Relating User Experience with MobileApp Quality Evaluation and Design

Philip Lew[1] and Luis Olsina[2]

[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

Abstract. Designing quality into web applications (WebApps) and evaluating WebApp quality and usability have been the subject of abundant research. However, both the design and evaluation of traditional WebApps cannot account for the particular features and usage contexts of mobile applications (MobileApps). MobileApps have several characteristics that pose challenges in their design and evaluation regarding current quality models and their included characteristics and sub-characteristics. For instance, the operability of a user interface has a much different and greater influence when evaluating MobileApp usability and user experience due to the context of the user. Characteristics such as multi-touch gestures, button size, and widget usage have a magnified impact on task completion rates. We propose in this paper utilizing our previously developed ISO 25010-based quality models and framework so-called 2Q2U (*Quality*, *Quality in use*, *actual Usability* and *User experience*) for MobileApps. Specific MobileApp task screens and attributes are illustrated in order to show our evaluation approach applicability.

Keywords: MobileApp, Quality models, User Experience, Usability, QinU.

1 Introduction

WebApps, a combination of information, integrated functionalities and services have become the most predominant form of software delivery today with users and businesses choosing to rent or use software rather than buy it. This has led to increased and focused attention in software quality models that facilitate understanding, evaluating, and especially improving their quality.

With respect to software quality models, ISO 25010 [7] outlines a flexible model with product/system quality –also known as internal and external quality (EQ)-, and system-in-use quality –also referred to as quality in use (QinU). Product quality consists of those characteristics that can be evaluated in early development stages, for instance, design documents, code quality, etc., while system quality consists of those characteristics and attributes that can be evaluated in late stages, with the application in execution state. On the other hand, system-in-use quality consists of characteristics as evaluated by an end user when actually executing application tasks in a real context. An example would be a nurse or a doctor entering patient record and diagnosis information into an electronic health records system. A doctor even while

Q.Z. Sheng and J. Kjeldskov (Eds.): ICWE 2013 Workshops, LNCS 8295, pp. 253–268, 2013.

doing the same task, may have different error and completion rates than a nurse for whatever reason. Doctors may also take longer and have less efficiency in completing tasks simply because they don't do as many. In addition, system QinU is heavily dependent on the context of the task and user. For instance, a user in a dim warehouse doing inventory will have a different viewpoint than a doctor in a well-lit hospital.

ISO 25010 also delineates a relationship between the two quality views whereby system quality 'influences' system-in-use quality and system-in-use quality 'depends' on system quality. We recently have developed 2Q2U version 2.0 [12], which ties together all of these quality concepts by relating system quality characteristics and attributes with QinU and user experience (UX). Using 2Q2U, evaluators can select the quality characteristics to evaluate and conduct a systematic evaluation using the 'depends' and 'influences' relationships [9].

Today, for MobileApps, more robust network infrastructures and smart mobile devices have led to increased functionality and capability thereby warranting special attention in comprehending how they are different from the UX point of view because user requirements, expectations, and behavior can be somewhat different. For instance, the quality design and evaluation of operability from a system viewpoint has a much different and greater influence for MobileApp usability and UX due to the size of the screen and context of the user. Characteristics such as button size, placement, contextual help, and widget usage for example have a much greater impact on task completion rates and task error rates [1, 3, 4, 15]. Ultimately, UX characteristics are very often neglected in quality modeling or seldom placed appropriately in quality views and this is magnified in a mobile context [10, 12].

Given this, there is a need for a characterization of MobileApps considering non-functional aspects in both UX and EQ. Consequently, the particular features of MobileApps –regarded both as a system and a system-in-use entity- pose new challenges regarding current quality models and their included and more relevant characteristics and sub-characteristics, as well as the particular attributes or properties to be measured and evaluated. Additionally, for MobileApps, there is an increased emphasis on many contextual elements related to the task and therefore the QinU. So, starting with the task at hand, and applying our 2Q2U model [12], we can incorporate the importance of task and particular MobileApp UX factors into the design of MobileApps. Starting with MobileApp UX, we can outline practical guidelines to design a MobileApp with optimal EQ characteristics based on UX goals.

Ultimately, the specific contributions of this research are: (a) Analyzing UX and usability for MobileApps in the light of 2Q2U v2.0 quality models; (b) Characterizing relevant quality features of MobileApps with regard to context, system-in-use and system; and (c) Illustrating MobileApp features and their potential impact in MobileApp quality design and evaluation.

Following this introduction, Section 2 outlines our nonfunctional requirements component and 2Q2U quality framework for better understanding where context, UX and usability fit in, among other issues. Section 3, discusses relevant features of MobileApps (both as a system and a system-in-use entity category) useful for designing and evaluating UX and usability. In Section 4, we discuss the usefulness of the proposed framework, while examples of UX/usability attributes and screens for MobileApps are illustrated. Section 5 describes related work and, finally, Section 6 draws our main conclusions and outlines future work.

2 Nonfunctional Requirements and the 2Q2U Quality Framework

In [12] we have updated the ISO 25010 standard [7] regarding quality models for evaluating new generation WebApps. The new quality models and framework is called 2Q2U v2.0, which also relies on the C-INCAMI (*Contextual-Information Need*, *Concept model*, *Attribute*, *Metric* and *Indicator*) measurement and evaluation conceptual framework [13]. C-INCAMI is structured in six components, namely: i) *Measurement and Evaluation Project*; ii) *Nonfunctional Requirements*; iii) *Context*; iv) *Measurement;* v) *Evaluation*; and vi) *Analysis and Recommendation*.

Of particular interest to illustrate in this article for understanding and instantiating MobileApp quality models are C-INCAMI's Nonfunctional Requirements and Context components, both shown in Fig. 1 –below, key words in the figure are highlighted in italic. The Nonfunctional Requirements component (labeled *requirements*) specifies the *Information Need* of any measurement and evaluation project; that is, the *purpose* (e.g. "understand", "improve") and the *user Viewpoint* (e.g. "developer", "mobile final user"). In turn, it *focuses* on a *Calculable Concept* (whose *name* is for instance "External Quality", "User Experience", etc) and *specifies* the *Entity Category* to evaluate –e.g. a system, system-in-use *super Category*-, by means of a concrete *Entity* –e.g., the "United flight booking".

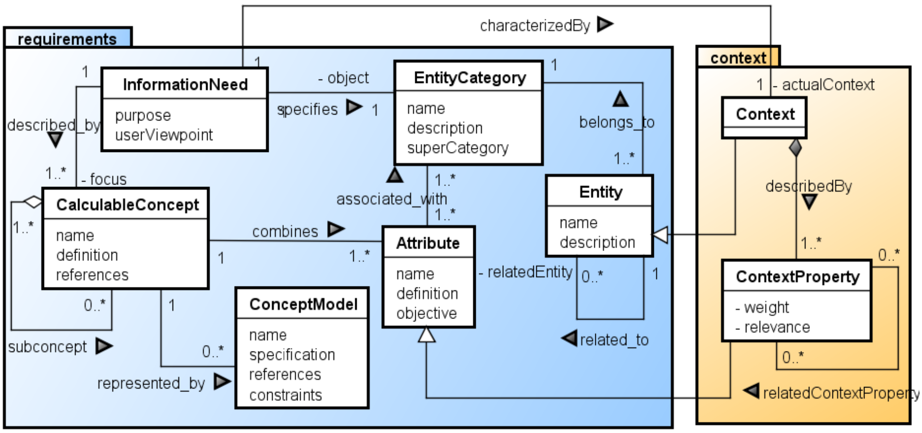

Fig. 1. C-INCAMI Nonfunctional Requirements and Context components

On the other hand, a calculable concept and its *sub-concepts* (e.g. "Usability") can be *represented by* a *Concept Model* (e.g. in "External Quality model") where the leaves of an instantiated model (requirements tree) are *Attributes associated with* an entity category. In fact, the concrete entity *belongs* to an entity category whose *name* is e.g. "Flight MobileApp".

The context component (labeled *context* in Fig.1) delineates the state of the situation of the entity to be assessed with regard to the information need (see *characterized By* relationship). *Context,* a special kind of *Entity* in which related relevant entities are involved, can be quantified through its related entities such as the environment of a mobile user, location, user age, user tasks, device type and screen size, etc. or even networking environment and performance.

To describe the context, properties of the relevant entities, which are also attributes called *Context Properties* are used. Context is particularly important regarding QinU requirements as instantiation of requirements must be done consistently in the same context so that evaluations and improvements can be accurately compared. But also context in a given project can be important regarding EQ requirements, as we discussed in [12], in which also we have proposed to delete the ISO 25010 *Context Coverage* characteristic from its QinU model, since as shown here the context specification can be represented independently of quality models.

Regarding *Concept* (quality) *Models* we have recently enhanced the ISO 25010 external quality and quality in use models, while maintaining many characteristics and the 'depends' and 'influences' relationships between both views. Fig. 2 depicts the quality model enhancement named 2Q2U, which was used in different studies [8, 9, 12].

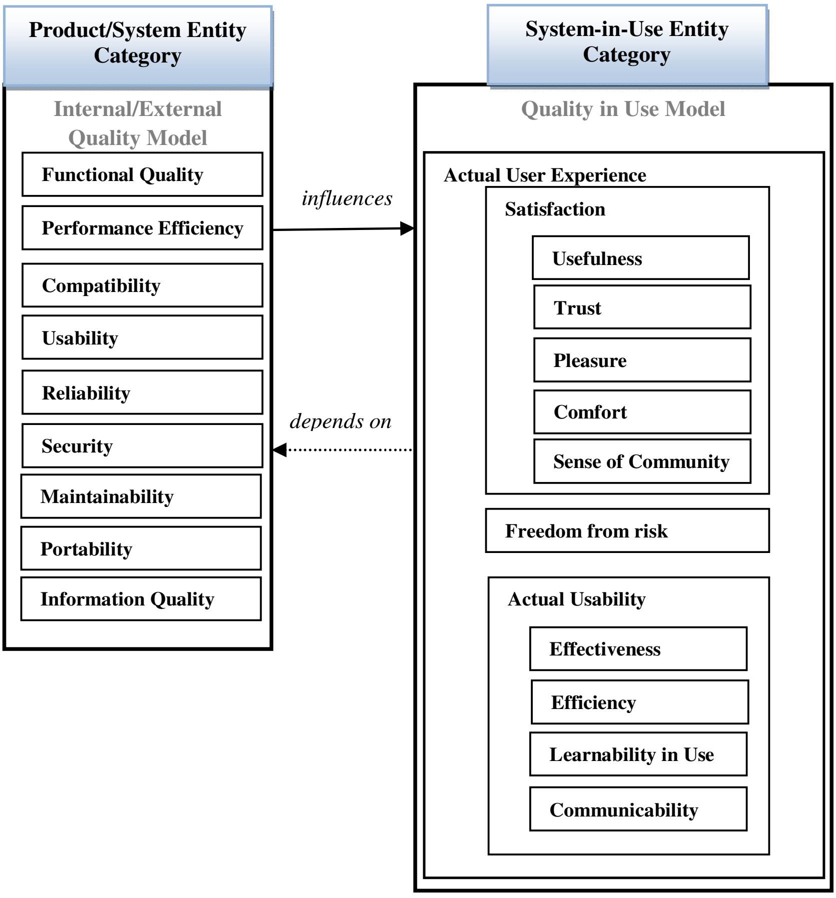

Fig. 2. 2Q2U v2.0 model characteristics with some sub-characteristics and relationships

Table 1. Definitions of QinU characteristics and sub-characteristics that are absent in ISO 25010 or were rephrased in 2Q2U v2.0

2Q2U v2.0 QinU Characteristic or Sub-characteristic Definition	ISO 25010 QinU Definition
Actual User Experience: Degree to which specified users can achieve actual usability, freedom from risk, and satisfaction in a specified context of use	Note: Absent calculable concept
Actual Usability *(*synonym *usability in use):*Degree to which specified users can achieve specified goals with effectiveness, efficiency, learnability in use, and without communicability breakdowns in a specified context of use	Note: Absent calculable concept, but similar concept (i.e. *usability in use*) was in the ISO 25010 draft
Effectiveness: Degree to which specified users can achieve specified goals with accuracy and completeness in a specified context of use	*Effectiveness*: Accuracy and completeness with which users achieve specified goals
Efficiency*:* Degree to which specified users expend appropriate amounts of resources in relation to the effectiveness achieved in a specified context of use	*Efficiency:* Resources expended in relation to the accuracy and completeness with which users achieve goals
Learnability in use*:* Degree to which specified users can learn efficiently and effectively while achieving specified goals in a specified context of use	Note: Absent calculable concept
Communicability*:* Degree to which specified users can achieve specified goals without communicative breakdowns in the interaction in a specified context of use	Note: Absent calculable concept
Sense of Community: Degree to which a user is satisfied when meeting, collaborating and communicating with other users with similar interest and needs	Note: Absent calculable concept

On top of each column (Fig. 2) the entity category is shown. For the QinU focus the *Entity Category* is named "System in Use", while for the EQ focus the *Entity Category* is named "System". For the QinU model we have added two main characteristics which are absent in the ISO standard viz. Actual User Experience, and Actual Usability. These concepts are related hierarchically as shown in Fig. 2 and also defined in Table 1.

For the EQ model we have added a new characteristic named Information Quality (which includes sub-characteristics such as Information Accuracy and Information Suitability), and rephrased others such as Functional Quality (which includes sub-characteristics such as Functional Accuracy and Functional Suitability) instead of the ISO functional suitability name. Also, we have also rephrased Usability as the "*degree to which the product or system has attributes that enable it to be understood, learned, operated, error protected, attractive and accessible to the user, when used under specified conditions*". Ultimately, the rationale of these adaptations is in [12].

In using the ISO 25010 quality models for MobileApps, it is assumed that the software quality models, definitions, and concepts in the standard were intended for application to software information systems as a whole and therefore are also applicable to a great extent to MobileApps, a type of software application. Moreover, we have built up 2Q2U on top of ISO standards in order to consider new features of new generation WebApps. We argue that using 2Q2U is sufficient for instantiating quality models for either native MobileApps or web-oriented MobileApps, as we see later on.

Next, we examine several features and entities relevant for the quality design and evaluation of MobileApps. In doing so, we consider three main aspects of the above C-INCAMI framework, namely: i) Context; ii) Entity Category, regarding both system and system in use; and iii) Calculable Concepts, i.e. quality characteristics, sub-characteristics and attributes, regarding both the EQ and QinU focus. The rationale for considering these aspects relies on a previous work [9], where we instantiated QinU and EQ models for the purpose of understanding and then combined these instantiations with SIQinU (*Strategy for Improving Quality in Use*) to carry out evaluations to ultimately accomplish improvement in a WebApp.

3 Featuring MobileApp Usability and UX

QinU, UX and Usability have all recently come to the research forefront due to a general shift in emphasis to satisfying the end user as part of the customer experience. For MobileApps (including native MobileApps and Mobile WebApps), they become even more important due to the significance of the user context. In particular the user's activity at the time of usage, location, and time, amongst other influencing factors have significant impact on the quality of the user's experience.

This section examines several features and entities relevant for designing and evaluating UX, Usability, Information Suitability, etc. for MobileApps. For doing this, we consider the above three main aspects: i) Mobile contextual factors; ii) Mobile system-in-use (QinU) factors; and iii) Mobile system (EQ) factors.

3.1 Mobile Contextual Factors

The context of a MobileApp user is much different than a traditional WebApp or desktop user not only due to the size of the screen but other factors that influence the user's environment and therefore their behavior. A few of these factors include:

- *Activity*: What the user is doing at the time of usage has a significant influence on the user's attention span. For example, if they are driving, then they have a very short attention span, maybe 1 second, versus if they are in the middle of a conversation, perhaps they have an attention span of 3 seconds. On the other hand, if they are at home sitting on the sofa, then the user may have other distractions as well;

- *Day/time of day*: The day and time can impact what a user is doing, and the level of natural light. Unlike desktop or WebApps which are typically accessed indoors, the usage of MobileApps is particularly sensitive to this contextual factor influencing visibility;
- *Location*: The location of the user influences many elements. For instance, indoors, outdoors, in a car, in an elevator, train or factory all of which can also be related to the user activity;
- *User profile*: The increasing complexity of software combined with an aging user demographic has an interesting effect on the usability of MobileApps. For aging users, usually their close range vision capability has diminished along with their dexterity. On the other hand, applications have become complex, and therefore function and content simple-ness and understandability are also critical and influenced by the particular user group. Not only are there more aging users, there are also more younger users as children these days begin using computing devices as toddlers;
- *Device*: The size and type of the device and its physical characteristics influences what the user can see (or not see) as well as the placement and number of controls and widgets in reduced real-estate displays. This shortage of resources combined with the amount of text (information conciseness) all impact the usability and UX of the MobileApp; and
- *Network performance*: Obviously the speed at which an application uploads and downloads data is going to have a great impact because of the decreased attention span. It may also be compounded by other contextual elements such as *Activity* as a user engaged in a time sensitive activity will have decreased tolerance for poor network performance.

These contextual elements in turn heavily influence the QinU of a MobileApp with the degree of influence dependent on the task being executed. By understanding these contextual factors, and defining the task at hand, those EQ characteristics and the design decisions that determine the system quality can be optimized to account for the MobileApp user's experience.

3.2 Mobile System-in-use Factors

As seen in Fig. 2, system-in-use quality has UX, Actual Usability, and Satisfaction as sub-concepts. UX is determined by the satisfaction of the user's be goals (hedonic), and do goals (pragmatic) as noted by Hassenzahl [5]. On the one hand, do-goals relate to the user being able to accomplish what they want with Effectiveness and Efficiency while be-goals relate to the user's satisfaction. Satisfaction [7] includes those 'soft' and subjective sub-characteristics including Usefulness, Trust, Pleasure and Comfort.

While completing a task, the user's experience depends on much more than just completing the task itself. The task, the context of the task, including the conditions under which the task is undertaken and the user's background have a strong influence on the user's experience. Different tasks for the same user as well as different users for the same task will cause totally different user experiences.

Hence, the MobileApp UX must be evaluated with respect to a real user performing a real task that includes several key system-in-use factors that have significant impact on the effectiveness and efficiency of the user. For example:

i) *Task Workflow*: For MobileApps design, Pareto's 80/20 rule (80 percent of the usage goes toward 20 percent of the functionality) applies more than ever because applications need to be designed with the most common tasks in mind that would be suited to mobile usage. Because of the context of use of a mobile user, and the mobile user's limited attention span, task workflow and length of the workflow is extremely important for this limited task set. As noted by Budiu [3], tasks most suited for mobile devices include a) tasks that have a deadline such as paying a bill or buying a gift at the last minute, b) tasks that require rapidly changing information such as bank balances, flight information, movie schedules, directions, and c) tasks that require privacy as small screens are ideal for privacy. Given these activities, if task workflows are not designed to be short, there is a higher probability of error and a lower rate of completion -see effectiveness definition in Table 1. Workflows therefore need to be compressed by combining several steps into one through careful task definition and analysis. Reduced workflows, in turn, reduce task times and increase Efficiency (see definition in Table 1) while at the same time, reducing error rates and error rate reduction is extremely critical for users with short attention spans. If you are driving and executing a task and get an error, do you continue trying?;

ii) *Learnability*: For those applications that are more complex, Learnability in use (see definition in Table 1) is critical. Users that cannot learn how to quickly become masters of your application with soon look for and find another one by the competition. The App Store paradigm has created this behavior because so many applications in the same domain are either extremely inexpensive or free.

Using 2Q2U v2.0 in conjunction with a measurement and evaluation strategy documented in [9], we can define the task, and start with the QinU, noting the importance of task for the particular MobileApp and work backwards to determine the system characteristics and attributes that are necessary to achieve high UX performance. With this process, we can derive some key aspects of design from the system quality point of view while specifically taking QinU into account.

3.3 Mobile System Factors

Regarding system factors, because of these aforementioned contextual and system-in-use aspects, designing a MobileApp –as a system from the entity standpoint- needs to incorporate several elements. These elements include sub-entities (e.g. widgets, menus, forms, etc.) and their associated attributes and characteristics (e.g. Operability, Understandability, Functional Suitability, etc.). Some of the typical sub-entities for MobileApps that should to be considered for quality design and evaluation are:

- *Typing/input*: which includes search bars, and other data entry fields whereby the users should be assisted as much as possible to reduce errors and the 'cost' of typing. This includes such measurable attributes as default values, default value removal and shortcuts.

- *Entry widgets* such as carousels, drop down boxes and lists. System designers need to prevent the need for typing and reduce error rates by using widgets.
- *Sort, search and filter*: Special considerations are needed for MobileApps in order to reduce the workload and typing input. In addition, the small screen size makes it easy for the user to lose context, so attributes like typo tolerance, predictive contextual help would be desirable.
- *Menus* should be limited, simple and easily navigated with a clear breadcrumb path showing where the user has come from and where they can go. This is mostly applicable to Mobile WebApps where a small screen limits the users' context or field of vision in navigating from one place to another.
- *Forms and registration*: Forms need be clear with context sensitive help. The last thing you want is a user unable to complete a form because they didn't quite understand one particular mandatory field. Either defaults, or help within the entry field giving an example of what goes in the field should be provided.

Regarding the aforementioned sub-entities of typical MobileApps, we can now consider some examples of sub-characteristics of Usability, Information and Functional Quality that are particular to MobileApps, for example:

i) *Learnability*: Through its various entities listed above such as menus and widgets, learnability can be designed into the MobileApp through defaults, facilitating predictive actions, context sensitive help, and so on;

ii) *Navigation*: The MobileApp's ability to enable to a user to easily find the functionality or information that they need is critical. Not only do they need to easily find it, they need to do it fast. These are sometimes in alignment but not necessarily;

iii) *Operability*: This is a central sub-characteristic of usability. It means the degree to which a MobilApp has attributes that make it easy to operate and control. For example, controllability represents the degree to which users can initiate and control the direction and pace of the task until task completion. Easy to operate is related to those provided mechanisms which make entering data as easy and as accurate as possible while maintaining consistent usage and placement of controls even in different contexts and platforms of use;

iv) *Error handling*: Error prevention, error awareness, and error status are key attributes that need to be designed correctly to not only prevent errors, but enable the mobile user to recover with minimal effort. In addition, errors should be easily and quickly understood so that the user can move forward with their task;

v) *Understanding*: For MobileApps, because the screen is so small, it requires special consideration for understanding what the application is about, and what it does almost instantly. As mentioned previously, mobile users have a very short attention span so they must glance at the application and understand how it operates. For instance, an airline application, when they open it up, they already know they want to, either check a flight status or make a reservation, so the design must heavily consider this expectation so that there is reduced ramp up time;

vi) *Visibility*: Many factors determine whether or not the application is easily visible to a user. Depending on the context, different text colors and backgrounds can have a positive or negative impact. This sub-characteristic, while related to aesthetics is not identical. Remember that mobile users want to glance quickly and understand almost immediately and there may be glare on their screen if they are outdoors. This means appropriate usage and placement of text in appropriate format can impact the user's speed of comprehension greatly.

Finally, among other sub-characteristics and attributes to take into account for evaluating MobileApps is Information Conciseness, which in few words means shorter is better. In [12] Information Conciseness is an attribute of the Information Coverage sub-characteristic, which in turn is related to Information Suitability, and is defined as "*degree to which the information coverage is compactly represented without being overwhelming*".

In alignment with the ISO 25010 'influences' relationship (see Fig. 2), each of these system sub-characteristics/attributes can have an influence on the system in use for both the do-goals and be-goals. Depending on the context, user, and task, the influence can impact the user's ability to operate, navigate and use the application efficiently (do-goals) or comfortably and with pleasure and satisfaction (be-goals).

Thus, in designing a MobileApp, one must also take into account the decision to make a native MobileApp versus a Mobile WebApp. For instance, navigability, as described above, and congruence and consistency with the regular WebApp are significant. On the other hand, many widgets and other features of the mobile phone are not accessible to WebApps so they inherently start at a disadvantage regarding widget usage which, depending on the application, can impact system in use greatly.

4 Illustrating Quality Design and Evaluation for MobileApps

Now, given the aforementioned 2Q2U framework and mobile UX and usability concepts we can apply them to MobileApps. This section builds upon Section 3 and demonstrates the importance of a few of the factors through some examples whereby the context, system, and system in use come into play for MobileApps. Note that a systematic instantiation of quality models for MobileApps -as we did in [9] for the JIRA WebApp's QinU and EQ- will be documented in a separate manuscript.

When designing MobileApp tasks for QinU, for example for evaluating Effectiveness and Efficiency characteristics, content and functions are embedded in the task design itself rather than as attributes of the application. And, mentioned previously, UX do-goals for a MobileApp-in-use are heavily influenced by the workflow of the task. Because the most prevalent mobile tasks include deadline oriented, or time sensitive information, the workflow should be short in order to be effective and efficient to make up for the small screen size and user context (in a hurry with short attention span). User's are often executing tasks 'on the spot' and 'in the moment' so a delay, or a mistake is critical at that moment. Can you imagine getting stranded at an airport, and then looking up all the flights going out of an airport and not being able to search effectively because the search menu was designed only with one search parameter?

From this simple example, it is easy to see that QinU depends on many system sub-entities having particular sub-characteristics and attributes depending on the context, user and task. Note that when executing a particular task, there should be very few controls that could lead the user astray or toward a mistake (see for example the Fidelity Investment MobileApp screenshots in Fig. 3.a and b). They purposely design the widgets and controls to be large, thumb friendly, so that the user can complete the task given the information needed as effectively and efficiently as possible. There is contextual help, or a simple drop down menu with limited choices in the fields that need to be filled in.

As mentioned for the typing/input sub-entity, defaults are extremely important to reduce the workload of the user. Defaults can be based on what the user has typed or submitted in the past (e.g., zip codes, names, addresses). Defaults are bound to be wrong sometimes and you don't want to have a default that takes unnecessary effort to change. For example don't use 0 as the default for a telephone number or zip code and don't make users manually erase the text field character by character, by clicking the Delete key. Instead, erasing defaults with a single button that clears the entire field can save several seconds. In Fig. 3.a, the MobileApp shows the default user name as well as the option to save it or not, giving the user a sense of controllability. Also, for typing/input and widget sub-entities, the United Airlines MobileApp shown in Fig. 3.c, shows a carousel for data entry rather than typing. It also gives the user another option for input. Additionally, the screen is very clear for the task at hand. Notice that there is no other input or information not directly related to the task of Finding a flight.

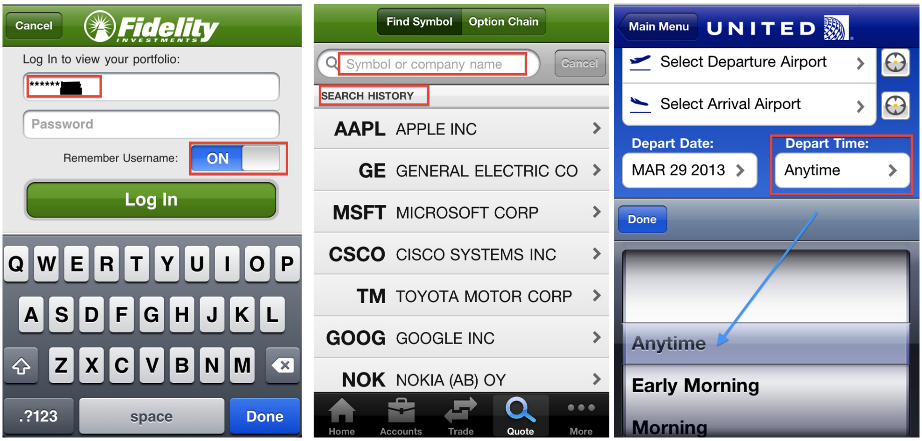

Fig. 3. a) Fidelity Investment MobileApp login screen; b) Fidelity Investments MobileApp find a ticker symbol task; and c) United Airlines MobileApp finding a flight

Examining Fig. 3.b, shows the Fidelity Investment MobileApp for the task of "Finding a stock ticker symbol". Notice the context sensitive help in light grey color symbol or company name, which aids the user. Also, there is a list of past searches which helps the user as well. Both the ticker symbol and company name are shown to eliminate any confusion and prevent using the incorrect ticker symbol. The information is also presented in a clear and concise manner with no other information (information conciseness) to distract from the task at hand while features have been designed to reduce errors and increase efficiencies and task completion rates. The last thing Fidelity wants is for a user to not complete this task which may affect executing a trade.

Figures 4.a and b show the Embark application for the Washington DC Metro train system. Notice that (Fig. 4.a) for the task of "finding a station", the user is given context sensitive help below the search bar. There is also light grey letters in the search bar to make sure the user is not lost, reminding the user what they are trying to do. Also notice that the search bar is quite long, thus providing the user with ample space to type in their desired location. On the other hand, for the same entity, Washington DC Embark application, you can see from Fig. 4.b that the errors are not comprehensible thus very difficult for the user to recover and move forward with their task.

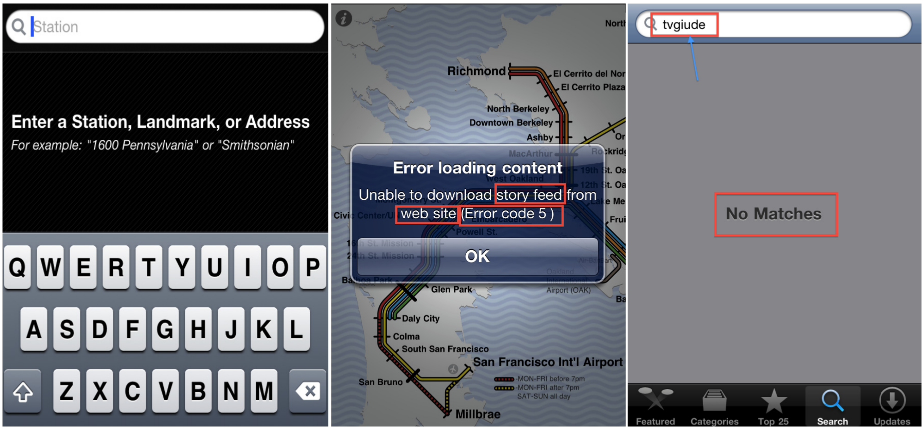

Fig. 4. a) and b) Embark Washington DC Metro MobileApp; c) Apple App Store MobileApp

Lastly, Fig. 4.c shows a search function implemented in the Apple App Store. For the App Store, a user in search of 'tvguide' accidently typed 'tvgiude' and got no results. Thus in terms of typo tolerance and error prevention, it would rate poorly for that particular task.

These examples demonstrate MobileApp sub-entities such as forms, menus and search boxes and why design principles, i.e. mobile system factors can significantly influence the UX and other system in use factors. With a screen so small, it is easy to see why concise information, contextual help, etc. are critical attributes.

5 Related Work and Discussion

Given the evolution of WebApps both as systems and systems in use, characteristics such as QinU and UX, combined with usability and information quality, have all recently come to the research forefront especially for MobileApps due to advances in mobile device capabilities concurrent with cloud technology further jettisoning SaaS-based offerings. In addition, the shift in emphasis to satisfying the customer experience has put greater corporate focus on UX and customer satisfaction. However, based on design and usability guidelines for MobileApps [1, 3, 4, 11], quality standards [7], and related literature [6, 10, 14, 16], it was difficult for us to understand their relationships into quality models.

Hence, it is important in this discussion to highlight definitions given by various researchers to the UX term. For instance, Bevan [2] examined ISO 25010 from the viewpoint of actual usability and UX and drew relationships regarding usability as performance in use, and satisfaction as it relates to UX. Hassenzahl's work [5] in classifying UX in two categories, hedonic and pragmatic is also useful when examining actual usability from the *do*, pragmatic viewpoint and the *be*, or hedonic satisfaction standpoint. To understand the term requires breaking down the word 'user experience' and examining first what experience means.

Experience is a general concept that refers to 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 system. Hassenzahl notes that the time dimension could be either momentary or accumulated and changing over time. Examining the 'user' part of the UX concept, Hassenzahl characterizes a user's goals into pragmatic, do goals and hedonic, be goals and assumes the system-in-use quality is perceived in two dimensions, pragmatic and hedonic. Pragmatic quality refers to the system's perceived ability to support the achievement of tasks and focuses on the system's actual usability in completing tasks that are the 'do-goals' of the user. Hedonic quality refers to the system's perceived ability to support the user's achievement of 'be-goals', such as being happy, or satisfied with a focus on self.

Hassenzahl also argues that the fulfillment of be-goals is the driver of experience and that lack of actual 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.

These two concepts, viz. hedonic and pragmatic have been recently introduced in the ISO 25010 standard to refer to satisfaction and effectiveness/efficiency characteristics respectively. However, UX and actual usability are missing characteristics in the ISO standard, as we discussed in Section 2 for our 2Q2U quality models and framework.

On the other hand, in the Apple [1] and Google [4] design and user interface guidelines, the relationship between system and application design with usability and UX is not explicit, nor is it explained in the well-known usability and UX columns of Nielsen [11], nor in other quality-related researches such as [6, 10]. Budiu [3] lists out many characteristics of MobileApps in that would be desired in certain contexts of use but does not use quality modeling approach and therefore, the capability for consistent application using a framework to systematically apply concepts and evaluate and improve a MobileApp is limited.

In [16] authors discussed how the triangulation of various methods might provide insights about the identification of UX factors that can be detected both in early and later phases of the development process. They pointed out that "*It is important to associate the identifying UX factors with the artifacts used to the design. In our study, we have found that scenarios and task models works as a lingua franca for mapping user requirements and UX factors*" (cf. p. 42). However, this research was not intended to characterize UX in the framework of the QinU view nor its relationship with EQ (system) characteristics.

As discussed elsewhere [9], we can instantiate QinU and EQ models for the purpose of understanding and then combine these instantiations with a strategy to carry out evaluations to ultimately accomplish improvement. As such, based on our examination of existing research, there has been progress in the individual elements such as modeling and evaluation, but limited focus on using a strategy and tailored models for the purpose of improvement considering the QinU/EQ/QinU relationships and cycle.

6 Conclusions and Future Work

As the contributions mentioned in the Introduction Section, we have characterized relevant features of MobileApps with regard to usability and UX analyzed them using previously proposed 2Q2U v2.0 quality models. In the process, we looked at MobileApp usage context and its impact in MobileApp design when considering quality characteristics and attributes for system and system-in-use quality evaluation. Lastly, we illustrated some example MobileApps to show their design features could have a significant influence on the system quality and the user's perception of quality, i.e. system in use and UX.

Our proposal shows the usage of 2Q2U v2.0 as an integrated approach for modeling requirements for external quality and quality in use (i.e., actual usability, satisfaction and user experience), combined with a consistent, and flexible way for representing calculable concepts (characteristics), sub-concepts (sub-characteristics) and attributes used in conjunction with the C-INCAMI measurement and evaluation components and its strategy.

Another manuscript will expand the work further including the definition of attributes, metrics, and indicators for user group types performing specific tasks in real MobileApp context of use.

Ongoing research is focused on further utilizing the 2Q2U v2.0 framework for systematic instantiation of quality models for MobileApps –as we did in [9]- in order to provide foundations when modeling and understanding the relationships among EQ, QinU, actual usability and UX. In doing so, our end goal is improvement. That is, to be able to directly use the system-in-use quality evaluation to directly improve the design of the MobileApp system. This concern has often been neglected in the literature, but may help improve quality design recommendations and ultimately to increase the MobileApp UX as a whole.

Acknowledgments. Thanks to the support given from the Science and Technology Agency, Argentina, in the PAE-PICT 2188 project, and in the 09-F047 project at Universidad Nacional de La Pampa, Argentina. We also thank the State Key Laboratory of Software Development Environment of China under Grant No. SKLSDE-2012ZX-13 and National Natural Science Foundation of China under Grant No. 90818017.

References

1. Apple iOS Human Interface Guidelines, http://developer.apple.com/library/ios/#documentation/UserExperience/Conceptual/MobileHIG/Introduction/Introduction.html (retrieved April 2013)
2. Bevan, N.: Extending quality in use to provide a framework for usability measurement. In: Kurosu, M. (ed.) Human Centered Design, HCII 2009. LNCS, vol. 5619, pp. 13–22. Springer, Heidelberg (2009)
3. Budiu, R., Nielsen, J.: Usability of Mobile Websites, http://www.nngroup.com/reports/mobile (retrieved April 2013)
4. Google User Interface Guidelines: http://developer.android.com/guide/practices/ui_guidelines/index.html (retrieved April 2013)
5. Hassenzahl, M.: User experience: towards an experiential perspective on product quality. In: Proc. 20th Int'l Conference of the Assoc. Francophone d'Interaction Homme-Machine, IHM, vol. 339, pp. 11–15 (2008)
6. Herrera, M., Moraga, M.Á., Caballero, I., Calero, C.: Quality in Use Model for Web Portals (QiUWeP). In: Daniel, F., Facca, F.M. (eds.) ICWE 2010 Workshops. LNCS, vol. 6385, pp. 91–101. Springer, Heidelberg (2010)
7. ISO/IEC 25010: Systems and software engineering. Systems and software Quality Requirements and Evaluation (SQuaRE). System and software quality models (2011)
8. Lew, P., Qanber Abbasi, M., Rafique, I., Wang, X., Olsina, L.: Using Web Quality Models and Questionnaires for Web Applications Evaluation. In: IEEE Proceedings of QUATIC, Lisbon, Portugal, pp. 20–29 (2012)
9. Lew, P., Olsina, L., Becker, P., Zhang, L.: An Integrated Strategy to Understand and Manage Quality in Use for Web Applications. Requirements Engineering Journal 17(4), 299–330 (2012)
10. Nayebi, F., Desharnais, J.-M., Abran, A.: The state of the art of mobile application usability evaluation. In: 25th IEEE Canadian Conference on Electrical Computer Engineering, pp. 1–4 (2012)
11. Nielsen, J.: Mobile Usability Update, Jakob Nielsen's Alertbox (September 26, 2011), http://www.nngroup.com/articles/mobile-usability-update/ (retrieved April 2013)
12. Olsina, L., Lew, P., Dieser, A., Rivera, B.: Updating Quality Models for Evaluating New Generation Web Applications. Journal of Web Engineering, Special Issue: Quality in New Generation Web Applications 11(3), 209–246 (2012); Abrahão, S., Cachero, C., Cappiello, C., Matera, M. (eds.)
13. Olsina, L., Papa, F., Molina, H.: How to Measure and Evaluate Web Applications in a Consistent Way. In: Rossi, Pastor, Schwabe, Olsina (eds.) Springer HCIS Book Web Engineering: Modeling and Implementing Web Applications, pp. 385–420 (2008)
14. Sohn, T., Li, K.A., Griswold, W.G., Holland, J.: A diary study of mobile information needs. In: ACM: Conference CHI 2008, Florence, Italy, pp. 433–442 (2008)

15. Thom-Santelli, J., Hedge, A.: Effects of a multi-touch keyboard on wrist posture, typing performance and comfort. In: Proc. of the Human Factors and Ergonomics Society, 49th Annual Meeting, Santa Monica, USA, pp. 646–650 (2005)
16. Winckler, M., Bach, C., Bernhaupt, R.: Identifying User eXperiencing Factors along the Development Process: A Case Study. In: Lai-Chong Law, E., Abrahão, S., Vermeeren, A., Thora Hvannberg, E. (eds.) Proc. of the 2nd International Workshop on the Interplay between User Experience Evaluation and Software Development (In conjunction with the 7th NordiCHI), Copenhagen, Denmark. CEUR Workshop Proceedings, vol. 922, pp. 37–42. CEUR-WS.org (2012)