of knowledge and because of this is more important to use efficiently the technologies and information.
Technology in fact wouldn't exist without innovation. All technology we have was a result of someone inventing a better way of doing something, a better tool.
Learning document processing is the main area of the implementation of information and communication technologies into education today. [3] said that have in view learning document processing are many activities connected with the quality of education. ISO/IEC 9126 specifies quality as “the totality of features and characteristics of a product or service that bears on its ability to satisfy stated or implied needs”.
Educational institution are currently confront with the dilemma of the use of m-learning in teaching and learning and developing strategic plans and processes that will take them forward in sustainable ways.[1]
This paper discusses whether m-learning is a solution for improve quality in educational systems. Also, the paper defines a quality model for m-learning application.
The m-learning represent a solution to access the knowledge and information anytime and anywhere, so we can say that it help us to learn more and to make things better.
The place of m-learning in education is clearly aligned to the issue of quality. Educational institution tries to use mobile technologies to improve the teaching and to help users to understand and to access the information when they have free time.

Fig.1 M-learning in our society

2. Characteristics of mobile learning
The emphasis in m-learning is on learning rather than teaching. The objective is to create active learning environments that empower learners to take responsibility for their own learning through enhanced thought stimulation and encouragement of exploration, discovery, adaptation, innovation, and creativity.[L4]
By using mobile device, m-learning solutions can benefit from the potential of mobile web-based interactions. The web and mobile devices help to communicate better, because it is the media that allows its users to communicate with each other directly, publicly, and instantly. Mobile learning (m-learning) is
the process of delivering educational content to individuals through mobile technologies and devices. [2]

Mobile learning is offering a large number of benefits to their users:

- Help people with disability
  M-learning is based on the idea that the use of mobile device can facilitate the access to learning (and training) resources, having the potential to offset and overcome some of the disadvantages caused, for example, by disability. For some people these benefits are achieved by a standard computer working in a standard way: for others the empowerment is achieved by using alternative and adaptive technologies to overcome the apparent disability.

- Flexible kind of learning
  Secondly, learning can take place anytime, anywhere. Learning can happen across locations, or mobile learning takes advantage of learning opportunities offered by portable technologies. Students are overtime in go, so they are interested by more flexible kind of learning.

- Collaboration with everyone
  Through mobile learning everyone uses the same content, which will in turn also lead to receiving instant feedback and tips. This learning will reduce cultural and communication barriers between faculty and students by using communication channels that students like.

- Motivation people
  Fourthly, multimedia resources can make learning fun. With this kind of learning, it is much easier to combine gaming and learning for a more effective and entertaining experience. This is a great point of view because most of students are learn more when they are do something just in play.

- Accessible from anywhere
  Mobile is accessible virtually from anywhere which provides access to all the different learning materials available.

- Portability
  Moreover, the small size and weight of mobile devices means they can be taken to different sites or moved around within a site.

- Reduce of time
  Technology expands time and compresses space. There is no need for students to be in the same place at the same time in order to have a sense of live exchange. So, in this mode university can build relationship on different continents.

Mobile technology can effectively support a wide range of activities for users. It provides for each user to have a personal interaction with the technology in an authentic and appropriate context.

Educational mobile software application represents great benefits for both students and teachers. The increase in use is attributed to the portability that mobile device offer, making it possible for each user to have access to a computer at any time and any place. No longer dependent on computer labs for computing capability, students can work on mobile devices right at their desks. Students can also take mobile devices on field trips to collect, store, and analyze data on site.

Quality plays an integral role in all aspects of our life. [2] see the educational process like a complex service which involves a producer and a consumer. Delivering good quality services on time and on budget is every educational institution goal. The purpose of m-learning application is to improve the quality of learning, to identify and report successful results, and also to identify and report those activities and processes where we find space for improvements and use this as a reference for future applications.

An m-learning application [6] is a special type of software application particularly designed to be used on mobile processing units with limited processing power, storage memory and input capabilities such as mobile phones, smartphones, PDAs, navigation assistants, mobile guides, etc.

Like [2] said that the software characteristics describe the quality of the m-learning application. In ISO 9126 an international standard for the evaluation of software product, software quality characteristics represent a set of attributes of a software product by which its quality is described and evaluated. So, we can say, start from [5], that software quality of mobile application may be evaluated by the following characteristics:

- Accessibility: A set of attributes that bear on the ability of software to be use from anywhere and anytime.
- Functionality: A set of attributes that bear on the existence of a set of functions and their specified properties. The functions are those that satisfy stated or implied needs.
- Reliability: A set of attributes that bear on the capability of software to maintain its level of performance under stated conditions for a stated period of time.
- Usability: A set of attributes that bear on the effort needed for use and on the individual assessment of such use by a stated or implied set of users.
- Efficiency: A set of attributes that bear on the relationship between the level of performance of the software and the amount of resources used, under stated conditions.
- Maintainability: A set of attributes that bear on the effort needed to make specified modified modifications
- Portability: A set of attributes that bear on the ability of software to be transferred from one environment to another

3. Quality model

In [2] specialists chose a list containing the most important characteristics considered to contribute in assessing m-learning applications. For each quality characteristic, indicators are built in order to objectively measure each level. Indicators must take values within [0; 1] range in order to be easily compared and aggregated. When the criterion is to be maximized, then the zero value shows the absence of the quality characteristic, the one value showing the best level proposed for that characteristic. If the criterion is to be minimized, then the zero value of the indicator shows the presence of a desired level for the quality characteristic, while the one value shows an undesirable level, or the absence of quality. Table 1 shows indicators, the characteristic and if it is to be minimized or maximized in order to obtain a good quality of the m-learning application.

User surveys determine limits of acceptance for the defined characteristics of the product. For the loading time quality characteristic, the following indicator is defined:

\[ I_L = \min \{LT_{rec}, LT_{max}\} \]

where
- \(IL\) – the indicator ranging in [0;1], showing the degree of quality regarding the loading time
- \(LT_{rec}\) – the recorded loading time for the m-learning application
- \(LT_{max}\) – the maximum accepted loading time determined by user opinions.

It is useful that needed information to be accessed in little time. There are cases when successive searches are run, the access time and the number of inputs to access a certain resource becoming very important. For the path length quality characteristic, the following indicator is defined:

\[ I_P = \min \{PL_{res}, PL_{max}\} \]

where
- \(IP\) – the indicator for the path length quality characteristic, ranging in [0, 1];
- \(PL_{res}\) – average path to a resource in the application;
- \(PL_{max}\) – maximum accepted path accepted by users.

The homogeneity of the interface has impact on the learning curve of the application, having direct influence over the acceptance of the application by the user group. The m-learning has different components such as question answering for quiz tests, or menu options. If answer options for quiz tests are implemented using all of combo-boxes, list-boxes and radio buttons, this induces less homogeneity than using a single type of control. For the homogeneity degree of input data process, the following indicator is used:

\[ I_H = \frac{NUIC}{NIIC} \]

where:
- \(IH\) – the degree of homogeneity for input components
- \(NUIC\) – number of unique types of components;
- \(NIIC\) – number of ways of implementation for components.

In m-learning applications, users are required to enter information in order to complete tasks. It is desired that the amount of required information to be small. The indicator measuring this characteristic is given by:

\[ I_R = \min \{NC_{task}, NC_{max}\} \]

where
- \(IR\) – the indicator showing the level of the characteristic
- \(NC_{task}\) – the average number of controls accessed to complete a task;
- \(NC_{max}\) – a maximum limit for the number of accessed controls that makes the user to give up his interaction with the application; this limit is determined by user surveys.

Continuity of interaction is given by the presence of navigation elements that permit the user to continue working with the application from a certain point in the application flow. The indicator built for this characteristic takes into account a certain node in the application flow, incoming connections from other nodes to that node and outgoing connections from that node to other nodes.

For a certain node \(i\), the level of continuity is given by the indicator
\[ CIN_i = \frac{\min\{IN,OUT\}}{\max\{IN,OUT\}} \]  

where

- \( CIN \) – continuity of interaction for application node \( i \);
- \( IN \) – number of incoming connections to node \( i \);
- \( OUT \) – number of outgoing connections from node \( i \).

Taking into account all the nodes in the application flow, the indicator for the continuity of interaction at application level is given by

\[ I_{CN} = \frac{\sum_{i=1}^{nn} CIN_i}{nn} \]  

where

- \( I_{CN} \) – indicator for continuity of interaction at application level;
- \( CIN_i \) – continuity at node \( i \);
- \( nn \) – the number of nodes in application flow.

The uniform character of layout components is given by the indicator:

\[ I_{CP} = \frac{NULC}{NILC} \]  

where

- \( NULC \) – number of unique layout components;
- \( NILC \) – number of ways of implementation for layout components.

This indicator stands for the way the application shows the user outputs, reports and options.

The fact that all proposed indicators range in \([0; 1]\) make them easy to interpret and also easy to aggregate in order to obtain a global quality indicator for the m-application.

<table>
<thead>
<tr>
<th>Indicator</th>
<th>Characteristic</th>
<th>Criterion</th>
</tr>
</thead>
<tbody>
<tr>
<td>( I_L )</td>
<td>Loading Time</td>
<td>minimum</td>
</tr>
<tr>
<td>( I_P )</td>
<td>Path Length</td>
<td>minimum</td>
</tr>
<tr>
<td>( I_H )</td>
<td>Homogeneity Degree</td>
<td>maximum</td>
</tr>
<tr>
<td>( I_R )</td>
<td>Required Information</td>
<td>minimum</td>
</tr>
<tr>
<td>( I_{CN} )</td>
<td>Continuity</td>
<td>maximum</td>
</tr>
<tr>
<td>( I_{CP} )</td>
<td>Uniform Complexity</td>
<td>maximum</td>
</tr>
</tbody>
</table>

An aggregated indicator is built taking into account the individual levels for each quality characteristic. The aggregated indicator ranges also in \([0; 1]\). As the global quality of the m-learning applications is to be maximized, then the zero level shows the absence of quality characteristics and the one level shows the achievement of the best quality with regard to the imposed requirements.

The analytical form for the aggregated quality indicator must take into account the importance weights chosen by specialists

\[ I_Q = \frac{0.12 \cdot I_{CN} + 0.11 \cdot I_P + 0.17 \cdot (1 - I_L) + (1 - I_P) + 0.09 \cdot I_{CP}}{0.77} \]  

After operating the constants, the model becomes:

\[ I_Q = \frac{0.12 \cdot I_{CN} + 0.11 \cdot I_P - 0.17 I_L + 0.09 \cdot I_{CP} + 0.45}{0.77} \]  

When the quality is maximum on each component, thus \( I_{CN}=I_{CP}=1 \) and \( I_L=I_P=I_R=0 \), the global quality is also maximum \( I_Q=1 \). When on each considered component has no quality, thus \( I_{CN}=I_{H}=I_{CP}=0 \) and \( I_L=I_P=I_R =1 \), also the global quality reaches its minimum \( I_Q=0 \).

When analyzing an m-learning application, specialists have to define acceptance thresholds for individual and global quality. If the global quality indicator is less than 0.75, \([9]\), the product is rejected, otherwise it is accepted for use.

The aggregated quality indicator is used to assess the m-learning application quality over time or to compare two applications with respect to the quality characteristics that are specific for m-learning domain.

The weights taken into account in the analytical expression of the model are determined after a user survey presented in \([2]\). If the importance of the characteristics change, or other indicators are built to assess each characteristic, then the model must be changed. In the hypothesis that the indicators used range in \([0; 1]\), and they are to be minimized or maximized in order to obtain a certain level of quality, the general form of the model is:

\[ I_Q = \frac{\sum_{i=1}^{nm} \alpha_i \cdot \text{IMAX}_i + \sum_{i=1}^{nm} \beta_i \cdot (1 - \text{IMIN}_i)}{\sum_{i=1}^{nm} \alpha_i + \sum_{i=1}^{nm} \beta_i} \]  

where

- \( I_Q \) – the aggregated quality indicator to be maximized.
Applying the proposed model requires customization. Quality managers must identify the abstract elements that build up each indicator. Those elements need to be put in relation with concrete elements from the application. Information has to be collected regarding the application. Indicators are assessed individually in order to take decisions that influence certain aspects of the quality. The aggregated indicator is then computed in order to assess quality at global level and compare it with previous estimations or measurements or compare the studied application with other with the same objectives.

4. Model validation

In order to define and implement models for economic and social phenomenon, it is necessary to achieve model validation. Validation is a process by which the model is used to solve several problems. Over time the solutions quality is analyzed, comparing them with actual developments of the modelled phenomena. If an important share of the model based solutions does not differ significantly from the actual developments, it concludes that the model is appropriate and future uses will lead to solutions, also without significant differences from actual values.

There are many methods of validation for models [8]. For the m-learning applications quality model there are used two of them. One is based on an experimental approach and the other relates to a method based on tests using statistical methods.

Experimental validation of software quality indicators and models involves the study on sufficiently large model behaviour and the comparison of quality estimated values with the actual recorded levels, [7]. To define the framework conditions of the experiments it is made the observation that during experimental validation there will be met three situations:

- **case 1**: the model corresponds to all conditions and it is validated;
- **case 2**: the indicator does not correspond to any of the conditions is invalidated;
- **case 3**: indicator meets certain terms and conditions but not all of them; the experiments continue until it is reach case 1 or 2; if the process continues without leading to a final result, it is concluded that validation results are uncertain and the use of that indicator is not a risk free process.

Experimental validation is a process difficult to achieve due to specific conditions regarding use of software:

- in order to test analyzed software must be provided a uniform use conditions environment;
- software product distribution must be made to a sufficiently large number of users.
- user registration is required to make correct measurements regarding user behaviour in exploiting the software from the viewpoint of occurrence time, duration and type of events.
- it is needed to establish connections between software producers and users in order to build a database for software behaviour.
- it is necessary to create a system with distinct functions, with bounded inputs and outputs so that each user can be clearly identified as part of this system and its records regarding software behaviour have favourable consequences through the circulation of information between users and producers.
- it is defined a test sample, used for an assessment system at producer level and on the other hand, for an independent reference system. The objective of experimental validation is to compare the two systems and to accept or reject the consistency of the results, which were provided by the model, against the reference system.
- there are defined the conditions under which the model is stable, the model with estimated coefficients is used directly, without modifications, and also it is defined the moment when the model coefficients must be recalculated.

To obtain experimental validation conditions, there are defined flows of electronic documents, automated procedures for collecting data and there are build data management programs that reflect the behavior of a software product. Not reaching a homogenous working environment affects information quality, with negative effects on the whole process of model validation.

Experimental validation involves performing experimental measurements. In most cases, software users carry out experiments mostly during the training stages when they learn how to work with the software. Attempts, successful or not, are aimed to highlight, at user level, how data is entered, what results are obtained. Though are
experiments, test, all the processes done in the stage of accommodation with the software, are not considered activities required by the model validation. Model experimental validation involves working under normal conditions with software, which is not the case of learning to use it. Validation does not involve creating experimental laboratory conditions, like the test scenario. The experiment is a result of current use, in practice, of the software, and the measurements are made on the spot, at each user.

Efficient software performs automated records, and automatically saves behaviour data. In the absence of such level of performance, one valid alternative is to collect and to transmit data, accepting that the activity is performed by an algorithm defined by the manager of the model validation process.

Statistical validation methods involve collecting models, data series regarding models behaviour in order to identify and establish links between the model structure and the errors.

Validated model is used without major risks of introducing errors between the estimated and the actual level of outcome variables. If the phenomenon dynamics is recording qualitative leaps, the process is remodeled and validation is resumed for the new model. The validation process is redone periodically. In order to revalidate the model it is assumed that:
- parameters are recalculated;
- the model structure is reanalyzed and modified;
- analyzing new model values it is taken the decision to accept or to reject the new model;

Knowledge gained during the use of current model helps experts to define aggregate indicators to determine the optimal start time of the remodeling, estimation and revalidation processes.

The construction of a model takes into consideration:
- a set of factors $F_1$, $F_2$, ..., $F_{NRF}$ that influence measured characteristics; they are ranked in terms of importance;
- methods of quantifying the factors $F_1$, $F_2$, ..., $F_{NRF}$ to obtain levels $x_1$, $x_2$, ..., $x_{NRF}$;
- multiple software programs and data results, input or results, which are assessed from the point of view of analyzed characteristics.
- working hypothesis regarding indicator analytical form.

There are proposed different methods for validating the properties of the models. Each model associated to a software quality characteristic is proposed and used to evaluate products to make comparisons and to substantiate whether or not it is profitable to develop an application with a high level of complexity.

The model is significant, if it is easy to calculate, without requiring a long time for data recording. It becomes important over time and it is necessary to be calculated for each M-learning application delivered to users to highlight its properties compared to other programs.

The model user, the M-learning application developer, has must validate its properties before using it. Validation of the model representativeness is based on establishing the extent to which it is used in practice and how accurate are resulting values regarding the analysis objective.

The model is statistical validated in relation to three properties, [9], namely: sensitivity, non-catastrophic and the non-compensatory. Sensitivity is an essential property because it highlights how the small and large variations of independent variables are generating variations of the dependent variables, the model resulting values.

Model factors are analyzed and there are defined intervals of normal values for independent variables. This it is used to highlight how the dependent variable evolves.

Validation of model sensitivity unlike its analysis is a process that arises naturally through software development, because it is not possible to directly manage small or large variations management. The producers records factors values during the software development process. This data is processed using statistical methods to validate models sensitivity.

Indicators should be non-compensatory, ie variations in the levels of independent variables should not conduct to equal levels of outcome variables. This requirement is an important factor of influence in the analysis, because it represents the base for the uniqueness assumption regarding situations included in the study. To ensure the representativeness and significance of the results must be avoided the cases in which same results are obtained for different levels of input variables.

A model is catastrophic if there are any particular values that make impossible to obtain a value for the outcome variables.

Catastrophic nature of an model does not involve the use of another indicator that lacks this attribute, but only draws attention to particular situations. These are defined by values of variables that are measured under specific conditions. Use of such indicators should be preceded by a clear definition and analysis of the problem.

Taking into account rules of mathematical calculation, the catastrophic character is generated by situations defined by:
- denominator cancellation;
negative values or equal to 0 for the logarithmic function argument;
radical of negative values;
situations in which computing operations usually generate error.
It is necessary that a model should be examined from all points of view.
This allows the discovery of those particular situations in which the components values generate errors. This will also define the basis of a building criterion for the sample of the analyzed values.
For example, if the indicator is sensitive, there are selected test values for independent variables close to each other in order limit the effects on the sample dispersion. If the indicator is characterized as a compensatory or catastrophic one, there are removed from the sample, those elements which verify the properties assumption and that produce errors in results.

5. Conclusions
Quality models are used to reduce the complexity of the quality concept regarding a software product by identifying a limited set of influencing factors and by providing the methods to measure quality levels, thus providing a way to translate relative levels as good, worse or perfect to numeric and objective levels.
Because each category of software products has a great influence on the quality system of software characteristics it is needed to define particular quality models that will take into account specific properties.
The proposed quality model is going to be analyzed and validated using data from a real M-learning application. Results presented in this paper are part of a larger project that aims to implement a complex M-learning application. This approach is important for the developing process because it helps identify factors that are considered important for the resulting software application.
This paper presents some results of the research project IDEI 2673: "Project management methodologies for the development of mobile applications in the educational system", financed within the framework of Romanian IDEI research program.

References