Towards the Quality for Web 2.0 Applications

Luis Olsina[1], Roberto Sassano[2], Luisa Mich[2]

[1]GIDIS_Web, Engineering School, Universidad Nacional de La Pampa, Argentina
[2]Department of Computer and Management Science, University of Trento, Italy
olsinal@ing.unlpam.edu.ar; roberto.sassano@gmail.com;
luisa.mich@unitn.it

Abstract. One of the steps to evaluate quality is to define first quality requirements, for instance, by means of models. In order to specify quality models for Web 2.0 applications we propose an integrated approach which considers quality dimensions for contents and functionalities. In particular, we discuss how to model the different quality views reusing and extending the ISO 9126-1 quality models taking into account not only the software characteristics but also the own features of Web applications. To illustrate the added *Content quality* characteristic to quality models, excerpts of two evaluations are shown. The resulting models contribute towards a multidimensional integrated approach to evaluate Web applications of different eras –both Web 1.0 and 2.0 ones-, and lifecycle stages.

Keywords: Web 1.0/2.0, Quality Model, ISO 9126-1, Content Characteristic

1. Introduction

As the Web usage continues to grow, users expect Web applications (WebApps) to be more mature with regard to the functions and contents delivered for their intended goals and tasks at hand. Moreover, users expect these functions and contents be relevant, accurate, suitable, usable, secure, and ultimately with perceived quality in use. Despite the major breakthroughs in Web development methods and technologies there are some Web Engineering branches still in their infancy. Modelling quality requirements at different lifecycle stages is one of them. In this paper we propose an integrated approach to specify quality requirements for content quality to WebApps.

WebApps were conceived as content-oriented artefacts from the very beginning. Few years later, websites started to provide not only contents but software-like functionalities and services as well. Since then, WebApps have at a fast pace emerged for many different sectors as e-commerce, e-learning, e-government, and so forth. After that era named Web 1.0, a recent era that considers a set of strategies and technologies focusing on social networking, collaboration, integration, personalization, etc. is emerging –currently called Web 2.0 [8, 12]. We argue that WebApps will continue being centered on functionalities, services and contents independently of the new technologies and collaboration strategies. However, as aforementioned a challenging issue still is how to specify and assess the quality and the quality in use of WebApps at different lifecycle stages since their intrinsic nature will continue being both content and function oriented.

In the present work, by reusing and extending the ISO 9126-1 [4] quality representation, we discuss how to model internal, external quality, and quality in use views taking into account the previous concerns. Particularly, we thoroughly discuss

the modeling of the *Content quality* characteristic for evaluating the quality of information –so critical for the whole WebApp eras.

Later on, as a motivating example for evaluating the content quality dimension [9], we will gauge that Cuspide (www.cuspide.com) should plan changes in the shopping cart basic information, for instance, in the *Line item information completeness* and *Product description appropriateness* attributes (see Fig. 1). Particularly, the *Line item information completeness* should have at least the author's name besides the title of the item, since users for lack of information can not distinguish between two or more items with the same title (e.g., "Ingenieria de Software") when they want to add a specific item quantity or compare prices.

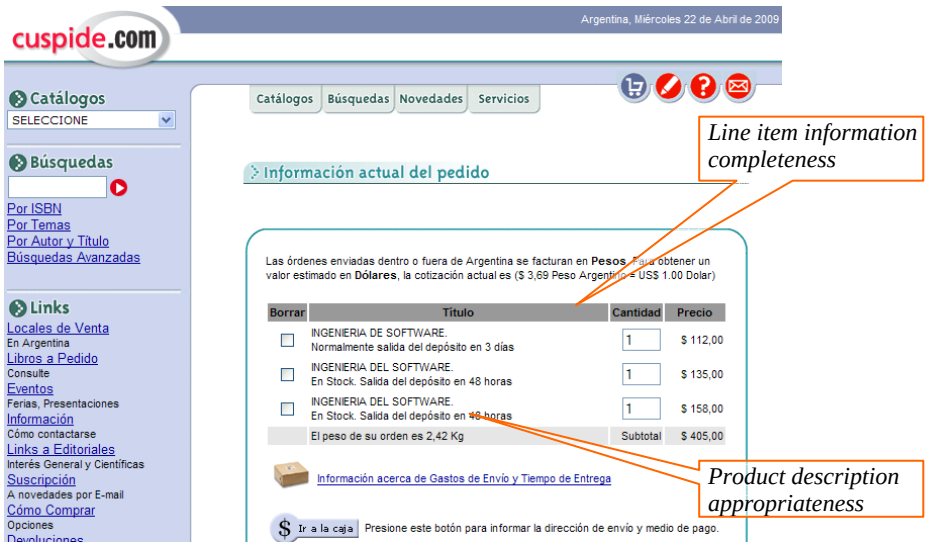

Figure 1. Screenshot to the Cuspide's shopping cart with two content attributes highlighted.

The rest of this paper is structured as follows. In Section 2, we give an overview of the Web eras as well as the unique features of WebApps compared to traditional software applications. A review of the ISO software quality models follows in Section 3. In Section 4, we discuss what is missing in these models with regard to the Web features; we also illustrate the proposed extension to ISO quality models in order to include the content quality dimension. Then, Section 5 illustrates an excerpt of a quantitative evaluation for a shopping cart of a Web 1.0 app regarding the content suitability characteristic; in addition, it describes the application of a qualitative evaluation by comparing three touristic apps related to Web 1.0 and 2.0 eras. Finally, in Section 6, we analyze our proposal in the light of related work, and then, in Section 7, we draw concluding remarks.

2. Characterizing WebApps

The first WebApps can be grouped in the Web 1.0 era, and they can be categorized into static and dynamic; most recent WebApps can be grouped in the so-called Web

2.0 era as per O'Reilly [12]. These allow people collaborate, share and edit information online in seemingly new ways of interaction. Other applications, which can be referred as the mobile Web era, offer additional features such as personalization and context-aware capabilities and services; and the semantic Web era –also named Web 3.0- where applications offer the automatic processing of information meaningfully. For Web 2.0 apps we can highlight some features, namely:

- *User generated content:* checking the rating of the most popular sites, we can figure out that currently, after 'google.com' and 'yahoo.com', one of the most visited is 'youtube.com'. Maybe the best example to explain how big has become the Web 2.0 phenomenon and what user generated content means.
- *User active involvement*: the active participation of users is one of the most important features that changed the way users have to interact with WebApps. Now users' role can be defined as '*prosumer*' since s/he is content producer and consumer at the same time. WebApps like blogs are significant examples.
- *Sharing information*: let's think about social network services, where people share interests and activities. Examples of these applications are 'facebook.com', 'myspace.com' and 'orkut.com'.
- *Endless beta condition*: considering the above three features it is easy to understand that Web 2.0 apps are mostly dynamic and under ongoing changes. Wikipedia is for instance continually subject to editing by users so there is no a 'final version' of it.

In addition, Murugesan [8] says these new sites "*offer smart user interfaces and built-in facilities for users to generate and edit content presented on the Web and thereby enrich the content base. Besides leveraging the users' potential in generating content, these applications provide facilities to keep the content under the user's own categories (tagging feature) and access it easily (Web feed tool)*". He also remarks "*Web 2.0 is gradually becoming recognized as an important collection of technologies, business strategies, and social trends. As a result of these developments the Web is changing fast from a static, one-way publishing medium to a highly interactive, dynamic application platform for fielding new kinds of applications*".

On the other hand, as said in Olsina *et al* [9], WebApps –besides distinctive features of Web development processes-, have their own features distinct from traditional software applications, namely:

- WebApps will continue being driven by contents. Most WebApps, besides the increasing support to functionalities –seen since the dynamic Web 1.0 era- will continue aiming at showing and delivering multimedia information. This info orientation is a basic feature stemming from the early, static Web 1.0 era.
- WebApps are interactive, user-centered applications, where the user interface plays a central role; thus, they will continue being focused on the look and feel. Web interfaces ought to be easy to use, understand, and operate because millions of users with different profiles and capabilities interact with them daily; in addition, WebApps currently have to cope with a variety of display devices and screen sizes.
- Internationalization and accessibility of Web contents for users with various disabilities are real and challenging issues, independently of eras.
- Security is a central issue in data- transaction-oriented WebApps. Likewise, performance is also critical for many WebApps, although both are also critical features for traditional applications.

- The medium where WebApps are hosted and delivered is generally more unpredictable than the medium where traditional software applications run. For instance, unpredictability in bandwidth maintenance or in server availability, can affect the users' perceived quality.
- Contents privacy and intellectual property rights of materials are current issues too. They involve ethic, cultural, and legal aspects as well.

Most of the above features make a WebApp a particular artifact. However, like any software application, it also involves source and executable code, persistent structured data, architectural design and so on. Ultimately, many of the above features will influence the way quality requirements are modeled. We need to deal not only with usability, functionality, efficiency, reliability and maintainability, as in traditional software products but also with content quality, i.e. with content accuracy, suitability, accessibility, and legal compliance.

3. Overview of the ISO 9126-1 Quality Models

Quality is not a simple and atomic concept, but rather a multidimensional and relative one. Common practice assesses quality by means of the quantification of lower abstraction concepts, such as attributes of entities. An attribute can be defined as a measurable property of an entity category. Therefore, quality –and its sub-dimensions, called characteristics and sub-characteristics in the ISO 9126-1 standard-, is an abstract relationship between attributes of an entity and a specific information need, with regard to its purpose, context, and user's viewpoint [10]. On account of such multidimensionality [7], a quality model, which specifies the relationships between characteristics, sub-characteristics and associated attributes, is usually necessary. Further, an instantiated quality model can in the end be calculated and evaluated in order to determine the level of satisfaction achieved.

The ISO 9126-1 standard (and the ongoing SQuaRE project) distinguishes among three different approaches to software product quality, viz. internal quality, external quality, and quality in use. These views of quality can be summarized as follows [9]:

- *Internal Quality*, which is specified by a quality model based on a set of six characteristics – functionality, usability, reliability, efficiency, maintainability and portability– and a set of sub-characteristics per each characteristic. It can be measured and evaluated by static attributes of documents such as specification of requirements, or designs; pieces of source code; and so forth.
- *External Quality*, which is specified by a quality model with the same characteristics as the previous model. It can be measured and evaluated by dynamic properties of the running code in a computer system, i.e. when the module or full application is executed in a computer or network simulating as closely as possible the actual environment.
- *Quality in Use*, which is specified by a quality model based on a set of four characteristics –effectiveness, productivity, safety and satisfaction. It can be measured and evaluated by the extent to which a software or WebApp meets specific user needs in an actual, specific context of use.

The same quality models have been maintained both to internal and external views (see Fig. 2). For instance, functionality characteristic is defined as "*the capability of*

the software product to provide functions which meet stated and implied needs when the software is used under specified conditions" [4]. Functionality has in turn five sub-characteristics, i.e. accuracy, suitability, security, interoperability and compliance. Functionality –from the nonfunctional requirements point of view- is concerned with what the software does to fulfill the user needs (software is defined as a set of programs with the associated data and documentation). Considering for example the accuracy and security definitions both function and data attributes can be associated in order to assess them. This is also valid for WebApps where programs and persistent, structured data (and its effects) are there as well. Note that in the information quality literature data and information quality are treated very often as synonymous terms but we make a clear difference as we discuss in the next section.

Besides, usability characteristic is defined as "*the capability of the software product to be understood, learned, used and attractive to the user, when used under specified conditions*" [4]. Usability is subdivided in turn into five sub-characteristics, i.e. understandability, learnability, operability, attractiveness, and compliance. Usability and its sub-characteristics apply also to specifying internal and external quality requirements for WebApps.

Lastly, the core aim in designing a WebApp is to provide users with degrees of excellence or quality in use by interacting with the application and by performing its tasks comfortably. Regarding the spirit of the ISO 9126-1 standard, quality in use is the end user's view of the quality of a running system containing software, and it is measured and evaluated in terms of the result of using it rather than by properties of the product itself. Ultimately, taking into account meaningful software or WebApp attributes for internal quality are a prerequisite to achieve the required external behavior, and considering meaningful software attributes to external behavior are a prerequisite to achieve quality in use.

4. Extending to the ISO Quality Models for WebApps

The assumption of our proposal is that the ISO 9126-1 software quality models introduced in the previous section are also applicable to a great extent to intermediate and final lifecycle WebApps. To justify this we are going to extend the discussion made in [9], reinforcing the same line of argumentation.

Like any software production line, the Web lifecycle involves different stages of its products, whether in early phases as inception and development, or in late phases as deployment, operation, and evolution. To assure the quality of products, we can plan to do it by evaluating and controlling the quality from intermediate products to final products. Thus, to the general question, if we can apply to WebApps the same ISO internal and external quality, and quality in use models, the natural answer is *yes.* Nevertheless, to the more specific question whether we can use the same six prescribed quality characteristics (and their sub-characteristics) for internal and external quality requirements, and the four characteristics for quality in use requirements, our answer is *yes* for the latter, but some other considerations might be taken into account for the former. As highlighted in Section 2, the very nature of WebApps is a mixture of contents and functions. Therefore we argue that the set of six characteristics, i.e. functionality, usability, reliability, efficiency, maintainability

and portability, are not well suited (or they were not intended) to specify requirements for information quality.

At this point, we would like to introduce a slight difference in meaning between *data* and *information* terms. A piece of data is raw material; even though it has some degree of information. Data come from attribute measurements, facts, formula calculations, etc. Structured data sets are often represented in databases. On the other hand, information has an added value over data. That is, information is the meaningful interpretation of data for a given purpose and context. Usually, a traditional software program is a mixture of functions and data. Very often a webpage is content oriented, i.e. is intended to deliver information (usually unstructured semantically). For example, this article can be hyperlinked and posted as content Web pages. Also a webpage component, e.g. a shopping cart, can edit an item quantity and recalculate prices (applying a function over data).

Therefore, to follow the thread of our argument, the central issue is how we can specify the content quality for WebApps from the internal and external quality viewpoints. We need to deal not only with usability, functionality, efficiency, reliability, and maintainability but also with the content quality characteristic, which in turn can be subdivided into content accuracy, suitability, accessibility, and legal compliance sub-characteristics, as shown in Fig. 2.

Figure 2. ISO characteristics for internal and external quality along with our extension to the *Content quality* feature –which could also be named *Infoquality*.

As a consequence, we propose to include the *Content quality* characteristic (or *Infoquality*, for short), and its sub-characteristics in the internal and external quality model of the ISO standard. A point worth mentioning is that in the spirit of the ISO 9126-1 standard is stated "*evaluating product quality in practice requires characteristics beyond the set at hand*".

On the other hand, the quality in use definition may be rephrased as "*the capability of the software or WebApp product to enable specified users to achieve specified goals with effectiveness, productivity, safety and satisfaction in specified context of use*". Note that effectiveness, productivity, safety, and satisfaction are influenced not only by the usability, functionality, reliability, efficiency, and content quality of a WebApp, but also by two resource components of the context of use. The context of use depends on both the infrastructure (i.e. the computer, network, or even the physical working medium) and the user-oriented goals, i.e. the supported WebApp

tasks and the properties of the user type such as level of IT training, expertise, age, and cultural issues as well. (See for example the quality in use case study for an e-learning WebApp [3] where user tasks were designed not only to deal with services and functions but with contents as well). Next, we analyze the proposed extension.

4.1 The Information Quality Characteristic

As aforementioned information has added value over data, and hereafter we consider Web information as Web content, which can be textual or other media. Hence, we define *Content quality* as "*the capability of a Web product to deliver information which meets stated and implied needs when used under specified conditions*". Taking into account previous contributions made in the area of information quality –as we will discuss in the related work section-, we have primarily identified four major sub-characteristics for the content quality characteristic, which can help to measure and evaluate information quality requirements for WebApps. This initial proposal was made in [9]. So we contribute here by redefining them (see table 1) and by extending and defining the sub-sub-characteristics (see Fig. 3).

The content sub-characteristics are: first, *content accuracy*, which addresses the very intrinsic nature of the information quality; second, *content suitability*, which addresses the contextual nature of the information quality; it emphasizes the importance of conveying the appropriate information for user-oriented goals and tasks; in other words, it highlights the quality requirement that content must be considered within the context of use and the intended audience; third, *content accessibility*, which emphasizes the importance of technical and representational aspects in order to make Web contents more accessible for users with various disabilities as regarded in [16]; and lastly, *legal compliance*, as defined in table 1. In Fig. 3, we define in turn sub-characteristics for both the content accuracy and content suitability dimensions. Some of them could be just treated either as measurable attributes or as sub-dimensions to which attributes should be further associated accordingly (as we exemplify in Section 5). Even if we have identified some attributes to content legal compliance, we are not addressing this aspect in this work.

Table 1. Definition of *Content quality* sub-characteristics for specifying information quality

Sub-characteristic	Definition
Content Accuracy	The capability of a Web product to deliver information that is correct, credible and current.
Content Suitability	The capability of a Web product to deliver information with the right coverage, added value, and consistency, considering the specified user tasks and goals.
Content Accessibility	The capability of a Web product to deliver information that is accessible for all users (with or without disabilities) taking into account both technical and representational aspects.
Content Legal Compliance	The capability of a Web product to adhere to standards, conventions, and legal norms related to content as well as to intellectual property rights.

1. **Content Accuracy**
   - 1.1. ***Correctness***, *the extent to which information is reliable in the sense of being free of errors.*
   - 1.2. ***Believability*** (synonym: Credibility), *the extent to which the information is reputable, objective, and verifiable*.
     - 1.2.1. ***Authority*** (synonym: Reputability), *the extent to which the source of the information is trustworthy*.
     - 1.2.2. ***Objectivity***, *the extent to which the content (i.e., information or facts) is unbiased and impartial*.
     - 1.2.3. ***Verifiability*** (synonym: Traceability), *the extent to which the owner and/or author of the content can be verified*.
   - 1.3. ***Currency*** (synonym: Up-to-dateness), *the extent to which the information can be identified as updated*.
2. **Content Suitability**
   - 2.1. ***Value-added***, *the extent to which the information can be novel, beneficial, and contribute to react to a given user for the task at hand*.
     - 2.1.1. ***Novelty*** *(*synonym*:* Freshness*), the extent to which the information is fresh and contributes to make new decisions for an intended user goal.*
     - 2.1.2. ***Beneficialness****, the extent to which the information is advantageous and contributes to make new decisions for an intended user goal*.
     - 2.1.3. ***Reactiveness***, *the extent to which the information is compelling and contributes to react for an intended user goal.*
   - 2.2. ***Coverage***, *the extent to which the information is appropriate, complete but also concise for the task at hand to a given user*.
     - 2.2.1. ***Appropriateness***, *the extent to which the information coverage fits to an intended user goal*.
     - 2.2.2. ***Completeness***, *the extent to which the information coverage is the sufficient amount of information to an intended user goal*.
     - 2.2.3. ***Conciseness***, *the extent to which the information coverage is compactly represented without being overwhelming*.
   - 2.3. ***Consistency***, *the extent to which the content is consistent to the website's piece of information or page with respect to the intended user goal*.

Figure 3. Definition of the proposed *Content Accuracy* and *Suitability* sub-dimensions for specifying internal and external information quality requirements.

In addition to the above content sub-characteristics, others to information architecture and organization can be addressed. Many of these sub-characteristics, such as global understandability, learnability, and also operability and attractiveness, can be related to the *usability* characteristic. Besides, other particular attributes to search and navigation functionalities can be specified in the functionality sub-characteristics; for example, are the basic and advanced search suitable for the end user? Or, are they tolerant of misspelled words and accurate in retrieving documents? In the same way, we can represent link and page maturity attributes, or attributes to deficiencies due to browsers' compatibility into the reliability sub-characteristics.

Finally, from the quality in use perspective, we have proposed to use the same ISO model. However, for the satisfaction characteristic, specific (questionnaire) items for evaluating quality of content should be included.

5. Evaluating the Content Quality Characteristics

The model for evaluating the content quality dimension shown above is a general schema adaptable to all kinds of WebApps. It means that the integrated model can be used with both the Web 1.0 and the Web 2.0 apps and different domain types. It is for this reason that when you decide to apply it to a given WebApp, it must be instantiated on the basis of some more specific parameters, e.g. measurable attributes and metrics, questionnaire items, and decision criteria (e.g. using indicators), considering also the specific domain, information need, and user viewpoint.

Next, for space reasons we show excerpts of two studies. One of them was a quantitative evaluation for a shopping cart of a Web 1.0 app (performed twice, in 2004 [9] and in 2007 [11]), where some content suitability sub-characteristics and attributes have intervened. The other one was an exploratory study where three tourist WebApps related to both Web 1.0 and 2.0 eras have intervened.

Table 2 shows two attributes coded 1.1.1.1 and 1.1.1.2 (highlighted in Fig. 1), for evaluating the coverage level of the basic information to the Cuspide shopping cart. Note that *Coverage* is a sub-characteristic of *Content Suitability* as specified in Fig. 3. The purpose of the evaluation was to understand the external quality level of *Content quality* and *Usability* dimensions for the shopping cart entity, and then recommend improvements in case low satisfaction of these requirements were reached. The improvements were made applying Web Model Refactoring (refer to [11] for details).

Table 2. External Content quality requirements for a shopping cart; EI = Elementary Indicator; P/GI = Partial/Global Indicator. (Excerpt extracted from [11])

Characteristics and Attributes (*in italic*)	**Measure**	**EI value**	**P/GI value**
1 Content Quality			**63.05%**
1.1 Content **Suitability**			63.05%
1.1.1 Basic Information **Coverage**			50%
1.1.1.1 Line item information completeness	***1***	*50%*	
1.1.1.2 Product description appropriateness	***1***	*50%*	
1.1.2 Coverage of other Contextual Information			76.89%
1.1.2.1 ……		…	

First, for each attribute of the requirement tree we selected a metric to quantify it. For example, the *Line item information completeness* attribute is defined as "*the extent to which the line item information coverage is the sufficient amount of information to an intended user goal and task*"; thus, final users expect to have suitable information as title, author, price, quantity, added on date, and availability in order to accomplish the task effectively. So we designed a direct metric named *Degree of completeness to the line item information* whose scale specifies three categories considering an ordinal scale type, namely: (0) *Incomplete,* it has less information than category 1, or no information at all; (1) *Partially complete*, it only has title, price, quantity, and sometimes availability fields; and (2) *Complete*, it has the whole required information.

After measurement it resulted 1, as shown in table 2. However, the measure by itself does not represent the elementary requirement's satisfaction level, so we have to design an elementary indicator. Note that an elementary indicator interprets the level

of satisfaction of an elementary nonfunctional requirement (i.e. an attribute). Therefore, a new scale transformation and decision criteria (in terms of acceptability ranges) should be defined. In our study, we used three agreed acceptability ranges in a percentage scale: a value within 40-70 (a marginal –yellow- range) indicates a need for improvement actions; a value within 0-40 (an unsatisfactory –red- range) means changes must take place with high priority; a score within 70-100 indicates a satisfactory level –green- for the analyzed attribute. Table 2 shows an elementary indicator value of 50% for the 1.1.1.1 attribute taking into account that a measure value of 1 mapped to 50%. In conclusion, this attribute would satisfy suitability totally if the measure is 2.

In our case, as mentioned in the introduction, for lack of information users are not able to distinguish between two or more items with the same starting title, when they want to add a specific item quantity or compare prices. The reader can refer to [11] for more details on measurement, elementary/global evaluation, and recommendations.

On the other hand, regarding the qualitative study, three WebApps to the tourism sector, namely www.opodo.com, www.tripadvisor.com, and www.wayn.com were evaluated by experts on a comparative basis. This sector, these WebApps, and the evaluation aim have been chosen for the following reasons: (i) the comparison is a method used to seize on correspondences and specific differences between the entities observed; (ii) the evaluation as aforementioned was conducted by experts in the tourism sector. The eTourism Research Group[1] collaboration to which one of the authors belongs allows us to take advantage of its expertise and knowledge gained over the years in the tourism research field; and (iii) the selected WebApps represent to some extent the development of applications from Web 1.0 to Web 2.0 eras. We expected the model be useful with both Web 1.0 and Web 2.0 applications. Opodo has all the features to be classified as Web 1.0, Tripadvisor offers some Web 2.0 features, and finally Wayn can be entirely classified as a Web 2.0 application.

The evaluation considered the whole sub-characteristics/attributes of table 1 as well as some attributes regarding content accessibility and content legal compliance. For the former were included sub-characteristics as perceivable, operable, understandable, robust, which are the principles of WCAG 2.0 [16]; for the latter: security, privacy, copyright, and sector law. In the study, per each sub-characteristic/attribute we formulated a question. Hence, we design a questionnaire that allowed us to evaluate the three WebApps qualitatively by means of an inspection method intervening two experts. Though we have non-conclusive evidence from this study, some initial comments and observations can be drawn about content features that distinguish Web 1.0 from Web 2.0 apps.

The first relevant issue to underline is that in the process of content production in Opodo (Web 1.0 app) a top-down logic is pursued, it means that only content providers supply information to users. On the other hand, in TripAdvisor and Wayn (Web 2.0 apps) the process of content production becomes bottom-up; i.e. mainly users upload and update information. Moreover in Wayn the content is submitted to a

[1] Research team working at the Department of Computer and Management Sciences, Faculty of Economics at University of Trento; group site available at http://www.economia.unitn.it/eTourism/home.asp

social control mechanism, since with a Web 2.0 app (e.g. blog, wiki, social network) a user can edit, add or comment content of other users. Besides, in the initial observations it is quite evident how "non-structured" information can be considered more accurate and suitable in 'tripadvisor.com' with respect to 'opodo.co.uk'. Note that when we talk about non-structured information we refer for example to hotel review, location comment, etc. Conversely, for structured information –e.g. flight timetables, holiday price lists, among other, it can be considered more accurate and suitable in 'opodo.co.uk' (see more details in [13]).

Lastly, in general terms we argue that the WebApp's content quality does not depend on the kind of applications –whether Web 1.0 or Web 2.0; however, some kind of contents and services are more appropriate for Web 1.0 apps, while others for Web 2.0. Moreover, we can state the content sub-characteristics we have modeled for evaluation purposes can be applied to all WebApps, independently from which era they belong. However, some specific measurable attributes associated to content sub-characteristics could be considered for Web 2.0 and follow-on applications, as we discuss in a future work.

6. Discussion

The model presented in this paper as an extension of the ISO 9126-1 quality models, has been elaborated taken into account related researches about dimensions of data and information quality. We describe here briefly the main information quality models, often called frameworks in the literature. A broader reviews for WebApp quality is given for instance in [6]. It is worth mentioning that the difference in meaning between data and information –as remarked in Section 4- has often been neglected in these quality frameworks. Moreover, very often –and in some cases explicitly- these terms are used interchangeably.

One of the first studies intended to categorize data quality dimensions was made by Strong *et al* [14]. They aimed at considering the dimensions of data quality for three user roles, namely, data consumer, data custodian, and data producer. According to them, high quality data is data that fit for use by intended users. They developed a framework made up by four categories –intrinsic, contextual, representational, and accessibility- including 16 dimensions of data quality. Specifically, the intrinsic category indicates that information has its own quality *per se*. It contains four dimensions: accuracy, objectivity, believability, and reputation. The accessibility category states that information must be easily accessible but secure; it includes: accessibility, and security dimensions. The third category is contextual data quality, which indicates that information should be provided in time and in appropriate amounts; it includes: relevancy, value-added, timeliness, completeness, and amount of data. The last category is representational data quality, which focuses on format of data/information and its meaning; it includes: interpretability, ease of understanding, concise representation, and consistent representation. As a matter of fact, this quality framework was initially developed for traditional information systems. Nevertheless, this model has been used for WebApps too. For instance, Katerattanakul *et al* [5] reuse the four categories and the characteristics including free-of-error webpage content, workable and relevant hyperlinks, and the navigational tools provided for

accessibility. Caro *et al* [2] have in a recent study reused the Strong *et al* framework for modelling data quality of Web portals from the data consumer viewpoint. All these data quality frameworks neither consider different lifecycle WebApps stages/eras and therefore different quality models as we propose, nor make a clear distinction between data and information quality either.

A different slightly way to model and evaluate the information quality for a WebApp –both at page and site level- is proposed by Alexander *et al* [1]. They take into account six dimensions (criteria) such as authority, accuracy, objectivity, currency, orientation, and navigation. Authors include a checklist for a step-by-step quality evaluation to some kind of websites, namely to the advocacy, business, informational, personal, news, and entertainment sectors. They evaluate information rather than data without considering different information quality models at different WebApp lifecycle stages.

The first published study about extending the ISO 9126 model has been made by Zeist *et al* [17]. In a nutshell the extended model consists of adding some sub-characteristics for each characteristic, with the aim of specifying data/information quality. Unfortunately, this study is quite limited because at that moment the ISO standard did not consider the internal, external, and quality in use views –these were included just in the 2001 revised standard. Finally, there exists an ongoing SQuaRE project[2] which proposes harmonizing many ISO standards related to quality models, measurement and evaluation processes. According to Vaníček [15] "*these standards have not a unified terminology and do not fully reflect the current state of art in software engineering*". In his contribution he proposes a data quality model regarding the three ISO views, but these models are just for data (data as a new entity) separated of the quality models for software functions. As the author was aware "*the main problem concerning the development of new SQuaRE series of standard and also concerning the data quality standard is the enormous volume of standardisation documents … If we extend the number and span of standards, nobody will use them*". Instead of the SQuaRE approach, our aim is modeling nonfunctional requirements for WebApps' functions and contents, taking into account the three integrated quality models and Web lifecycle views.

7. Concluding Remarks

Most WebApps besides the increasing support to functionalities and services will continue aiming at showing and delivering content. This basic feature stemming from the early Web 1.0 applications is currently empowered by the Web 2.0 and follow-on applications. Web 2.0 applications rely strongly on actual users sharing, collaborating and performing content tasks in real contexts of use. So evaluating the quality of WebApps is still or even more challenging.

In the present work, we have proposed how to specify quality requirements for WebApps' functions, services and contents employing a multidimensional and integrated approach. By reusing and extending the ISO 9126-1 quality models'

[2] Note that as part of the project the ISO 25012:2008 standard on data quality model was recently issued

characteristics, we have discussed the need of modeling and adding the content quality characteristic for evaluating the quality of information. Specifically, we have argued that the internal and external quality models with the set of six characteristics, i.e. functionality, usability, reliability, efficiency, maintainability and portability, and their sub-characteristics respectively, are not sufficient to specify WebApps' information quality. As a consequence, we have proposed to include in both models the *Content quality* (*Infoquality*) characteristic and its sub-characteristics. Besides, from the quality in use perspective, we have proposed to use the same ISO model.

References

1. Alexander J. E. & Tate M. A. *Web Wisdom: How to evaluate and create information quality on the web*. NJ: Erlbaum. 1999.
2. Caro A., Calero C., Mendes E., & Piattini M. A Probabilistic Approach to Web Portal's Data Quality Evaluation. *Proc. IEEE 6th Int'l Conference on the Quality of Information and Communications Technology*, Portugal. 2007, pp.143-153.
3. Covella G. & Olsina L. Assessing Quality in Use in a Consistent Way. *In ACM proceedings, Int'l Congress on Web Engineering, (ICWE06)*, SF, USA, 2006, pp. 1-8.
4. ISO/IEC 9126-1. Software Engineering—Software Product Quality:—Part 1: Quality Model, 2001.
5. Katerattanakul P. & Siau K. Measuring information quality of web sites: Development of an instrument. *Proc. 20th Int'l Conference on Information Systems*. US. 1999, pp.279–285
6. Knight S. & Burn J. Developing a Framework for Assessing Information Quality on the World Wide Web. *Information Science Journal.* V.8, 2005.
7. Mich L., Franch M., & Gaio L. Evaluating and Designing the Quality of Web Sites. *IEEE Multimedia*, (10)1: 34-43, 2003.
8. Murugesan S. Web Application Development: Challenges and the Role of Web Engineering. Ch. 2 in Springer *Web Engineering: Modeling and Implementing Web Applications*. Rossi, Pastor, Schwabe & Olsina (Eds). 2007, pp. 7-32.
9. Olsina, L.; Covella, G.; & Rossi, G. Web Quality. Ch. 4 in Springer book: *Web Engineering*, Mendes & Mosley (Eds.). 2005, pp. 109–142.
10. Olsina L; Papa F. & Molina H. How to Measure and Evaluate Web Applications in a Consistent Way. Ch. 13 in Springer book: *Web Engineering: Modeling and Implementing Web Applications,* Rossi, Pastor, Schwabe & Olsina (Eds). 2007, pp. 385–420.
11. Olsina L.; Rossi, G; Garrido, A.; Distante, D.; Canfora, G.; Web Applications Refactoring and Evaluation: A Quality-Oriented Improvement Approach, In: *Journal of Web Engineering, Rinton Press*, US, (7) 4: 258-280, 2008.
12. O'Reilly T. What is Web 2.0? Design Patterns and Business Models for the Next Generation of Software, 2005 (online at http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html).
13. Sassano R. Content quality model for Web 2.0 Websites, Master thesis (in Italian). University of Trento, Italy, 2008.
14. Strong D., Lee Y. & Wang R. Data Quality in context. *Communications of the ACM*, 1997, 40(5): 103-110.
15. Vaníček J. Software and Data Quality. Proc. conference Agricultural Perspectives XIV. Czech University of Agriculture in Prague, 2005, Sep. 20–21.
16. WWW Consortium, Web Content Accessibility Guidelines 1.0., 1999, (available online at http://www.w3.org/TR/WAI-WEBCONTENT/).
17. Zeist R.H.J. & Hendriks P.R.H. Specifying software quality with the extended ISO model. *Software Quality Management IV – Improving Quality*, BCS, 1996, pp.145-160.