

A Methodology for Structured Use-centered Quantitative Full-life-cycle Usability Requirements Specification & Usability Evaluation of Web Sites

Guoqiang Hu and Kai H. Chang

Department of Computer Science and Software Engineering

Auburn University, Auburn, AL 36849-5347, USA

Phone: 1-334-844-6322

Fax: 1-334-844-6329

{guoqiang, kchang}@eng.auburn.edu

Abstract - World Wide Web has gained its dominant status in the information and services delivery world in recent years, but how to build a good usability web site is still a problem. In this paper, we proposed a methodology for structured use-centered quantitative full-life-cycle usability requirements specification and usability evaluation of web sites. Our approach is that: the system's usability is defined in terms of its consisting goal-tasks' usability; a goal-task's usability is defined in terms of several major usability aspects; each major usability aspect is further defined in terms of several basic use features. In this way, a quantitative web sites usability engineering framework is set up. According to this framework, each goal-task's usability requirements are specified in terms of its basic use features, and then are tested against to see if they are satisfied; a system's usability can be easily derived by combining its consisting goal-tasks' usabilities through a weighted scheme.

Keywords: Web Sites Usability, Usability Metrics, Use, Use Features, Quantitative Usability Requirements & Evaluation, QUEST.

1 Introduction

World Wide Web consists of tens of millions of Web sites or Web-based applications¹ [1], which are distributed all over the world. Because of WWW's significant value to all of us, how to require, evaluate and thus to improve the usability of web sites is a big concern to all the stakeholders. However, currently there exist no good ways to address this issue. In order to help solve this problem, we propose a methodology that features a structured use-centered quantitative full-life-cycle method to assist usability requirements specification and usability evaluation of web sites.

The principle of this methodology is: A system's usability is defined in terms of its consisting goal-tasks' usabilities; a goal-task's usability is defined in terms of several major usability aspects; each major usability aspect is further defined in terms of several basic use features. In contrasting to the basic use features, a system's usability, a goal-task's usability, and the major usability aspects of a goal-task are all called composite or

derivative use features, because they are derived by combining their respective components through a weighted scheme.

The applying process of this methodology is: At system analysis stage, after task analysis, each task is assigned its usability user requirements by specifying the required quantitative value for each of its basic use features. Here, the weight and use frequency of each task can also be specified because at this time the task's importance and use frequency in the target system can be clearly determined according to the analysis of the current system. After these quantitative specifications having been obtained, the composite use features of each task and the usability of the entire system can be derived. Then, all the above information together forms the usability requirements specification of the entire system. Apparently, the usability user requirements specification should be agreed upon between the system analyzer and end user(s) (sometimes, even the procurer in lieu of end users). The key point to be considered here is the economic (budget) implication of the usability user requirements specification, because as quality requirements, the higher the usability user requirements are, the more expensive the target system will be. It should be noted that the usability requirements have equal status with other user requirements. So, at all the other stages of the lifecycle, each time a review or testing is needed, the usability requirements specification will be tested against just like functional requirements specification is, the only difference is the testing methods used, i.e., for the functional requirements specification, the testing method is the traditional software testing, for the usability requirements specification, the testing method is usability review or usability testing by use.

It should be noted that this quantitative usability methodology is independent of, and therefore can be seamlessly integrated into, any engineering methodologies, processes and techniques. Take Waterfall model for example, this methodology is illustrated in Figure 1.

The rest of this paper is organized as follows. Section 2 introduces some important features of web sites that are critical for understanding our approach to web sites usability. Section 3 is the core of this paper. It first addresses several key issues related to our methodology and then provides the complete set of quantitative usability definitions of a web site. Section 4 briefly discusses the possible benefits of this methodology. Section 5 concludes this paper by a call for empirical studies.

¹ For convenience, in this paper, Web sites or Web-based applications will be uniformly referred to as web sites.

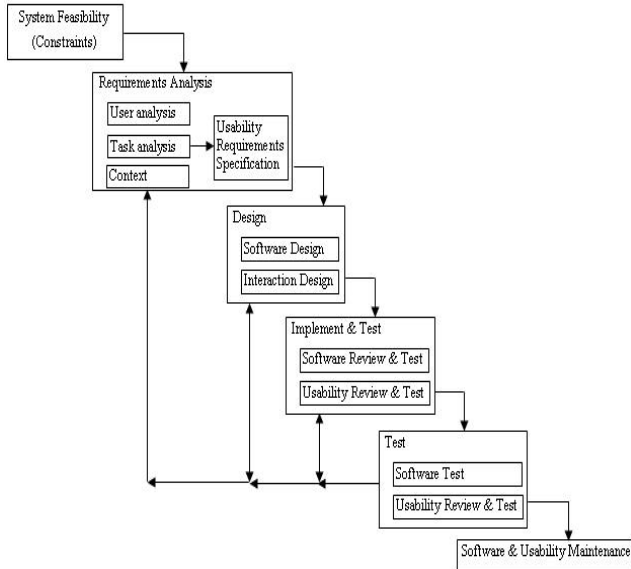


Figure 1. The quantitative full-life-cycle usability methodology illustrated in Waterfall model

2 Some features of web sites

2.1 Unification of functions and contents

Generally speaking, no matter what its purposes are, any web site is a nonlinear composition of browsing items and functional services items that are presented on web pages and linked together by links. Different from traditional applications that strictly differentiate between commands and documents, web sites do not distinguish between functional services items (commands) and browsing items (documents), because, on the WWW, both kinds of items are presented as a series of web pages. So, on the surface (in terms of presentation and behavior), functional services item and browsing item on the WWW are not that different from each other as their traditional counterparts on traditional applications are. This distinctive feature makes the WWW extremely flexible and has proved to be a major beauty and strength of WWW, that is, Web-based applications are easily all-purpose-composable.

Because of the above reason, on the WWW, both kinds of items — functional services items and browsing items — can be given a unified term: **Conceptually Independent Composing Item (CICI)** (read as *kick*). Strictly, CICI is defined as a Web object unit, which is conceptually independent and indivisible. Typical instances of CICI, for example, are things like a complete online article, a complete online book, a complete ordered series of functional transaction pages, etc. Each CICI consists of a series of web pages and on each of its web pages, besides its presentation, it can be associated with methods (presented as links or buttons) that operate on it. In essence, a CICI is simply a designed goal-task supported by a web site. Contrasting to normal CICI's, the navigation mechanism of a web site is a unique designed goal-task solely for gluing the whole web site together and providing a means to navigate between and beyond the web site's CICI's.

2.2 Web sites are dynamic

Web sites are extremely dynamic. Some web sites can be updated many times a day; a web site's user population can also

be very dynamic; some user may only be interested in a small part of a web site. But no matter what, walk-up-and-use for everybody is a default usability expectation for most web sites.

2.3 Navigating mechanism is content-ized

A web site's navigating mechanism is analogous to the menu system of traditional software. But because of the Web's "all-purpose composability" mentioned above, a web site's navigating (organization) architecture can often be so content-ized (or extended) that the traditional distinctions between the navigating items (menus) and the real contents become blurred (each "menu" can be a very descriptive page). Even so, the main purpose of a web site's navigating mechanism is still to provide an efficient means for an end user to reach the desired CICI's.

3 Quantitative usability of web sites

3.1 What is usability

The definition of usability for software products was attempted by Miller [2], Shackel [3][4][5][6], Bennet [7][8], Shneiderman [9][10][11], Nielsen [12], Bevan [13], Dix [15], Löwgren [14], Quesenbery [16][17], etc. In 1998, ISO 9241-11 [18] defined usability as:

Usability: The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.

Goal: Intended outcome.

Effectiveness: The accuracy and completeness with which users achieve specified goals.

Efficiency: The resources expended in relation to the accuracy and completeness with which users achieve specified goals.

Satisfaction: Freedom from discomfort, and positive attitude to the use of the product.

Context of use: Characteristics of the users, tasks and the organizational and physical environments.

Task: Activities required to achieve a goal.

It is the ISO 9241-11 usability definition that has been recognized as authoritative and become widely adopted. But in this definition, there are some ambiguities and vagueness that have caused usability problems of the definition itself in practice:

Firstly, ISO 9241-11 definition of usability does not specify clearly the goal and task defined in it should be the designer's goal and task or the intended end user's goal and task. We believe that, in order to achieve good usability, the goal and task defined in a definition of usability apparently should be the intended end user's goal and task. The designer can also have goal and task in mind, but the designer's goal and task should match the intended end user's goal and task as closely as possible, because there is no doubt that the bigger is the difference between the two pairs, the severer will be the usability problems caused by this difference in the final system.

Secondly, ISO 9241-11 definition of usability does not specify against what and how to measure the *effectiveness*, *efficiency* and *satisfaction* defined in it; also, it does not specify how to combine these measures into a single aggregate usability measure of the entire system. Because of this problem, in

practice, it is impossible to quantify usability by just following it. The big downside of the inability to quantify the usability is that if you cannot measure it, you cannot manage and control it.

Thirdly, ISO 9241-11 definition of usability defines *efficiency* as an absolute “amount of resources expended in relation to the accuracy and completeness with which users achieve specified goals”. We believe this is not a good way to define efficiency, because in each absolute amount of resource expended, there are at least two portions: one portion that is rightfully expended in relation to the accuracy and completeness with which users achieve specified goals; the other portion that is wasted but is imposed-upon by an awkward design. Take time for example. It does not make much sense to measure the absolute amount of time expended on a task as efficiency of that task. The reason is that each work needs some time to finish, and depending on the complexity of the work, the necessary time can be very long or very short, and it does not matter how much time has to be expended, but it does matter how much time in the total is rightfully expended. Let’s assume the total amount of time expended is T , the part of T that is the rightfully expended is T_n (time necessary), the other part that is wasted (because of mistakes or awkward design) is T_w (time wasted). Apparently,

as a measure of efficiency of time expended on a task, $\frac{T_n}{T}$ or

$\frac{T - T_w}{T}$ makes much more sense than T and can be used to

compare the efficiency of any kind of work, no matter how big or small the work is. So, we believe efficiency should be defined as a ratio rather than an absolute amount.

Fourthly, ISO 9241-11 definition of usability does not pay any attention to each goal-task’s human–tool interaction process. If only *effectiveness*, *efficiency* and *satisfaction* are measured, the different specific aspects of usability problems related to a goal-task’s human–tool interaction process cannot be reflected in the final usability evaluation so that the final usability evaluation appears to be too abstract and empty. We believe that the different specific aspects of usability problems related to a goal-task’s human–tool interaction process need to be included in the definition of usability, because they not only determine the ease of use (including ease to learn), but also have much to do with the end user’s cognitive feeling about the goal-task, and most importantly, they are real usability problem causers. According to Norman’s “stages of action” model [19] (illustrated in Figure 2), there are two usability aspects related to a goal-task’s human–tool interaction process that can be measured: *the presentation (including feedback presentation) of the interaction* and *the organization (choreography) of the interaction*.



Figure 2. Norman’s “stages of action” model

3.2 Quantitative web site usability definition

Based on the above recognitions, we can improve the ISO 9241-11 definition of usability by providing the following quantified definition of web site usability.

3.2.1 Goal-task quantitative usability definition

End user’s goal (G_u): End user’s goal G_u is the intended outcome by end user.

Designed goal (G_d): Designed goal G_d is the intended outcome for end user by the design of a product.

Designed task: One or one of several possibly allowed sequences of activities required to achieve a goal planned for end user by the design of a product.

Use: A real execution of an improvised possible designed task by end user.

Use feature: A feature of a designed product that is significant or essential for use.

Designed context of use (C_d): Designed context of use C_d is a use feature that signifies the set of quantified or enumerable ranges of characteristics of the users, designed task and organizational and physical environments that are accepted as restrictions of use by the design. For example, the designed context of use of *balance transfer transaction* goal-task can be specified as:

```

Cd = { users { range of gender: male or female,
              range of age: ≥ 18,
              range of language: English,
              range of level of expertise: All levels,
              range of use purpose: Logged-in Personal Customer Online
              Services },
       computer { range of hardware configuration: All,
                 range of internet connection: All,
                 range of operating system: All,
                 range of browser: IE V5.0 or above, Cipher Strength: 128-bit,
                 range of browser plug-in: any )

```

Effectiveness (E_{fec}): Effectiveness E_{fec} is a composite use feature that signifies the accuracy and completeness in percentage with which end user can achieve his/her goal at the end of a use under a satisfied context of use. Effectiveness’s basic use features include *accuracy* (A_{gt}) and *completeness* (C_{gt}), and we define:

$$(1) E_{fec} = 50\% C_{gt} + 50\% A_{gt}$$

Efficiency (E): Currently, for web sites, we only consider the amount of time spent on a goal-task. So, efficiency E is a use feature that signifies the ratio in percentage between the amount of time that has to be expended perceived necessary by the end

user and the actual amount of time that has to be expended imposed-upon by the design to achieve the effectiveness. In practice, the end user will report the perceived usability problems (mistakes or awkward design) to usability expert after testing, and then the usability expert can analyze the design of the task again and find the amount of time that has been wasted because of the reported problems. Let's assume the total actual amount of time expended is T , the wasted amount of time is T_w , we define:

$$(2) E = \frac{T - T_w}{T}$$

Satisfaction of a use (S): Satisfaction of a use S is a use feature that signifies the degree in percentage of freedom from discomfort in a use and positive attitude to the use. In practice, satisfaction of a use can be obtained from end users through questionnaires.

Use interaction process aptness (U_p): Use interaction process aptness U_p is a composite use feature that signifies the comprehensive aptness in percentage of the interaction process in a use by taking into account a set of use interaction process use features. The basic use interaction process use features include:

Mistake-error intolerance ratio ($P_{m-e-int}$): The number of actions that cannot be corrected, undone, or cancelled divided by the total number of actions that have to be taken.

Mistake-error ratio (P_{m-e}): The number of actions that have caused mistake or error DUE TO the design divided by the total number of actions that have to be taken.

Imposed-upon awkwardness (P_{awk}): The number of actions that are unnecessary, unreasonable or awkwardly designed divided by the total number of actions that have to be taken.

Total-success ratio (P_{ts}): The number of end users who can finish the task (including those after correcting errors or mistakes and/or seeking any help) divided by total number of end users tested.

Unsuccess ratio (P_{uns}): $P_{uns} = 1 - P_{ts}$.

We define:

$$(3) U_p = 1 - (25\% P_{m-e-int} + 25\% P_{m-e} + 25\% P_{awk} + 25\% P_{uns})$$

Use interaction interface and presentation aptness (U_{p-int}):

Use interaction interface and presentation aptness U_{p-int} is a composite use feature that signifies the comprehensive aptness in percentage of the interfaces and presentations in the interaction process in a use by taking into account a set of interface and presentation use features. The basic use interaction interface and presentation use features include:

Confusing-misleading items ratio (P_{i-cm}): The number of confusing or misleading items related to the interaction process divided by the total number of interface items related to the interaction process.

Theme ratio inappropriate pages ratio ($P_{i-theme-in}$): Each page should have a theme. The size of the theme's valid presentation area divided by the size of the entire page's valid presentation area is the page's theme ratio. Theme ratio inappropriate pages ratio is defined as the number of pages whose theme ratio is less than 65% divided by the total number of pages of a task.

Methods-incomplete pages ratio ($P_{i-method-in}$): Besides the default set of global methods (see 3.3.3), each page should provide a complete set of methods that can be immediately used to operate on the states of the current CICI (task). Also, the short-cuts to start any logically follow-through CICIs should be provided. Methods-incomplete page ratio is defined as the number of pages with incomplete set of necessary methods divided by the total number of pages of a task.

Memory-overloading pages ratio (P_{i-mem}): The number of pages that force users to be memory-overloaded (i.e., users have to accurately remember more than 7 facts from previous pages) divided by the total number of pages of a task.

Distracting pages ratio (P_{i-dis}): The number of pages that have severely distracting extra features divided by the total number of pages of a task.

Layout and/or item-grouping inappropriate pages ratio ($P_{i-layout-in}$): The number of pages that have inappropriate layout and/or item-grouping divided by the total number of pages of a task.

Bad feedback pages ratio ($P_{i-fdbk-in}$): The number of pages that do not present expected or appropriate feedback to the previous page divided by the total number of pages of a task.

No/bad page level help pages ratio ($P_{i-hip-in}$): On each page, there should be page level help methods. In the content of page level help, the method for global help should be provided. No/bad page level help pages ratio is defined as the number of pages have no/bad page level help divided by the total number of pages of a task.

Bad readability pages ratio ($P_{i-read-in}$): The number of pages have bad readability divided by the total number of pages of a task.

Refreshing-abnormality pages ratio ($P_{i-refresh-in}$): the number of pages that change current theme presentation focus after refreshing divided by the total number of pages of a task.

We define:

$$(4) U_{p-int} = 1 - (10\% P_{i-cm} + 10\% P_{i-theme-in} + 10\% P_{i-method-in} + 10\% P_{i-mem} + 10\% P_{i-dis} + 10\% P_{i-layout-in} + 10\% P_{i-fdbk-in} + 10\% P_{i-hlp-in} + 10\% P_{i-read-in} + 10\% P_{i-refresh-in})$$

Goal-task's use-centered usability (U_{gt}): We define:

$$(5) U_{gt} = (25\% U_p + 25\% U_{p-int} + 25\% S + 25\% E) E_{fec}$$

(5) means:

1. U_{gt} will be 100% only when U_p , U_{p-int} , S , E and E_{fec} all are 100%;
2. E_{fec} is a critical factor of U_{gt} , that is, if $E_{fec} = 0$, then $U_{gt} = 0$.

3.2.2 System navigation mechanism aptness

In a system that supports more than one CICI, a system level navigation mechanism is needed to provide a means to reach the desired CICI's. System navigation mechanism aptness (U_{nav}) is a composite use feature that signifies the comprehensive aptness in percentage of the navigation mechanism by taking into account a set of system navigation mechanism use features.

Because a system's navigation mechanism is just another goal-task, we can use the goal-task usability evaluation equations discussed above to assess its aptness; but because it is also unique compared to other goal-tasks in the system, the normal goal-task usability evaluation equation needs to be customized to fit this unique goal-task's special situation:

Effectiveness: A system navigation mechanism's effectiveness can be re-defined as reachability. Let's say the total number of CICI's that need to be accessible through the system navigation mechanism is N_t , the total number of CICI's that are actually accessible through the navigation mechanism is N_a , then the

$$\text{system navigation mechanism's effectiveness } E_{fec} = \frac{N_a}{N_t}.$$

Efficiency: Instead of time, a system navigation mechanism's efficiency is better considered in terms of human effort needed to reach a desired CICI through the system's navigating (organization) architecture. Specifically, here, the human effort means how many levels an end user has to click through a web site's navigating architecture in order to reach the desired CICI (the top level is counted as 1). Let's assume a fixed navigation CICI i has an access probability p_i , its reaching distance

through the system's navigating architecture is d_i , the total number of unique reachable fixed navigation CICI's is n , then the average probability reaching distance D_{ap} of the reachable fixed navigation CICI's can be defined as:

$$(6) D_{ap} = \sum d_i p_i$$

In order to have best efficiency, a system's navigating architecture is normally designed to have an optimal (minimal) average probability reaching distance.

Besides D_{ap} , another efficiency-affecting factor is the breadth of the navigating architecture. The breadth of a navigating architecture (W_{max}) is defined as the maximum number of items in the same level of any branch of the navigating architecture.

It is believed that a navigating architecture is most efficient when $D_{ap} = 1$ and that any navigating architecture with $D_{ap} \geq 5$ should be avoided [20][21][22][23][24][25]; It is also believed that the breadth of a navigating architecture has much less effects on the efficiency of the navigating architecture than its depth [26][27][28][29], but it is normally suggested that the breadth of a navigating architecture should be limited to less than 9 items [30]. In other words, an efficient navigating architecture should be shallow and wide but not too wide. The efficiency of a system's navigation mechanism can be defined as:

$$(7) E = 1 - (90\% \cdot \frac{d_e}{4} + 10\% \cdot \frac{w_e}{9})$$

$$\text{in (7), } d_e = \begin{cases} D_{ap} - 1; & \text{if } D_{ap} < 5; \\ 4; & \text{if } D_{ap} \geq 5; \end{cases}$$

and,

$$w_e = \begin{cases} 0; & \text{if } W_{max} \leq 7; \\ W_{max} - 7; & \text{if } 7 < W < 16; \\ 9; & \text{if } W_{max} \geq 16; \end{cases}$$

Use interaction process aptness: A system navigation mechanism's use interaction process aptness can be simply defined as the total-success ratio P_{ts} , that is, the number of end users who can finish the task divided by total weighted number of end users. Here, finish means an end user can locate a desired CICI through the system navigating architecture.

Use interaction interface and presentation aptness: its basic use feature can be simplified as:

Confusing-misleading-illegible navigating methods ratio ($P_{nav-methods-in}$): the number of confusing, misleading and/or illegible navigating methods in the navigating

architecture divided by the total number of navigating methods in the navigating architecture.

Layout and/or item-grouping inappropriate navigating pages ratio (P_{nav-lg}): the number of navigating (menus) pages in the navigating architecture that have inappropriate layout and/or navigating-methods-grouping divided by the total number of navigating (menus) pages in the navigating architecture.

So we define: $U_{p-int} = 1 - (50\% P_{nav-methods-in} + 50\% P_{nav-lg})$

3.2.3 Universal interface & presentation aptness

Universal interface and presentation aptness ($U_{uni-int}$) is a composite use feature that signifies the comprehensive universal aptness in percentage of all the interfaces and presentations in a system by taking into account a set of universal interface and presentation use features that universally apply to all the interfaces and presentations in a system. Its basic use features are good examples for those very super-facial but extremely important requirements for an entire web site. They all have nothing to do with the specific semantics of each particular page, but they are site level consistency rules, Web level conventions and standards which are so critical to the usability and look-and-feel of an entire web site that to which each page of a Web site MUST conform.

The universal interface and presentation conventions are:

Default set of global methods convention: The default set of global methods on each page include the most frequently used or most important global methods such as: home (not on homepage), a form of site level navigation mechanism, sitemap, immediate search, security terms, privacy terms, glossary and abbreviations list, help, page-refreshing (if the browser refresh button can not be used on the page), contact, institution info, log out, etc. These methods' naming, sizing, grouping and positioning must be consistent on all pages. For large pages, the default set of global methods on each page should be provided at both the top and the bottom (at the bottom, at least a method to return to the page top should be provided). For pages more than two screens long, at each folding position, a method to return to the page's top should be provided.

Link color-coding convention: All the links should be color-coded (if possible) following Web level convention. The color of unvisited link should be significantly contrasting to the color of visited link. Normally the brighter color means unvisited.

Time sensitive CICI timestamping convention: For time sensitive CICI, a most current update-timestamp must be provided throughout of it if appropriate. The format of the time stamp must conform to Web level convention.

Biggest-page-size convention: Page size means the sum of sizes in byte of all the files (except those files that are not intended to be downloaded all at once) belonging to a page's presentation. Biggest page size convention means for all the pages of a Web site, their page sizes must not exceed the allowed biggest page size.

Biggest page-generating time convention: Page-generating time is the time difference between the page-request receiving time and the page-response sending time on a web site. Biggest page-generating time convention means for all the pages sent from a web site, their page-generating time should not exceed the allowed biggest page-generating time.

Text-only version convention: For each page of a web site a text only version must be provided.

Broken links convention: All the pages of a Web site must not contain broken links.

Syntax conformance convention: All the pages of a web site must pass W3C syntax conformance check.

Universal accessibility conformance convention: All the pages of a Web site must pass W3C universal accessibility conformance check.

Site level consistency rules: For all kinds of links, UI-widgets (labels, buttons, etc.), there should be site level consistency rules. The same link or UI-widget that appears in more than one place should never be different except its necessary changes can be justified.

The basic universal interface and presentation use features (for each goal-task) are:

Default set of global methods convention violation pages ratio ($I_{v-methods}$): The number of pages that violated the default set of global methods convention divided by the total number of pages of a task.

Link color-coding convention violation ratio (I_{v-link}): The number of links that violated the link color coding convention divided by the total number of links on the pages of a task.

Time sensitive CICI timestamping convention violation ratio (I_{v-time}): For a task (CICI), $I_{v-time} = 0$ or 1.

Biggest-page-size convention violation ratio (I_{v-psz}): The number of pages that violated the biggest-page-size convention divided by the total number of pages of a task.

Biggest page-generating time convention violation ratio (I_{v-pgen}): The number of pages that violated the biggest page-generating time convention divided by the total number of pages of a task.

Text-only version convention violation ratio (I_{v-txt}): The number of pages that violated the text-only version convention divided by the total number of pages of a task.

Broken links convention violation ratio (I_{v-blk}): The number of links that violated the broken links convention divided by the total number of links on the pages of a task.

Syntax conformance convention violation ratio (I_{v-synt}): The number of pages that violated the syntax conformance convention divided by the total number of pages of a task.

Universal accessibility conformance convention violation ratio (I_{v-uacc}): the number of pages that violated the universal accessibility conformance convention divided by the total number of pages of a task.

Site level consistency rules violation ratio ($I_{v-consist}$): the number of links and UI-widgets that violated the site level consistency rules divided by the total number of links and UI-widgets on the pages of a task.

We define:

$$(8) U_{uni-int-gt} = 1 - (10\% I_{v-methods} + 10\% I_{v-link} + 10\% I_{v-time} + 10\% I_{v-ptime} + 10\% I_{v-pgen} + 10\% I_{v-txt} + 10\% I_{v-blnk} + 10\% I_{v-synt} + 10\% I_{v-uacc} + 10\% I_{v-consist})$$

Then,

$$(9) U_{uni-int} = \sum W_{gt_i} U_{uni-int-gt_i}$$

In (9), W_{gt_i} is the weight of the number i designed goal-task, and $\sum W_{gt_i} = 1$.

3.2.4 Quantitative web site usability

The quantitative web site usability (U) can be defined as:

$$(10) U = (W_1 (\sum W_{gt_i} U_{gt_i}) + W_2 U_{nav}) U_{uni-int}$$

In (10), U_{gt_i} is the quantitative usability of the number i designed goal-task; W_{gt_i} is the same as in (9); W_1 is the weight of the combined usability of all the normal tasks in a system; W_2 is the weight of the system navigation mechanism aptness. Depending on the system's complexity (total number of supported designed goal-tasks that have to be reached through the system's navigation mechanism), W_2 's possible values is between 0% and 30%, and $W_1 + W_2 = 1$.

(10) means that system usability U is a composite use feature that signifies the comprehensive overall usability in percentage of a system by combining together all its designed goal-tasks' usabilitys and the system navigation mechanism aptness and then taking into account the universal interface and presentation aptness as a critical factor that severely affects the overall usability of a system.

4 Discussion of the methodology

As a first endeavor to provide a quantitative full-lifecycle web sites usability engineering framework, this methodology is still at its infant stage, so any aspect of it is open for improvement. In order to avoid any confusion, we will give each set of quantified usability equations a version number. The

above set of quantified usability equations can be named as **Quantified Usability Equations SeT version 1.0 (QUEST v1.0)**. So, a system's quantified usability value should be stated together with the version number of the QUEST used to arrive at that value in order for it to make full sense. The format can be **Usability: U (Version#)**. For example, a system with 95% usability with version 1.0 of QUEST can be noted as Usability: 95 (QUEST v1.0) or 95% (QUEST v1.0).

Although a system's usability value itself is very meaningful already, it cannot tell it all. The best way to publish the usability information of a system is to list the quantified usability value together with major or all use features' values and their respective allocated weights that are behind it in a structured manner. This practice especially serves the system's usability engineering purpose.

By adopting this quantified usability methodology, we can anticipate the following benefits:

1. This quantified usability methodology is independent of, and therefore can be seamlessly integrated into, any engineering methodologies, processes and techniques.
2. Usability is structured, use centered and quantified and can be consistently applied to the full lifecycle of a system.
3. The usability of the definition of usability is improved.
4. Quantified usability requirements can be specifically and systematically demanded by end users to the user requirements acquirer at the user requirements acquisition stage of a project along with other user requirements. So, usability requirements have gained equal status with other user requirements.
5. Quantified usability requirements can be tested against to see if they have been satisfied.
6. Quantified usabilitys of different systems or the same system over time can be compared with each other.
7. Quantified usability provides a basis for usability improvement of a system over time.
8. This methodology is not only both formative and summative, but also both analytic and global [31].

5 Call for empirical studies

As future work, we will do some empirical studies of this methodology. We will also encourage students to try this methodology out in their usability engineering course projects. At the same time, we are cordially inviting any empirical studies on any aspects of this methodology from both academic and industrial world. Any comments and evaluations will be highly appreciated.

6 References

- [1] Netcraft.com: "October 2005 Web Server Survey", URL: http://news.netcraft.com/archives/web_server_survey.html
- [2] Miller, R. B. "Human ease of use criteria and their tradeoffs", IBM Technical Report TR 00.2185. Poughkeepsie, NY: IBM Corporation, 1971.

- [3] Shackel, B. "The concept of usability", Proceedings of IBM Software and Information Usability Symposium, September 15-18, 1981, Poughkeepsie, New York, USA, 1-30. Poughkeepsie, NY: IBM Corporation.
- [4] Shackel, B. "The concept of usability", In Bennett, J. L., Case, D., Sandelin, J., & Smith, M. (Eds.), *Visual display terminals: Usability issues and health concerns*, pp. 45~87, Englewood Cliffs, NJ: Prentice-Hall, 1984.
- [5] Shackel, B. "Ergonomics in design for usability", In Harrison, M. D. & Monk, A. F., *People and computers: Designing for usability*, pp. 44~64. Proceedings of HCI 86. Cambridge, UK: Cambridge University Press, 1986.
- [6] Shackel, B. "Human factors and usability", in Preece and Keller (Eds.), *Human Computer Interaction*, pp. 27~41. Prentice Hall, Hemel Hempstead, 1990.
- [7] Bennett, J. L. "The commercial impact of usability in interactive systems", In Shackel, B. (Ed.), *Man/computer communication: Infotech state of the art report, Volume 2*, pp. 1~17. Maidenhead, UK: Infotech International, 1979.
- [8] Bennett, J. L. "Managing to meet usability requirements: Establishing and meeting software development goals", In Bennett, J. L., Case, D., Sandelin, J., & Smith, M. (Eds.), *Visual display terminals: Usability issues and health concerns*, pp. 161~184, Englewood Cliffs, NJ: Prentice-Hall, 1984.
- [9] Shneiderman, B. "Designing the user interface: Strategies for effective human-computer interaction", 1st edition, Addison-Wesley, 1987.
- [10] Shneiderman, B. "Designing the user interface: Strategies for effective human-computer interaction", 2nd edition, Addison-Wesley, 1992.
- [11] Shneiderman, B. "Designing the user interface: Strategies for effective human-computer interaction", 3rd edition, Addison-Wesley, 1998.
- [12] Nielsen, J. "Usability Engineering", Boston: Academic Press, c1993. ISBN: 0125184050
- [13] Bevan, N. "Usability is quality of use", In Anzai, Y., Ogawa, K., & Mori, H. (Eds.), *Symbiosis of human and artifact: Future computing and design for human-computer interaction*, pp. 349~354. Amsterdam, The Netherlands: Elsevier.
- [14] Löwgren, J. "Perspectives on usability", 1995 IDA Technical Report LiTH-IDA-R-95-23, Linköping, Sweden: Department of Computer and Information Science, Linköping University.
- [15] Dix, A. J.; Finlay, J.; Beale, R.; Abowd, G. "Human Computer Interaction", 2nd edition, Prentice Hall, 1998.
- [16] Quesenbery, W. "What does usability mean: Looking beyond 'ease of use.'", Proceedings of 48th Annual Conference Society for Technical Communication, 2001.
- [17] Quesenbery, W. "Dimensions of usability", In Albers, M., & Mazur, B. (Eds.), *Content and complexity: Information design in technical communication*. Mahwah, NJ: Lawrence Erlbaum Associates, 2003.
- [18] ISO. "Ergonomic requirements for office work with visual display terminals (VDTs) -- Part 11: Guidance on usability", (ISO 9241-11:1998).
- [19] Norman, D.A. "Cognitive engineering", Chapter 3 in Norman and Draper ed., *User centered system design: New perspectives on human computer interaction*, pp. 31~61, Lawrence Erlbaum Associates, Hillsdale, NJ, 1986.
- [20] Dustin, E.; Rashka, J.; McDiarmid, D. "Quality Web systems: performance, security, and usability", Boston: Addison-Wesley, 2002.
- [21] Snowberry, K.; Parkinson, S.; Sisson, N. "Computer Display Menus", *Ergonomics*, 26, pp. 699~712, 1983.
- [22] Schultz, E.; Curran, P. "MENU STRUCTURE AND ORDERING OF MENU SELECTION: INDEPENDENT OR INTERACTIVE EFFECTS?", *ACM SIGCHI Bulletin*, Volume 18 Issue 2, pp. 69~71, October 1986.
- [23] Tolle, J.; Prasse, M.; Gott, R. "EFFECTS OF VARIATION IN MENU LENGTH AND NUMBER-OF WINDOWS ON USER SEARCH TIME", *ACM SIGCHI Bulletin*, Volume 18 Issue 3, pp. 73~74, January 1987.
- [24] Gonzales, J. "A theory of organization", Proceedings of the 12th annual international conference on Systems documentation: technical communications at the great divide, pp. 145~155, October 1994.
- [25] Larson, K.; Czerwinski, M. "Web page design: implications of memory, structure and scent for information retrieval", Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 25~32, January 1998.
- [26] Miller, D. P. "The depth/breadth tradeoff in hierarchical computer menus", Proceedings of the Human Factors Society, pp. 296~300, 1981.
- [27] Kiger, J. I. "The depth/breadth tradeoff in the design of menu-driven interfaces", *International Journal of Man-Machine Studies*, 20, pp. 201~213, 1984.
- [28] Jacko, J. A.; Slavendy, G. "Hierarchical Menu Design: breadth, depth and task complexity", *Perceptual and Motor skills*, 82, pp. 1187~1201, 1996.
- [29] Zaphiris, P.; Mtei, L. "Depth vs Breadth in the Arrangement Web Links", 1997, Available at <http://otal.umd.edu/SHORE/bs04/>.
- [30] Miller, G. A. "The magical number seven plus or minus two: Some limits on our capacity for processing information", *Psychological Review*, 63, pp. 81~97, 1956.
- [31] Scriven, M. "Evaluation thesaurus", 4th edition, Sage Publications, 1991.