Monograph: Experiences and Advances in Software Quality
(published jointly with Novática*)
Guest Editors: Darren Dalcher and Luis Fernández-Sanz

1 Presentation: Improving Quality in Business Processes, Products and Organizational Systems — Darren Dalcher and Luis Fernández-Sanz
6 Preventative Software Quality Control: Using Human Checking to Change Defective Human Practice — Tom Gilb and Lindsey Brodie
14 The Software Process Improvement Hype Cycle — Miklós Biró
21 Quality Going for Gold — Derek Irving and Margaret Ross
26 Can Teamwork Management Help in Software Quality and Process Improvement? — Esperança Amengual-Alcover and Antònia Mas-Picacho
34 Evidence-based Software Engineering and Systematic Literature Reviews — Barbara Kitchenham, David Budgen, and O. Pearl Breton
42 Software Project Success: Moving Beyond Failure — Darren Dalcher
51 Software Measurement for Better Project and Process Quality — Christof Ebert
67 A Quality Evaluation Model for Web2.0 e-Learning Systems — Stephanos Mavromoustakos and Katerina Papanikolaou

UPENET (UPGRADE European NETwork)

75 From Mondo Digitale (AICA, Italy)
History of Computing
The Turing Test: History and Significance — Giuseppe O. Longo

CEPIS NEWS

89 Selected CEPIS News — Fiona Fanning
Experiences and Advances in Software Quality

A Quality Evaluation Model for Web2.0 e-Learning Systems

Stephanos Mavromoustakos and Katerina Papanikolaou

Web2.0 is used in e-Learning to transfer the qualities of social networking to the virtual classroom. The continuous evaluation of the Web2.0 e-Learning systems requires the use of quality evaluation models in a continuously evolving environment. This paper proposes a quality evaluation model for Web2.0 e-Learning systems called EEQM which focuses on three main components: a) software quality factors, b) pedagogical requirements and c) Web2.0 elements. The EEQM model involves the combination of three methods for estimating the quality of e-learning systems: a) Frequency, b) Median, and c) Total Quality.

Keywords: e-Learning, Evaluation Model, Software Quality, Web Quality, Web 2.0.

1 Introduction
Software quality evaluation is essential to the production of software products that are reliable, efficient, understandable and acceptable to their stakeholders [1]. In developing Web applications it is also necessary to build and utilize sound evaluation models and tools so as to facilitate and ensure the continuous quality of those applications. In recent years, several style guides, lists of guidelines, and design principles have emerged to assist developers in the development process. While these guidelines and techniques are helpful in the Web design process, they do not necessarily constitute an evaluation method in themselves.

Several researchers have proposed various recommendations or evaluation techniques for Web applications [2][3][4][5][6][7]. However, quality evaluation of e-learning applications is a new and often neglected issue. With the advent of Web 2.0 in almost all areas of Web applications as well as in e-learning applications, a sound and practical evaluation model is necessary.

The E-learning Evaluation Quality Model (EEQM) proposed in this paper is based on a three axis approach: a) software quality factors, b) pedagogical requirements, and c) Web2.0 elements. The quality approach is based on internal and external evaluation criteria. Internal refers to organization Web experts, and external to a group of users (learners). The EEQM model combines three methods for estimating the quality of e-learning systems: a) Frequency, b) Median, and c) Total Quality.

The rest of the paper is organized as follows: Section 2 describes the three main quality criteria, Section 3 explains the EEQM model, and Section 4 describes step-by-step the proposed evaluation process. Finally, Section 5 draws the concluding remarks and suggests some steps for future work.

2 Quality Criteria
This paper aims at providing a systematic method for evaluating Web 2.0 e-learning systems, utilizing three main components: a) software quality factors, b) pedagogical requirements, and c) Web2.0 elements.

2.1 Software Quality Factors
The software quality factors component proposed in this quality evaluation model takes into account ISO 9126 and its decomposition into several quality factors, as proposed in the context of Web engineering and W3C standards and recommendations.

ISO 9136
ISO 9136 refers to:
Usability: Issues such as understandability, learnability, friendliness, operability, playfulness, and ethics are vital design factors that Web engineers cannot afford to miss. The system must be implemented in such a way as to allow for easy understanding of its functioning and behaviour, even by non-expert Internet users. User interface aesthetics, consistency, and ease of use are attributes of easy to learn systems with a rapid learning curve. By keeping a user profile and taking into consideration human emotions, e-learning systems can deliver related messages to the user, whether a welcome message or an order confirmation note, thus enhancing the friendliness of the system. Playfulness is a feature that should be examined to see whether the application requires this characteristic and, if so, to what extent. E-learning systems must reflect useful knowledge by looking at human interactions and decisions.

Functionality: The system must include all the necessary features to accomplish the required task(s). Accuracy, suitability, compliance, interoperability, and security are issues that must be investigated when designing an e-learning system, to ensure that the system will perform as it is expected to. An e-learning application must have search and retrieve capabilities, navigation and browsing features, and application domain-related features.

System Reliability: Producing a reliable system involves understanding issues such as fault tolerance, crash frequency, recoverability and maturity. The system must maintain a specified level of performance in the event of software faults with the minimum number of crashes possible. It must also have the ability to re-establish its level of performance. A system must consistently produce the same results, and meet or even exceed users’ expectations. The e-learning application must have correct link recognition, user input validation, and recovery mechanisms.

Efficiency: An e-learning system’s goal is usually to increase productivity, decrease costs, or a combination of both. Users expect the system to run in an efficient manner in order to support their goals. System’s response-time performance, as well as page and graphics generation speed, must be high enough to satisfy user demands. Fast access to information must also be examined throughout the system life to ensure that users’ requirements are continuously met on the one hand, and that the system remains competitive and useful on the other.

Maintainability: Some crucial features related to maintaining an e-learning application is its analysability, changeability, stability, and testability. The primary target here is to collect data that will assist designers to conceive the overall system in its best architectural and modular form, from a future maintenance point of view.

Given the rapid technological changes we are seeing, especially in the area of Web engineering, as well as the rigorous demand from users for continuous Web site updates, easy system modification and enhancement, both in content and in the way this content is presented, are also success factors for the development and improvement of an e-learning system.

W3C
The World Wide Web Consortium (W3C) aims to provide Web standards for Web interoperability. In addition, W3C provides a Markup Validator, a tool that verifies whether a document actually follows the rules of the markup language used. This is the process of checking a Web document against the grammar (typically a DTD) it is using.

W3C also provides guidelines for a quality Web application on:

- Title.
- Avoiding the “click here”.
- Using redirects.
- Using headers.
- Using the alt attribute.
- Using a doctype.
- Using the <link> element.
- Using CSS rather font.
- Using international date format.
- Using PNG rather GIF.
- Using headings.
- Choosing and managing URIs.
- Using quality content.
- Using class with semantics in mind.
- Using unordered lists.

2.2 Pedagogical Requirements
Transferring the dynamic nature of learning to the new e-learning environment, maintaining student individuality and differentiation according to personal preferences and abilities, as well as motivating and inspiring students, are all key factors for the acceptance of the new learning environment [8][9].

The key factors are identified as follows:

Identification of learners’ needs: The e-learning environment should be shaped according to predefined learners’ needs and the course required pedagogical outcome.

Structuring of the pedagogical material: The pedagogical material should be constructed in a way that facilitates the successful transfer of the required knowledge.

Enhancement of the e-learning environment: The e-learning environment can be used either as complimentary or in parallel to the real classroom environment. In either case the e-learning environment should adhere to the basic mechanisms and functions of the real environment. In the case of pure distance learning this enhancement is even more imperative.

Motivation for student participation: Transferring to a virtual environment is not always straightforward or easy. Students are not always willing to use a virtual environment for a number of reasons, such as the difficulty of the e-learning tool, the non-intuitive nature of the environment, problems of low interactivity, etc. The e-learning environment should have mechanisms to allow questions/answers sessions similar to those in a traditional classroom. The e-learning environment should be able to offer the students a basic problem solving mechanism. Mechanisms such as online tutorials, contact with the instructor, reference to use-
ful resources, and even access to a technical helpdesk would offer students support and help.

The establishment of collaborative mechanisms among students: In the virtual environment the student can be easily isolated and separated from the rest of the class. This is usually avoided in the real classroom and should be avoided in the virtual classroom too, by organizing and operating on a collaborative basis so that students can interact and communicate.

The use of relevant tools and components for the support of any specific solution: Depending on the target student audience and the required learning outcome, the appropriate tools should be implemented and differentiated accordingly. Vocational training requires different solutions from educational training and undergraduate training has different pedagogical targets from undergraduate training. Tools and components can be utilized to enhance the e-Learning environment more efficiently.

The right mix of the learning processes implemented: The most important learning processes are identified as follows: analysis, synthesis, reasoning, judging, problem solving, collaboration, simulation, evaluation, presentation and relation. These processes should be used dynamically to construct the learning scenario for each course and student.

2.3 Web 2.0 Elements

One of key factors affecting e-Learning effectiveness was identified to be the lack of interactivity [10][11][12]. The isolation of the learner behind a screen leads to falling motivation, loss of interest, and failure. In parallel, the same need for user interactivity has its effects on the Internet and its use. Internet users realized that the asymmetrical flow of information made most of them content consumers rather than content providers. But the Internet can inherently provide access to users both as content consumers and content creators.

The realization of this fact led to the establishment of Web2.0. Web2.0 is harnessing the Web in a more interactive and collaborative manner, emphasizing peers’ social interaction and collective intelligence, and presents new opportunities for leveraging the web and engaging its users more effectively. In the last three years, Web2.0, ignited by successful Web2.0-based social applications such as wikis and blogs and application specific software such as my MySpace, Flickr and YouTube, has been forging new applications that were previously unimaginable.

In this section, we present the basic applications enabling social networking and used in e-Learning interactivity.

Weblogs

The Web offered the perfect medium for immediate press releases and fast dissemination of news and information. The publishing on the Web of an individual’s diary, with the thoughts, views, comments and positions created the revolution of Web-Logging or Blogging.

The term was initially coined by Jorn Barger in 1997 and in its simplest form is a website with data entries, presented in reverse chronological order [13]. This is the outcome of a common need for the sharing and expression of thoughts, criticism and experiences by individuals and was, from its outset, one of the strongest tools of the Internet. A Blogger is the owner of a Blog and contributing to a Blog is called blogging. Each Weblog is part of the Blog-o-sphere. The number of existing blogs is rapidly growing and there seems to be no end in the near future. The new-found “democracy” has many supporters but there are concerns regarding the extent of freedom of speech. While the concept of an on-line diary is far from new, the popularity of Blogs is increasing rapidly.

There are two main reasons for their success, as identified by other researchers [14]:

1. Personalization: A Blog is personal and expresses the authors’ views and ideas but others can contribute too. A Blog provides directness of communication and a forum for discussion and the exchange of ideas.

2. Usability: The crucial factor for the success of Web2.0 applications is ease of use. They are not aimed at people with technical computer programming skills. Everyone is able to contribute to the WWW and become a content creator by clicking on his/her weblog, logging on, and writing with the help of a WYSIWYG-Editor.

The amount of information trafficking in Blogs can be enormous, and to cope with this a personal overflow RSS (Really Simple Syndication) technology is used. With the help of XML structure, an RSS-Reader can provide feeds from subscribed Blogs or other applications. The big advantage is that new information can be read without opening a site. Furthermore, the possibility of using Aggregators and Search functions help to make the information consumption more efficient.

The popularity of blogs has raised concerns regarding the legal and other ramifications of releasing confidential information, use of language etc.

Wikis

A wiki is a simple yet powerful Web-based collaborative-authoring (or content-management) system for creating and editing content. It lets anyone add a new article or revise an existing article through a Web browser. Users can also track changes made to an article. The term wiki is derived from the Hawaiian word wikitiki, which means fast or quick. The user-generated online encyclopaedia Wikipedia (http://en.wikipedia.org) is a wiki.

Wiki features include:

- A wiki markup language. "Wikitext" provides a shorthand way of formatting text and linking external documents and contents.
- Simple site structure and navigation. Contributors can create new pages and easily link one page to another. Because a blog site’s hierarchy and structure is flat, navigation is simple.
- Simple templating. When a page of wikitext is requested, wiki software converts the wiki markup to HTML and creates links between pages, then wraps this converted
content in a template to provide a consistent look to all pages in the wiki.

- Support for multiple users. Hyperlinks to pages within the wiki are created automatically. Wiki software makes links based on the page’s title, so the author does not need to use, remember, or type long URLs to link one page to another within a wiki.

- Simple workflow. You can write or edit and publish without editorial oversight or approval. Content in a wiki is managed through change monitoring and the wiki’s ability to roll back to a previous version and prevent spam. You can also control user access and privileges, if required.

- A built-in search feature. You can search for specific information or topic within a wiki using associated keywords.

Wikis facilitate collaborative work and this is their main difference from Blogs. Due to this collaborative ability, wikis can significantly enhance the learning environment.

**Mashups**

A Web mashup is a Web page or Web site that combines information and services from multiple sources on the Web. Web mashups can combine information and/or complementary functionality from multiple Web sites or Web applications. A Web mashup server lets you connect, collect, and mash up anything on the Web, including data on some backend systems. There are seven major categories: mapping, search, mobile, messaging, sports, shopping, and movies. More than 40 percent of mashups are mapping mashups. [15] Several other new-breed Web applications similarly integrate multiple services under a rich user interface.

Typical applications are HousingMaps <http://www.housingmaps.com>, that display sales and rental information from a classified ads Web site into Google Maps. Users can view the map enhanced with information on what property is available for rent or sale in the area. Another example is Fishing Solutions <http://www.fishingsolutions.com.au> that uses Google Maps and information from anglers to help users find fish.

It is easier and quicker to create a mashup than to code an application from scratch in a traditional way. This capability is one of Web2.0’s most important and valuable features.

**Podcast**

A Podcast is defined in Wikipedia as follows: "A Podcast is a multimedia file that is distributed by subscription (paid or unpaid) over the Internet using syndication feeds, for playback on mobile devices and personal computers". In the beginning, multimedia files were the same audio files (.mp3). Nowadays video files are also distributed via Podcasts.

Similar to Weblogs, the technology behind them is rather simple. Their popularity is due to the ease with which users can produce their own Podcast, with the help of RSS, the growing bandwidth of Internet connections (which make bigger downloads possible), and the availability of Internet-enabled mobile devices. Some examples that describe the use of Podcast in Education can be found in [16][17].

It seems that this technology is gaining in popularity. The need to include social and collaborative elements in e-learning academic environments has already been identified [14].

**3 The EEQM Model**

The EEQM process includes a three axis approach to quality based on software quality factors, pedagogical requirements and Web 2.0 elements. The Software Quality axis aims to evaluate all Web elements related to the quality of the e-learning system. The evaluation process is performed internally by the organization (e.g. academic institution). Internal in this paper refers to the sub-process of the first axis. It is defined as the evaluation performed by the organization’s experts (managerial and technical) and
views the Web application as a business tool and as a system. The management defines and verifies whether the Web application meets the organizational/business goals and the specific targets of the organization. The technical personnel evaluate the application characteristics based on general software quality standards and on certain Web quality factors as described in the previous section.

The Pedagogical axis aims to evaluate all the requirements necessary for an e-learning system. The evaluation process is performed internally and externally. External in this paper refers to the evaluation performed by a group of users. A group of learners evaluate the Web system based on their own requirements and expectations.

The Web 2.0 axis aims to evaluate social activities and, in particular, collaboration among learners. The evaluation process is performed only externally by a group of learners. Figure 1 shows the EEQM framework that was devised illustrating the three axes of the evaluation model and the boundaries between the internal and external environments.

A sample users’ questionnaire is developed for evaluating the quality of the Web domain under examination by measuring user satisfaction and the importance of each criterion (see appendix A). The questionnaire was designed with a focus on quality factors, as described earlier. For example, certain questions are related to usability, others to functionality, and so on. In addition, Web quality factors as proposed by researchers [18][19][20].

Table 1 is used for evaluating the various components that affect the quality of the Web application. The users fill out the questionnaire and the organization technical experts define the important quality factors from Table 1.

Both types of evaluators, technical experts and users, indicate their satisfaction by assigning numerical values ranging from 1 to 5 as follows:

1. Very Unsatisfactory.
2. Unsatisfactory.
3. Good.
4. Very Good.
5. Excellent.

The evaluators also assign numerical weights ranging from 1 to 5 to each requirement so as to distinguish the different levels of importance. The higher the number, the more important the requirement:

1. Least Important.
3. Important.
4. Very Important.
5. Most Important.

Since the weights are only estimated, only single digits of precision are used. The same also applies to satisfaction levels.

The EEQM model involves the combination of three methods for estimating the quality of e-learning systems in order to look at the Web application from different perspectives:

Method 1. Frequency: The frequency of users’ answers based on their satisfaction and importance for each question of the questionnaire will indicate users’ frequent impression of the Web application’s quality. Specifically, we can observe the number of users that are satisfied or not with a given feature together with their importance preference on a scale from 1 to 5. If the frequency of users’ satisfaction value is equal or more than 4 then we may consider that the Web application is ‘Satisfactory’, otherwise certain features may need further enhancement.

Method 2. Median: Median indicates the middle value of a set of data. The median is less sensitive to extreme scores than the mean and this makes it a better measure for highly skewed distributions. The median is the \( M \)th value in a distribution \( L \) of length \( n \), where \( n \) is the number of values in a sorted set of data. We calculate the median for each question by using the formula:

\[
M = \frac{n}{2}, n_{even}
\]

\[
\frac{(n+1)}{2}, n_{odd}
\]

The calculation of median can provide more details about the quality of a specific feature/component. Features with values below the median need improving and priority should be given to those where the difference between the importance median and the satisfaction median is more than 2 points (e.g. Importance is 4 and Satisfaction is 2).

Method 3. Total Quality: To find the total quality of the Web application we first estimate the median for each feature/component for the internal evaluation (axis 1) as described earlier, for each question asked for the external evaluation (axis 3) and for each feature/component and question for both the internal and external evaluation (axis 2). Then to calculate the total quality for the external evaluation we use the weight to give us the number of times the satisfaction median will appear in our sorted set of data. The median of this set will be the total quality for the external evaluation. For example, if the satisfaction median for question 1 is 3 and importance (weight) is 4 then number 3 will appear 4 times in our set. For the internal evaluation we also add weights of importance for each expert. For example, a Webmaster may have a weight of 5 while the system manager has a weight of 4, a decision that has to be agreed between them. Therefore, for the internal evaluation we multiply each expert’s weight with the attribute’s weight and then we follow the external evaluation procedure. Based on our example, if the satisfaction of expert 1 for attribute 1 of the internal evaluation is 4 and its importance is 3, then the number 4 will appear in our set 15 times \( (W_1 * W_2) \). The same process will be performed for expert 2, and again for axis 2.

4 7-Step Evaluation Process

The EEQM model follows a step-by-step process to evaluate any given e-learning application. In particular the
<table>
<thead>
<tr>
<th>QUALITY FACTORS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1. FUNCTIONALITY</strong></td>
</tr>
<tr>
<td>1.1 Accuracy</td>
</tr>
<tr>
<td>1.2 Suitability</td>
</tr>
<tr>
<td>1.2.1 Products taxonomy</td>
</tr>
<tr>
<td>1.2.2 Services taxonomy suitability</td>
</tr>
<tr>
<td>1.3 Interoperability</td>
</tr>
<tr>
<td>1.3.1 Browser independence</td>
</tr>
<tr>
<td>1.3.2 Hardware independence</td>
</tr>
<tr>
<td>1.4 Compliance</td>
</tr>
<tr>
<td>1.5 Security</td>
</tr>
<tr>
<td>1.5.1 Authentication</td>
</tr>
<tr>
<td>1.5.2 Payment transaction</td>
</tr>
<tr>
<td>1.5.3 Vulnerability</td>
</tr>
<tr>
<td>1.5.4 Confidentiality</td>
</tr>
<tr>
<td>1.6 Searching</td>
</tr>
<tr>
<td>1.6.1 Simple searching</td>
</tr>
<tr>
<td>1.6.2 Advanced searching</td>
</tr>
<tr>
<td>1.6.3 Sitemap</td>
</tr>
<tr>
<td>1.6.4 Search Feedback</td>
</tr>
<tr>
<td>1.7 Navigability</td>
</tr>
<tr>
<td>1.7.1 Links</td>
</tr>
<tr>
<td>1.7.2 Scrolling</td>
</tr>
<tr>
<td>1.7.3 Navigational prediction</td>
</tr>
<tr>
<td>1.7.4 Indicator of path</td>
</tr>
<tr>
<td>1.7.5 Shortcuts availability</td>
</tr>
<tr>
<td>1.7.6 Navigation error feedback</td>
</tr>
<tr>
<td>1.7.7 Mobile devices accessibility</td>
</tr>
<tr>
<td>1.7.8 Alternative access paths</td>
</tr>
<tr>
<td>1.8 Multilingualism</td>
</tr>
<tr>
<td>1.9 Printing mechanisms</td>
</tr>
<tr>
<td><strong>EEQM model involves the following steps:</strong></td>
</tr>
<tr>
<td>1. <strong>Select the internal evaluators:</strong> Identify the expert evaluators for the internal evaluation and the users for the external evaluation. The internal evaluators may include the project manager, the Webmaster, Web designers, a Quality Assurance Manager (QAM), etc. Usually, this is a small team of three to five persons. One of them leads the evaluation process. Preferably the QAM or in his/her absence the Webmaster. These evaluators are gathered together to perform the evaluation.</td>
</tr>
<tr>
<td>2. <strong>Select the external evaluators:</strong> Decide on the number of users and the way the external evaluation process will proceed. Users may evaluate the Web application either by filling in an online questionnaire or by taking part in an</td>
</tr>
</tbody>
</table>
evaluation focus group.

3. Establish the set of features to be evaluated and select evaluation criteria: The EEQM model can be utilized for the evaluation of the e-learning application as a complete system or for a specific group of features of the application depending on the organization’s requirements. Select from Table 1 those attributes from each quality factor that are more appropriate according to the type of Web application under study. Note that Table 1 may be enhanced by additional criteria if there are any other special characteristics of the application under evaluation.

4. Develop user questionnaire: Prepare a questionnaire for the users to evaluate the Web application focusing on revealing critical issues affecting the quality of the system. For the purpose of evaluation we have used a questionnaire as provided by Mavromoustakos and Andreou [18].

5. Perform the evaluation: This stage consists of three parts. First, the evaluation is performed by the organization based on its requirements and following the aforementioned ISO quality characteristics and Web standards. The second part includes the evaluation by both the organization experts and a group of users to determine the quality of the system in reference to pedagogical requirements. Finally, the third part includes the evaluation by a group of users to determine user satisfaction, by indicating the quality of the e-learning application from their own point of view. Evaluators complete their questionnaires by selecting their level of satisfaction and by differentiating the high-importance criteria component should remain intact, the model can be adapted to new technological changes, as well as to new user requirements. Therefore, the model may continually be updated in terms of quality attributes in order to accommodate these changes.

6 Conclusions

In this paper we have addressed the issue of quality in e-learning applications and have proposed a three-axis quality evaluation model for Web2.0 e-Learning systems called E-learning Evaluation Quality Model (EEQM). These are: a) software quality factors, b) pedagogical requirements and c) Web2.0 elements.

The software quality factors of functionality, efficiency, usability, reliability and maintainability and their sub-components are studied in the evaluation of an e-learning application, enhanced by importance-based criteria.

The key pedagogical factors for enhancing the learning process includes the identification of learners’ needs, structuring of the pedagogical material, enhancement of the e-learning environment, encouragement of student participation, establishment of collaborative mechanisms among students, use of the relevant tools and components in support of any specific solution, and the right mix of the various learning processes implemented.

The EEQM model involves the combination of three methods for estimating the quality of e-learning systems: a) Frequency, b) Median and c) Total Quality. Finally, the model is represented in a practical stepwise process.

Further research steps will include the enhancement of the evaluation criteria in response to new user requirements and technologies emerging during the research. Finally, it would be interesting to look into the degree to which the EEQM model is applicable to mobile e-learning applications and to identify those parts of our approach that need to be modified and/or enhanced so as to facilitate the evaluation of this special group of software systems.

References


