Quality, Quality in Use, Actual Usability and User Experience as Key Drivers for Web Application Evaluation

Philip Lew[1], Luis Olsina[2], and Li Zhang[1]

[1] School of Computer Science and Engineering, Beihang University, China
[2] GIDIS, Web Engineering School at Universidad Nacional de La Pampa, Argentina
philiplew@gmail.com, olsinal@ing.unlpam.edu.ar, lily@buaa.edu.cn

Abstract. Due to the increasing interest in Web quality, usability and user experience, quality models and frameworks have become a prominent research area as a first step in evaluating them. The ISO 25010/25012 standards were recently issued which specify and evaluate software and data quality requirements. In this work we propose extending the ISO 25010 standard to incorporate new characteristics and concepts into a flexible modeling framework. Particularly, we focus on including information quality, and learnability in use characteristics, and actual usability and user experience concepts into the modeling framework. The resulting models and framework contribute towards a flexible, integrated approach to evaluate Web applications. The operability and particularly the learnability of a real Web application are evaluated using the framework.

Keywords: Learnability, quality, usability in use, learnability in use, usability, information quality, user experience.

1 Introduction

Web applications (WebApps), a combination of information content, functionality and services are fast becoming the most predominant form of software implementation and delivery today. Due to these evolutions, *usability, information quality, quality in use* and, in the end, *actual user experience* have taken on increased importance. With the latest ISO 25010 quality standard [8], and other recent work by researchers in the field of *quality in use, usability* and *user experience* such as Bevan [2] and Hassenzahl [7], it is still not totally clear where characteristics such as *information quality*, *learnability in use*, *actual usability* and *user experience* fit in regarding quality modeling.

The recent ISO 25010 standard on quality models, updates and brings previous standards together while delineating three views of quality viz *internal*, *external quality* and *quality in use*. Some researchers including the ISO 9241-11 have suggested that *quality in use* is defined similarly to *usability*. Of particular interest in ISO 25010 is the standard's new breakdown of *quality in use* and *usability* which provides us insight into our framework for this paper. Bevan examined ISO 25010

B. Benatallah et al. (Eds.): ICWE 2010, LNCS 6189, pp. 218–232, 2010.

from the viewpoint of usability and user experience (UX) and drew some interesting relationships regarding usability as performance in use, and satisfaction as it relates to user experience. Hassenzahl´s work in classifying user experience in two categories, hedonic and pragmatic is also useful when examining usability from the *do*, pragmatic viewpoint and the *be*, or hedonic satisfaction viewpoint.

Meanwhile, ISO 25012 [9], a recent standard on data quality, considers data as an entity by itself, and as a separate model. However, extending the number and span of standards runs the risk that nobody will use them [19]. There is a need to integrate *information quality* as part of the overall quality of an application, particularly for WebApps, rather than as a separate entity. Olsina et al [14] made an initiative in this direction as discussed in Section 3.

Rather than rely on separate standards, we propose to augment the ISO 25010 standard to include *information quality* as a characteristic of internal/external quality because this is a critical characteristic of WebApps. Furthermore, the 25010 standard has categorized *learnability* as an internal/external quality subcharacteristic under the *operability* characteristic. We further propose to include *learnability in use* as a characteristic of *usability in use* to account for the learning process and the importance of context of use during learning. Lastly, we propose an integrated means to evaluate the characteristics of *usability in use* as they relate to *user experience*. All these issues will be combined into an integrated framework which embraces the possibility of instantiating different models. This framework and its models, i.e., *internal/external Quality*, *Quality in use*, *actual Usability* and *User experience* (2Q2U, for short) are as compliant as possible with the recent ISO standards, considering other well-known contributions as well.

Ultimately, the specific contributions of this research are: (a) An extension to ISO 25010 internal/external quality model to include *information quality* as a characteristic; (b) An extension to ISO 25010 *quality in use* model to include *learnability in use* as a subcharacteristic of *usability in use;* (c) An integrated and flexible framework for modeling quality requirements, and particularly *quality in use* to relate the concepts of *actual usability* and *user experience;* and (d) An instantiation, for illustration purposes, of the ISO 25010 *external quality* model with subcharacteristics and measurable attributes to evaluate *operability* for WebApps, which can influence *learnability in use*.

Following this introduction, Section 2 reviews recent related work and delineates opportunities for improvements. In Section 3, our proposal of extending the ISO 25010 standard in order to incorporate new characteristics and concepts into a flexible framework called 2Q2U is discussed. In Section 4, we discuss the usefulness of the proposed framework, in the context of evaluating operability and some of its subcharacteristics, e.g. learnability for a WebApp. Section 5 draws our main conclusions and outlines future work.

2 Related Work and Motivation

User experience, a relatively new term, combined with *usability*, *information quality*, and *quality in use*, have all recently come to the research forefront especially for WebApps due to the shift in emphasis to satisfying the end user. Yet based on the

current standards and literature reviewed, it can be difficult to understand their relationships and we often observe a lack of consensus in meaning. This section examines the related work with an eye for improvement opportunities.

In the recent ISO 25010 standard, the concept of quality has been broadened from 6 characteristics (ISO 9126-1 [11]) to 8 as shown in Fig. 1.a.

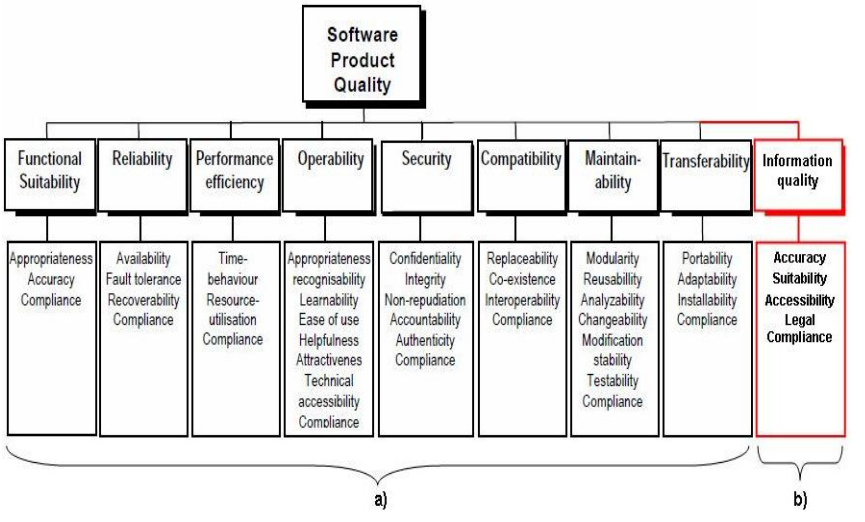

Fig. 1. a) ISO 25010 Internal/External Quality Model; b) Adding Information Quality

The standard is split into 2 quality models. The first is a software internal/external quality model. In comparison with ISO 9126-1, the previous characteristic of *usability* has been renamed as *operability* with broader meaning. For instance some sub-characteristics such as *learnability* among others have remained while new ones such as *technical accessibility compliance* and *helpfulness* were added. *Security* has been added as a separate characteristic, rather than as a subcharacteristic of *functionality* in the former, while other names have changed slightly to enhance descriptiveness. The second model in ISO 25010 depicted in Fig. 2.a refers to *quality in use* and includes previous ISO 9126-1 *quality in use* characteristics while adding others. Note that the *effectiveness* and *satisfaction* characteristics from ISO 9126-1 were imported into this newer standard as subcharacteristics of *usability in use*, while the *productivity* has been renamed as *efficiency in use*. In addition, *flexibility in use* has been added to accommodate different usage contexts including *accessibility in use*. It is worth mentioning that the suffix 'in use' was added to two characteristics and many subcharacteristics.

As can be seen from these figures, *learnability*, an internal/external sub-characteristic of *usability/operability* in ISO 9126-1/25010 has not been moved to *usability in use*. In addition, ISO 25010 does not include information or content quality as a characteristic of either model. Recent ISO's intentions are for data to be addressed by a complementary standard, namely ISO 25012.

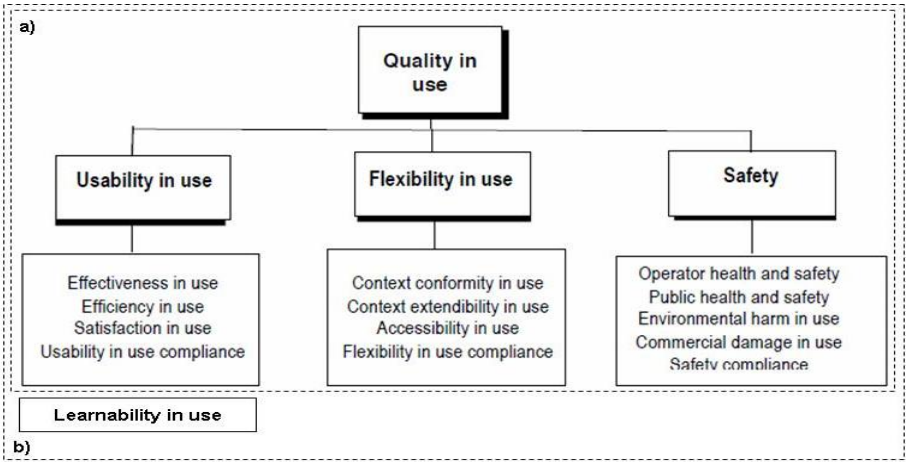

Fig. 2. a) ISO 25010 Quality in Use Model; b) Adding Learnability in Use into Usability in Use

ISO 25012 is a general data quality model intended to be used to establish data quality requirements, and plan and perform data quality evaluations. This standard is intended to be used in conjunction with ISO 25010, but by going to such length to define quality of data, it loses emphasis in using data as information and as a component of a WebApp rather than just data as an entity itself. Information quality as a characteristic was researched by [14] whereby they proposed extending ISO 9126-1 with content quality containing four sub-characteristics including *content accuracy*, *content suitability*, *content accessibility*, and *legal compliance*. We adapt their contribution to the ISO 25010 internal/external quality model as discussed in Section 3.

Learnability has been removed from the usability characteristic of ISO 9126-1 and moved to the *operability* characteristic of ISO 25010 where it is defined as "degree to which the software product enables users to learn its application" borrowing the dialogue principles from ISO 9241-110 [10] regarding suitability for learnability.

For WebApps, users are expected to learn intuitively with no user manual. However, *learnability* in ISO 25010 is solely a product quality, which does not incorporate evaluation of the process of learning and does not model *learnability* in different real contexts of use such as the domain of the system, and its target users. Research made by [17] exemplifies the need to examine learning from different user viewpoints, as learning observed for new users is not necessarily related to continued learning. Furthermore many researchers have determined *learnability* to be linked directly with *usability* as summarized by Abran [1]. Bevan [3] also noted *learnability in use* as part of *usability in use*.

On the other hand, the term UX is becoming more important as evidenced by the definitions by various researchers. To understand the term requires breaking down the word 'user experience' and examining first what experience means.

Experience is a general concept which refers both immediately-perceived events and the wisdom gained in interpretation of events. In the context of UX, it is a sequence of events over time for a user's interaction with the software product. [7]

notes that the time dimension could be either momentary or accumulated and changing over time. The 'moment' perspective does not exclude the accumulation or summary perspective, rather, it adds to it like a continuum.

Examining the 'user' part of the *user experience* concept, [7] characterizes a user's goals into pragmatic, do goals and hedonic, be goals and assumes the interactive product quality is perceived in two dimensions, pragmatic and hedonic. Pragmatic quality refers to the product's perceived ability to support the achievement of tasks such as paying a bill and focuses on the product's utility and usability in completing tasks that are the 'do-goals' of the user. Hedonic quality refers to the product's perceived ability to support the user's achievement of 'be-goals', such as being happy, or satisfied with a focus on self. He also argues that the fulfillment of be-goals is the driver of experience and that lack of usability or inability to complete do-goals may prevent achieving be-goals, but do-goals are not the end goal of the user. Rather the real goal of the user is "to fulfill be-goals such as being autonomous, competent, related to others, stimulated, and popular through technology use." He also states that pragmatic quality enables achieving hedonic quality be-goals and has no value by itself, but only through enabling accomplishment of be-goals. In summary, user experience comes from fulfilling be-goals in the time dimension, at the moment, and in summation over time.

Given that, it's easy to see why UX has become such a buzz word regarding WebApps. Websites and the interactive game industry have combined to change our expectations of software/WebApps. We not only expect them to work and help us get our task done, but also expect them to be pleasant to use and provide satisfaction. Yet, a common standard definition for user experience is still not available. According to Stewart [18], the next revision of ISO 9241-210 defines UX as 'all aspects of the user's experience when interacting with the product, service, environment or facility' and that "it is a consequence of the presentation, functionality, system performance, interactive behavior, and assistive capabilities of the interactive system. It includes all aspects of usability and desirability of a product, system or service from the user's perspective". Stewart states that *usability* depicts a narrower concept than *user experience* and simply focuses on systems being easy to use. In his words, "Easy to use is not enough" as exemplified through the IPod's market dominance through more than just ease of use.

Satisfaction in use, as noted by Bevan, correlates to Hassenzahl's hedonic goals whereas *usability in use* and its do-goals are related to a user's pragmatic goals. In summary, *usability in use*, *satisfaction in use*, and *user experience*, need clearer relationships in order to model and evaluate them. After a reviewing the related work, we found possible opportunities for improvement, as we discuss in the next Section.

3 2Q2U Models and Framework: Proposal and Discussion

The aim of the proposed models and framework is twofold: first, adding characteristics to extend the ISO 25010 standard and; second, add two new concepts, *actual usability* and *actual UX*, to which characteristics and subcharacteristics can be related and new models created in a flexible way. Regarding the extension of the ISO 25010 standard, as shown in Figures 1.b and 2.b, we added the following:

- *Information quality,* defined as the degree to which the software/WebApp provides accurate, suitable, accessible and legally compliant information. This new characteristic becomes part of the internal/external quality model.
- *Learnability in use* defined as the degree to which specified users can learn efficiently and effectively while achieving specified goals in a specified context of use. This new subcharacteristic becomes part of the *usability in use* characteristic in the *quality in use* model.

The ISO 9126-1 standard stated "evaluating product quality in practice requires characteristics beyond the set at hand". Hence, the revised ISO 25010 increased the total number of characteristics from 6 to 8. As software applications continue to change, we suggest including the above mentioned characteristics accordingly.

Second, regarding the addition of two new concepts, *actual usability* and *UX*, we hereby provide the following definitions:

- *Actual Usability*: the degree to which specified users can achieve specified goals with effectiveness in use, efficiency in use, learnability in use, and accessibility in use in a specified context of use. Note: *Actual usability* is measured and evaluated in a real operational environment where real users perform actual specified tasks.
- *Actual User Experience*: the degree to which specified users can achieve actual usability, safety, and satisfaction in use in a specified context of use. Note: *Actual UX* is evaluated not only by measures and indicators of user performance –as in *actual usability*-, but also by means of satisfaction instruments.

Subsections 3.1 through 3.3 further develop the aspects above and then relate them together in our flexible framework in section 3.4, which models characteristics of usability in use to bring together the concepts of *usability* and *user experience*.

3.1 Adding Information Quality

In our proposal to extend ISO 25010, we assume that the software quality models, definitions, and concepts in the standard were intended for application to software products as a whole and therefore are also applicable to WebApps, a type of software application. Like any software product, building WebApps involves different development stages, from inception and development to operation and maintenance. Thus we should be able to use the same ISO *internal* and *external quality* and *quality in use* models for WebApps with the same eight prescribed quality characteristics (and their subcharacteristics) for internal and external quality requirements, and the three characteristics for *quality in use* requirements, but some other considerations might be taken into account.

As highlighted elsewhere [5] the very nature of WebApps is a mixture of contents and functions. Therefore we argue that the eight internal/external quality characteristics (see Figure 1.a) are not well suited, nor were they intended to specify requirements for *information quality*.

We intentionally use the term 'information' to differentiate from 'data'. Data comes from attribute measurements, facts, formula calculations, etc. and are often organized and represented in databases. On the other hand, information is the meaningful interpretation of data for a given purpose and context. Given that a WebApp is very often content oriented and intended mainly to deliver information,

the central issue is the ability to specify the *information quality* for WebApps from the internal and external quality viewpoints. This viewpoint is also supported by ISO 9241-110 which relates characteristics of presented information to its dialogue principles. Therefore, we propose augmenting the internal/external quality model with the *information quality* characteristic, with *content accuracy*, *content suitability*, *content accessibility*, and *content legal compliance* as subcharacteristics, shown in Fig. 1b. Definition for each characteristic is done by Olsina *et al* in [14].

It may be argued that *information quality* should be added as a *quality in use* characteristic. However, it can be evaluated as an internal/external quality characteristic by measuring its attributes. In addition, when designing tasks for *quality in use*, for example for evaluating *efficiency* and *effectiveness in use*, content and functions are embedded in the task design itself rather than as attributes of the software/WebApp. *Satisfaction in use* questionnaires can also address *information quality* with specific questions related to its sub-characteristics [4].

3.2 Adding Learnability in Use

As mentioned earlier, *learnability* solely as a product quality does not incorporate evaluation of the learning process or different usage contexts. More specifically, we examine learning context to strengthen our reasoning to include *learnability in use* as a *usability in use* subcharacteristic, from the user group type and time viewpoints.

Regarding the former, the learning objectives and therefore behavior of different user groups have an impact on the learning process as novice users behave differently than expert users [17]. Ease of learning depends on the user group type and the task being attempted. As an extreme, a quality requirement characteristic to minimize the necessary learning time, or to make the learning time equal to zero depends entirely on who the user is, and what tasks they are trying to do. As another example of how user group types behave differently based on their background and task at hand, requirements for users who are trying to re-learn a task can be difficult to model as a product characteristic. Grossman *et al* [6] also noted several other user group types including: i) Level of experience with computers; ii) Level of experience with interface; iii) Level of related domain knowledge; iv) Experience with similar software. Therefore, the dimension of user group types and its influence in *learnability* is of paramount importance.

Regarding the time aspect, learning from different user viewpoints such as initial learning and continued learning as researched by [17] are not necessarily correlated. So, measuring the learning of users must be done in the time dimension as the time delta between initial learning and continued learning has an influence on the *learnability* of the software in a real context. Many *learnability* measures focus on initial *learnability*. As Nielsen states: “One simply picks some users who have not used the system before and measures the time it takes them to reach a specified level of proficiency in using it” [13]. However, continued *learnability* requires assessing performance over time using a constant user group.

Some may argue that *effectiveness in use* and *efficiency in use* either combined or solely, can constitute *learnability in use*. However, software that is easy to learn is not always efficient to use and vice versa. A WebApp´s design evaluated highly as part of the *learnability* external quality characteristic, may lead to less efficient procedures.

Learnability therefore depends on the domain of the software, its target users and tasks at hand. Hence it cannot solely be determined by inspection, as an internal or external quality characteristic. Bevan also related *learnability in use* as a sub-characteristic of *usability in use*, as discussed in Section 2. Given the aforementioned reasons, and recalling our definition as "the degree to which specified users can learn efficiently and effectively while achieving specified goals in a specified context of use", we argue that *learnability in use* should be added to the *usability in use* characteristic as a subcharacteristic. Moreover, if we agree on including *learnability in use* into the model, we can combine, for instance, the following measurable attributes [12]:

- *Assisted learning time:* Time for user from a specific user group type to learn to complete a specific task in a specified time, plus the amount of instructional guidance time if needed.
- *Relative user learning efficiency:* The efficiency of an ordinary user as compared to an expert measuring the time required to reach a predefined threshold ratio.

Other attributes such as *task time deviation*, *error rate over time*, *learning efficiency* and so forth can be supplemented and used in conjunction with external attribute learnability measurements in order to draw relationships between *quality in use* and *learnability*.

3.3 Modeling Actual Usability and User Experience

As mentioned in Section 2, these two concepts are mainly derivatives through the [2, 7] works. In Bevan´s work relating and explaining factors contributing to system *usability* and *UX* he defines 4 characteristics of *usability in use*: i) *Effectiveness and productivity in use* ii) *Learnability in use* iii) *Accessibility in use* and iv) *Safety in use*. He also matches *usability* to performance in use measures equivalent to those characteristics related to the pragmatic 'to-do' goals of the end user.

Measures of *UX* are noted by Bevan as being composed of *satisfaction in use* as equivalent to achieving pragmatic and hedonic goals, with its subcharacteristics as specified by ISO 25010 including *pleasure, likability, comfort,* and *trust*. Hassenzahl further elaborates on hedonic goals as: "fulfilling the human needs for autonomy, competency, stimulation (self-oriented), relatedness, and popularity (others-oriented) through interacting with the product or service". He further states that pragmatic quality facilitates satisfaction of be-goals. That is, be-goals are not dependent on, but facilitated by do-goals; i.e. a user could be satisfied even if do-goals are not satisfied. For example, if a user cannot buy a product online efficiently (slowly with mistakes), but ends up buying what they like, then they may achieve their be-goals and be very satisfied.

Thus, achieving *UX* is influenced through satisfaction of both *usability in use* (pragmatic goals) and *satisfaction in use* (hedonic goals). Ultimately, the *actual usability* and *UX* definitions given in the introduction of Section 3 are based on this rationale. The above concepts and relationships are shown in Figure 3 and further explained in the following section in the context of our proposed framework.

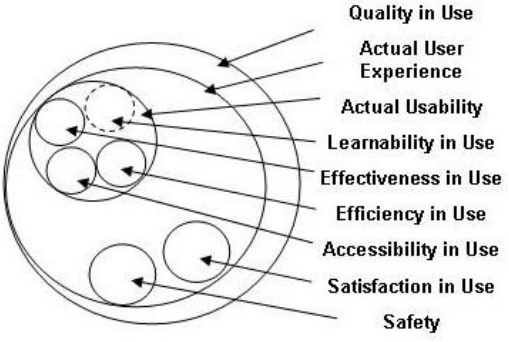

Fig. 3. Relationships of Quality in Use, Actual Usability, and Actual User Experience

3.4 Specifying and Using the Proposed Framework

Now, given the aforementioned definitions and relationships, our 2Q2U framework for modeling nonfunctional requirements for *internal/external Quality*, *Quality in use*, *actual Usability* and *User experience* can be specified and generalized to flexibly meet the evaluator needs to represent these calculable concepts. Figure 4 shows a basic Venn diagram to represent this viewpoint.

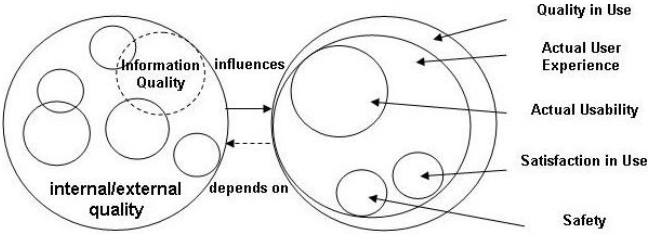

Fig. 4. Proposed 2Q2U modeling framework

The particular 2Q2U models can in turn be instantiated relying on the INCAMI (*Information Need, Concept model, Attribute, Metric and Indicator*) nonfunctional requirement specification component. As detailed in by Olsina et al [16], the *InformationNeed* allows evaluators to establish the evaluation *purpose*, user *viewpoint*, *Entity, focus*, and *CalculableConcept* such as external quality, actual usability, user experience, among others. As such, our proposed framework can be used flexibly to generate *ConceptModels* by choosing *CalculableConcepts* (characteristics in the ISO terminology) depending on the specific *InformationNeed*. We can then combine *Attributes* to characteristics or subcharacteristics to fully instantiate the selected model, resulting in a requirements tree, for further measurement and evaluation purposes.

2Q2U is in line with the intention of the ISO 25010 standard where tailoring the model is encouraged for relative importance of characteristics depending on the situation at hand: *"It is not practically possible to measure all internal and external*

…. Similarly it is not usually practical to measure quality in use for all possible user-task scenarios. The relative importance of quality characteristics will depend on the product and application domain. So the model should be tailored before use…"

Regarding the above, we just examine –for space reasons- the actual UX model (subset of quality in use) including our proposed learnability in use subcharacteristic in the light of our *actual usability* concept. To this aim, we modeled four characteristics, namely, *efficiency in use*, *effectiveness in use*, *learnability in use* and *accessibility in use* as shown in Fig. 5.

As per our discussion that *actual usability* is related to satisfying the do goals of the end user while completing real specified tasks, we include *satisfaction in use* as part of *actual UX* rather than in *actual usability*. Fig. 5 depicts our model composition as part of the framework with *actual usability* and *actual user experience* as defined and discussed above.

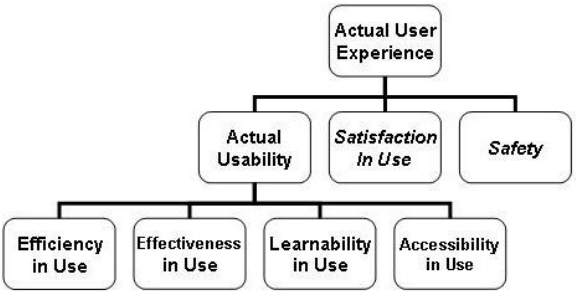

Fig. 5. Model composition representing Actual Usability and Actual User Experience

Note from Fig. 5 that *actual usability* is not a prerequisite for *actual UX*, but rather has influence as one out of three characteristics. Both concepts involve the temporal component, not just UX [7] but also *actual usability*. Note that *safety* is a *quality in use* characteristic defined by ISO 25010 as "Acceptable levels of risk of harm to people, business, data, software, property or the environment in the intended contexts of use". *Satisfaction in use* and *safety* are in italic to denote their hedonic nature. *Safety* is depicted as a hedonic characteristic of *actual UX* because often it contributes to the user´s emotional needs for security and trust rather than a characteristic which satisfies a do-goal. Lastly, *accessibility in use*, defined in ISO 25010 as the "degree of usability in use for users with specified disabilities" is modeled as part of *actual usability*.

4 Example of Evaluating WebApp Operability

As the contributions mentioned in the Introduction Section, we developed a flexible modeling framework that uses to a great extent the latest ISO quality models, but also we have enlarged their models to include new characteristics and concepts, taking into account other well-known contributions in the discipline. As discussed above, our outcome is a set of new related models, assembled using the 2Q2U framework, which ties together well-known terms: quality, usability and user experience. Moreover the 2Q2U framework can be flexibly inserted and used with the so-called INCAMI

nonfunctional component. In turn, we can use INCAMI to design the measurement, evaluation, and analysis.

INCAMI and its methodology have been used in different case studies. Recently, we performed a practical case by evaluating *external quality* requirements for a shopping cart, followed by *Web Model Refactoring* [15] for improvements. As stated by ISO 25010, for the end user, *quality in use* results mainly from *functional suitability, reliability, operability* and *performance efficiency*. Thus, for illustration purposes, we have applied the same methodology, taking into account the ISO 25010 *operability* characteristic because subcharacteristics and attributes related to *operability* as well as *information quality* could affect *quality in use* and *usability in use* including not only *efficiency in use* and *effectiveness in use* but also *learnability in use*. This will be addressed in our future work.

In Table 1, we now examine some of the subcharacteristics of *operability*, namely, *learnability*, *ease of use*, and *helpfulness* to measure and evaluate external quality requirements for software/WebApps. We strive to maintain consistency and alignment with the ISO 25010 quality models, while recognizing that there are many possibilities for other dimensions and attributes. In addition, these subcharacteristics have been modeled for general evaluative purposes for use with all WebApps. Some attributes associated to subcharacteristics could be more applicable to some domains than others and therefore receive higher relative weighting.

Table 1. Definition of used operability characteristics, subcharacteristics and attributes

Operability Characteristics and *Attributes*	**Definition**
1 Learnability	Degree to which the software/WebApp enables users to learn its application.
1.1 Predictability	Degree to which the software enables users to predict its interactions, functionality or content. Note: By being able to predict the consequences of an action, users can operate the software with minimal unintended consequences and fewer errors.
1.1.1 Action determinability	Degree to which the software /WebApp enables the user to predict what his action will do. Note: For instance, the user can evaluate potential inputs showing the result before changes are applied.
1.1.2Predictive textual anchor information	Degree to which the textual link provides users meaningful anchors or contextual information in order to help predict the target destination.
1.2 Feedback Suitability	Degree to which mechanisms and information regarding the success, failure or awareness of actions is provided to users to help them interact with the application. Note: Users need to know what might happen given the options available. Feedback about system states relieves users from having to remember these states, thereby making learning easier.
1.2.1Task Progress feedback suitability	Degree to which users are made aware of what they are doing for a specific task, function, or process. Note: For example, display progress in current process with number of steps completed and how many remaining to complete the task, or please wait, system is processing.

Table 1. (*continued*)

1.2.2 Navigability feedback completeness	Degree to which users are made aware of past, current, and possible locations while performing a navigation-oriented task.
1.2.3 Entry form feedback awareness	Degree to which users are made aware of the correctness or incorrectness of data entries.
1.2.4 Error message appropriateness	Degree to which meaningful error messages are provided upon invalid operation so that users know what they did wrong, what information was missing, or what other options are available. Note: This also relieves users from learning error recovery methods.
2 Ease of Use	Degree to which the software/WebApp makes it easy for users to operate and control it.
2.1 Controllability	Degree to which users can initiate and control the direction and pace of the task until task completion.
2.1.1 Permanence of main controls	Degree to which main controls are consistently available for users in all appropriate screens.
2.1.2 Stability of main controls	Degree to which main controls are in the same location in all appropriate screens.
2.2 Error Tolerance	Degree to which if, despite input errors, the intended result may be achieved with either no, or minimal, corrective action by the user.
2.2.1 Invalid Action Forgiveness	Degree to which users are allowed to attempt invalid actions without negative consequences.
2.2.2 Error Recovery Support	Degree to which the software/WebApp provides support for error recovery. Note: For instance, cursor is automatically positioned at the location where correction is required.
3 Helpfulness	Degree to which the software product provides help that is easy to find, comprehensive and effective when users need assistance.
3.1 Help Suitability	Degree to which the software/WebApp provides appropriate help given the users, their experience and context of help when required.
3.1.1 Context-sensitive help availability	Degree to which the software/WebApp provides context sensitive help depending on the user profile and goal, and current interaction.
3.1.2 Help Appropriateness	Degree to which the software/WebApp provides traditional online help with a structure that is easily readable and searchable. Note: For example, a top down hierarchy with hyperlinks for more detail for easier reading, semantic search, and advanced search.

Table 1 shows the selected characteristics, subcharacteristics and attributes *(in italic).* The purpose of the evaluation was to understand the *external quality* level of the *operability* characteristic for filling new prescriptions (the evaluated *entity*) of a pharmacy WebApp. For confidentiality reasons, we do not disclose the company name and site, but are closely cooperating with the company to recommend improvements in case low satisfaction of these requirements were achieved. Table 2 shows an excerpt of the whole current evaluation.

Note in table 2 that for each attribute of the requirement tree has a metric to quantify it. For example, for *Error recovery support* attribute (coded 2.2.2), users should not have to search to find their mistake to correct it. So we designed a direct metric whose scale specifies four categories considering an ordinal scale type, namely: (0) *None,* does not support at all; (1) *Partially*, sometimes there is support but not always; (2) Complete support but only partial controllability, and (3) *Complete*, always support with complete controllability.

Table 2. External quality requirements for new prescription filling for Operability; EI = Elementary Indicator; P/GI = Partial/Global Indicator

Operability Characteristics and *Attributes* (External Quality)	Measure	EI value	PI/GI value
Global Quality Indicator			62.8%
1 Learnability			66.7%
1.1 Predictability			75.0%
1.1.1 Action determinability	1	50.0%	
1.1.2 Predictive textual anchor information	2	100.0%	
1.2 Feedback Suitability			58.3%
1.2.1 Task progress feedback suitability	2	66.7%	
1.2.2 Navigability feedback completeness	1	33.3%	
1.2.3 Entry form feedback awareness	1	33.3%	
1.2.4 Error message appropriateness	2	100.0%	
2 Ease of Use			75.0%
2.1 Controllability			50.0%
2.1.1 Permanence of main controls	1	50.0%	
2.1.2 Stability of main controls	1	50.0%	
2.2 Error Tolerance			75.0%
2.2.1 Invalid action forgiveness	1	50.0%	
2.2.2 Error recovery support	3	100.0%	
3 Helpfulness			46.7%
3.1 Help Suitability			46.7%
3.1.1 Context-sensitive help availability	1	33.3%	
3.1.2 Help appropriateness	3	60.0%	

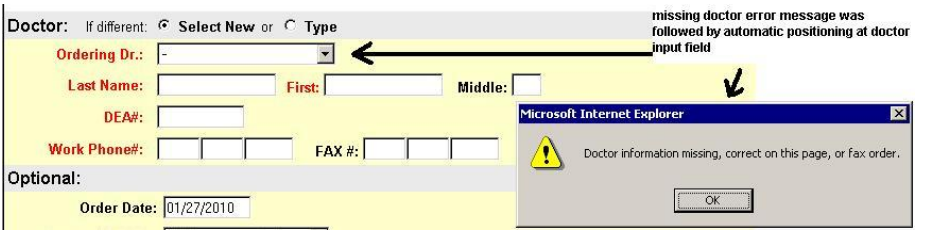

Fig. 6. Screenshot exhibiting error recovery support in a pharmacy prescription WebApp

As can be seen from table 2, its measurement resulted in 3 because it was clear to the user what the error was and the cursor was automatically placed at the error location throughout the task. However, the measure by itself does not have meaning so we must design an elementary indicator to interpret the level of satisfaction met. Therefore, a new scale transformation and decision criteria for acceptability ranges needs to be defined. In our study, we used three acceptability ranges in a percentage scale: a value within 40-70 (marginal –yellow) indicates a need for improvement actions; a value within 0-40 (unsatisfactory –red) 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 100% for the 2.2.2 attribute, but for instance resulted in 50% for the 1.1.1 attribute taking into account that a measure value of 1 mapped to 50%. This attribute would be totally suitable if the measure is 2. In the current state for the new prescription filling WebApp, users should not have to guess the results of their actions especially if they are infrequent users and improvement is needed.

Expanding the analysis as shown in table 2 can enable evaluators to understand the application's current external quality state and make recommendations for improvements. Note that the above model is just a possible instantiation of the external quality model represented in the 2Q2U framework (recall Figures 1 and 4). Going forward, we can also use the framework to model characteristics of *quality in use* and perform evaluation by extending the above study. By doing this with real users executing real tasks, we could derive relationships between *learnability* and *quality in use* characteristics such as *learnability in use* in order to determine how making improvements at the external quality level can affect *quality in use*. Regarding this, characteristics defined for *learnability in use* in section 3.2 can be measured with relationships drawn to the external quality *learnability* characteristics above.

5 Conclusions and Future Work

In this paper, we have developed a framework for modeling Quality, Quality in use, Usability and User experience (2Q2U) for Web applications, which in turn can be instantiated by using the INCAMI nonfunctional requirement specification component. In doing so, we have provided reasoning for and recommendations for adding 2 characteristics to extend the ISO 25010 standard, namely *information quality* and *learnability in use*. We have also characterized and described two new concepts, *actual usability* and *actual user experience* to bring to light their relationships while demonstrating using the framework.

To illustrate the applicability of the proposed approach, an example of external quality evaluation has been presented, in which the importance of taking into account *operability* was highlighted. 2Q2U offers not only an integrated framework for modeling requirements for quality, quality in use, actual usability and user experience, but also a consistent, and flexible way for representing calculable concepts (characteristics) and subconcepts which can then be used in conjunction with INCAMI for measurement and evaluation.

Another manuscript will thoroughly discuss the subcharacteristic, *learnability in use* covered in Section 3.2, including the further definition of attributes, metrics, and indicators for user group types performing specific tasks in real environments. Ongoing research is focused on further utilizing the 2Q2U framework when modeling and understanding the relationships among *internal/external quality, quality in use, actual usability* and *UX*. This concern has often been neglected in the literature, but may help evaluators to make sound design recommendations and ultimately better decision-making for improving the *user experience* as a whole.

Acknowledgments. Thanks to the support by Nat'l Basic Res. Prog. of China (973 project-2007CB310803) and Nat'l Natural Science Foundation of China (No. 60773155), and PAE-PICT 2188 project at UNLPam, Science and Technology Agency, Argentina.

References

1. Abran, A., Surya, W., Khelifi, A., Rilling, J., Seffah, A., Robert, F.: Consolidating the ISO Usability Models. In: 11th annual Int'l Software Quality Management Conference (2003)
2. Bevan, N.: Extending quality in use to provide a framework for usability measurement. In: Proc. of HCI Int'l 2009, San Diego, California, USA (2009)

3. Bevan, N.: Classifying and selecting UX and usability measures. In: Proc. of the 5th COST294-MAUSE Workshop on Meaningful Measures: Valid Useful User Experience Measurement (2008)
4. Covella, G., Olsina, L.: Assessing Quality in Use in a Consistent Way. In: ACM proceedings, Int'l Congress on Web Engineering (ICWE '06), SF, USA, pp. 1–8 (2006)
5. Ginige, A., Murugesan, S.: Web Engineering: An Introduction. IEEE Multimedia 8(1), 14–18 (2001)
6. Grossman, T., Fitzmaurice, G., Attar, R.: A survey of software learnability: metrics, methodologies and guidelines. In: Proceedings of the 27th Int'l conference on Human factors in computing systems, pp. 649–658 (2009)
7. Hassenzahl, M.: User experience (UX): towards an experiential perspective on product quality, IHM. In: Proc. of the 20th Int'l Conference of the Assoc. Francophone d'Interaction Homme-Machine, vol. 339, pp. 11–15 (2008)
8. ISO/IEC CD 25010 Software engineering – Software product Quality Requirements and Evaluation (SQuaRE) – Quality model and guide (2009)
9. ISO/IEC 25012 Software engineering – Software product Quality Requirements and Evaluation (SQuaRE) – Data quality model (2008)
10. ISO 9241-110, Ergonomics of human-system interaction, Part 110: Dialogue principles (2006)
11. ISO/IEC 9126-1. Software Engineering— Software Product Quality—Part 1: Quality Model (2001)
12. Lew, P., Zhang, L., Wang, S.: Model and Measurement for Web Application Usability from an End User Perspective. In: Proc. of QAW 2009: 1st Quality Assessment in Web Workshop held at Int'l Congress on Web Engineering (ICWE '09), CEUR Workshop Proceedings, San Sebastian, Spain, vol. 561 (2009) ISSN 1613-0073, http://ceur-ws.org
13. Nielsen, J., Levy, J.: Measuring usability: preference vs. performance. Comm. of the ACM 37(4), 66–75 (1994)
14. Olsina, L., Sassano, R., Mich, L.: Towards the Quality of Web 2.0 Applications. In: Proc. of 8th Int'l Workshop on Web-oriented Software Technology (IWWOST 2009) held at Int'l Congress on Web Engineering (ICWE09), San Sebastian, Spain, vol. 493, pp. 3–15, CEUR (ceur-ws.org) (2009) ISSN 1613-0073
15. Olsina, L., Rossi, G., Garrido, A., Distante, D., Canfora, G.: Web Applications Refactoring and Evaluation: A Quality-Oriented Improvement Approach. Journal of Web Engineering 4(7), 258–280 (2008)
16. Olsina, L., Papa, F., Molina, H.: How to Measure and Evaluate Web Applications in a Consistent Way. In: Rossi, Pastor, Schwabe, Olsina (eds.) Springer book: Web Engineering: Modeling and Implementing Web Applications, ch. 13, pp. 385–420 (2007)
17. Santos, P.J., Badre, A.N.: Discount learnability evaluation. Graphics, Visualization & Usability Center, Georgia Institute of Technology (1995), ftp://ftp.gvu.gatech.edu/pub/gvu/tr/1995/95-30.pdf (accessed by 20/12/2009)
18. Stewart, T.: Usability or user experience - what's the difference? System concepts (2008), http://www.usabilitynews.com/news/article4636.asp (accessed by 20/12/2009)
19. Vaníček, J.: Software and Data Quality. In: Proc. Conference of Agricultural Perspectives XIV, Czech University of Agriculture in Prague, vol. 52 (3), pp. 138–146 (2005)