Task-Based Evaluation Metrics for E-Commerce Sites

Abstract
Developers of e-commerce applications are often sceptical about Web-site usability guidelines. User testing is also usually not carried out because it is expensive in terms of time and expertise. The spectacular usability and commercial failure of some sites attest to the folly of such practices.

The main reason for developers neglecting current evaluation practices is that they are often vague, and in the case of user testing, too difficult to do effectively. This paper therefore offers an alternative.

The e-commerce shopping process has been analysed from a task-based point of view, and a set of task-weighted metrics to be used by developers in evaluating their sites has been proposed. These metrics have been applied to three sites and the results of the evaluation are given.

1 Introduction

Much work has been done on different aspects of E-Commerce (EC) Web site development in an effort to understand this relatively new phenomenon. While much has been published about the importance of usability of EC sites [17, 19, 25, 35, 37, 39] the current state-of-the-art in the EC industry shows that it is taking some time for the benefits of these findings to filter through to EC developers.

One of the reasons for this is that the results and guidelines being published may be too vague and difficult for Web developers to apply. Veen [38] refers to the difficulty of evaluating Web sites. Developers are often faced with an iterative and time-consuming evaluation process involving a number of users — until every perceivable problem is solved. We feel that the provision of a set of easily applicable metrics could make it easier for EC site developers to profit from accumulated research results and make the evaluation process a little less daunting.

The purpose of this paper is to provide an evaluation method for Web sites so that developers can have a metric by which to measure the quality of their EC site. In deriving such an evaluation method it was necessary to investigate the efforts of various researchers. Long before the advent of EC Ravden and Johnson [33] proposed a checklist-based evaluation mechanism which rates interfaces from various perspectives. These can be usefully applied to Web sites but need to be augmented for EC purposes. Some Web design guidelines appear on the Web — most notably those of the European Usability Support Centres (EUSC) [12] and IBM [18]. The EUSC guidelines are generalised Web-site design guidelines, and provide no specific EC advice. IBM’s guidelines are a very comprehensive and valuable set of instructions. They deal with subjects such as customer support, trust, product information, and visual layout. They do, however, deal with EC sites as a whole. They do not address
different phases of the EC shopping experience as being different and requiring different treatment — the one exception being the guidelines to be applied specifically to the checkout transaction.

The user is the most important facet of the EC shopping experience. Neglect the user at your peril! It is difficult for EC site developers, given a list of guidelines, to know which to follow. For example, developers are told to have the most important information visible to the user without scrolling [38]. They are also told to provide the user with enough information to keep him or her interested in the site — thus increasing the chance of a sale. In the face of differing dimensions and browsers how is the developer supposed to judge the relative importance of these conflicting instructions?

This paper will introduce a set of metrics which can be used by a developer to judge the quality of a site — enabling early detection and correction of problems. Section 2 takes a look at the users’ needs in using an EC Web site. To arrive at a meaningful set of metrics we analysed the purchasing session of the EC shopping experience. A model of this is given in Section 3. Section 4 then examines the consequences of a task analysis on the two phases of the purchasing session identified in Section 3. Section 5 identifies evaluation metrics which can be used by developers to refine their sites, and reports on the use of these task-weighted metrics to evaluate three typical EC sites. Section 6 concludes.

2 Understanding the Web Site

Norman [28] mentions three different concepts which exist when the human-computer interaction process is considered: the design model, the user’s model, and the system image. The user model will not be based on the design model (how the system really works), but rather on the system image — that which the user perceives. As the customers use a site for the first time they have to do two things:

1. try to understand the internal conventions of the site, which, if the developer has done things properly, will be consistent and familiar across the site. (The how and what of the web site.)

2. understand how to navigate across the site in order to achieve their goals. (The where of the web site.)

The former is represented in the user’s mind by a series of mental models, the latter by a mental map of the site. The following two sections discuss these concepts briefly.

2.1 Mental Models

When a user is using a Web site for the first time he or she has to build up an internal model of how the site works [38]. This can be referred to as the user’s mental model. “A mental model plays a central role in representing objects, states of affairs, sequences of events, the way the world is and the social and psychological actions of daily life” [22]. This model helps a user to make inferences and predictions and to decide on actions to be taken. All knowledge a user holds about the world is kept in the form of mental models.

The Web designer has to design the site with great care to expose the design model of the site to the user — to assist in the building of the mental model. Johnson-Laird [22] finds that mental models can only represent
indeterminancies if there is not an exponential growth in complexity. Sites with inconsistencies make things complex for the user. Veen [38] warns about the need for consistency within a site to avoid losing the user’s trust. Losing the user’s trust will affect the success of the site because if the user mistrusts the site he or she is hardly likely to purchase anything from it.

When two people have a conversation they build mental models about the subject of discourse as the conversation continues. The mental model could be composed of different interconnected models — the connections and models being built up as the conversation proceeds. If a mental model is being built visually a single model will be constructed in the user’s mind to reflect the results of the person’s visual perception. [22]

If we apply this to a Web site we have a combination of these paradigms. The interaction between the user and the Web site can be considered to be a conversation which mainly occurs visually [36]. The user cannot have a true discourse since there is no opportunity for the type of repair and correction which occurs routinely in conversation when a speaker judges comprehension to be insufficient or the listener expresses incomprehension. The Web site is inanimate and cannot be a conversational partner in the true sense of the word. The user needs to build a mental model by visual means and clues — and the resulting model is thus crucially dependent on what is visible on the screen.

2.2 Mental Maps

Veen refers to a special type of mental model called the mental map — the internal model the users build up which assists them in understanding how to navigate through the site. The user can be assisted by the Web designer by means of reminding the user of the connection to the tangible equivalent. The user always needs to know where he or she is, what is available and where he or she can go [38].

Current attempts at assisting the development of the mental map include Web site maps, tabs and local search engines [16]. Graphical site maps are useful when the purpose of the site is clear. The map provided by the MTN mobile phone network to allow users to send SMS messages, shown in Figure 1, seems to be perfectly adequate — but on closer scrutiny one notices that it does not reflect the current state of the site — the forum page is actually linked directly to the home page — and is not accessible from the forums. When one uses an image abstraction as a site map it is important to keep it updated.

When the site offers more than simple functions graphical maps are often very terse and it is difficult for new users to navigate intelligently based on such sparse information. Even worse is the trend to give the user lists and lists of Web page titles when the site map is requested — giving a Web site index rather than a map1.

Tabs, initially a good idea, are often over-used and are sometimes poorly implemented as site maps — Duck gives a taste of what an abuse of tabs could lead to (Figure 2). Guadiano and Kater [16] point out that these techniques require the user to navigate a complex map via terse page titles or keywords or to sift through numerous search results. They propose an intelligent user interface tool which helps users navigate through sites using natural language.

Figure 1: MTN's Site Map (http://www2.mtnsms.com/aboutus/guide.asp)

Figure 2: Dack's example of Over-used Tabs [6]
2.3 Facilitating Good Models

The designer has an almost impossible task in making the system image of a Web site explicit, familiar, intelligible and constant [28]. Various important factors need to be considered in order to assist the user in developing stable mental models and maps:

- **Conventions** — The developer needs to decide on a set of usages throughout the site, and to stick to them [38]. For example, the colour scheme is important — as is the background image [24]. It is important for users not to be confused because this breaks the mental model.

The inner site conventions should accord with external web conventions. To counter these is folly — the users trust will be broken and the site is likely to lose a customer.

- **Should feedback be given in textual or graphical format?** — Norman advises that sound and graphics should be investigated [29]. Faulkner, too, advises that feedback be presented in a graphical format and that all feedback messages should be clear and unequivocal [13]. Phillips [32] argues that visual imagery is superior to verbal representation in aiding memory and thinking. Gardiner [14] agrees, saying that recall is better for dynamically interacting items than for items stored in isolation. She avers that recall is further improved if items are presented pictorially, rather than textually.

From a cognitive point of view, a graphical site map may be far more helpful, since users have particular strengths which can be utilised by non-textual mechanisms such as processing visual information rapidly, coordinating multiple sources of information and making inferences about concepts or rules from past experiences [31].

- **Speed** — Users are easily put off by slow web sites [3]. When a site does not react within the expected time the user loses faith in the site — this breaks down mental models.

- **Navigation** — It is important for the construction of the mental map that the user be aware of where he or she is at any time. The judicious use of layout and navigational aids can make this easier for the user.

This list, while perfectly valid, is difficult for developers to follow since the guidelines it gives are fairly broad and all-encompassing — making them difficult to apply. The user will abandon a site if it does not meet certain standards which will ease the development of mental models and maps and thus enable the user to complete an EC task easily. We will provide a way for the developer to ensure that their site meets such standards. In setting up metrics for ensuring that users do not abandon sites in frustration it is necessary for us to understand the characteristics of the user's task when using an EC site. The user's task is seldom considered by the purveyors of published Web-usability guidelines. The following section reports on an analysis of the EC task and Section 4 explores the nature of the task execution during two distinct identified phases. This understanding is essential in developing metrics for evaluating EC sites.
3 E-Commerce Task Analysis

Researchers have examined the EC task from many different perspectives. We are interested in consumer behaviour because only by understanding human behaviour when interacting with EC systems can we determine the type of support which should be provided. Our previous analysis of the EC shopping experience can be consulted in [34] but we give an condensed version here for the sake of clarity.

![Diagram of E-Commerce Task Analysis]

**Figure 3: The Two Phases and Ten Stages of The Purchase Task**

The purchase task of the EC shopping experience can be split up into two distinct phases, as shown in Figure 3:

1. **LSD — Look, See and Decide.** This stage will typically be used to look at available products, compare them, and to make a decision about whether or not to purchase products. This may be done one or more times until the consumer has found products which satisfy his or her needs. This phase is intensely user-driven and has the following stages which can be traversed iteratively and in varying sequences:
   - *Welcome* • *Search* • *Browse* • *Choose*

2. **Checkout.** This phase ends the shopping session. When the user triggers this phase he or she has made a choice of the offered products and decided to purchase one or more products. They now have to provide certain details, such as their address and credit card details. They also have to make choices about things such as gift wrapping and shipping requirements. This phase changes the paradigm of the interaction process from user initiative to system initiative.

   This phase is typically composed of at least the following steps, which should be navigated in a serial fashion:
   (a) *User?* (b) *Where?* (c) *How?* (d) *Payment?* (e) *Sure?* (f) *Done.*

Below [2] examines the Web browsing and searching process and finds that three steps are involved:
1. asking a question,
2. constructing an answer, and
3. assessing the answer.

The user drives this process in the LSD phase, asking question after question until a particular item has been located. The process is exactly the same in the Checkout phase, except that the system drives the process — asking the user questions, receiving and assessing the answers. Most Web-design guidelines do not take these diametrically opposing operating paradigms into account.

4 Nature of Task Execution During Phases

A simple computer operating model may serve as an effective basis for an understanding of the goal-directed nature of an EC task execution. This model can also serve to further highlight the tasking difference between the two phases of the EC shopping process. A definition for task analysis that is suitable within the context of this application domain is that offered by Dix, Finlay, Abowd, and Beale [11] — they describe task analysis as the identification and description of the interactive system user’s problem space, in terms of domain, goals, intentions, and tasks. It can be expected that a task analysis should yield, inter-alia, the following benefits when applied to the EC interaction session:

1. It provides knowledge of the tasks that the user wishes to perform — i.e. find, select and purchase.
2. It is a reference against which the value of the system functions and features can be tested.
3. It is a cost-saving exercise because failure to allocate sufficient resources to the task analysis activity increases the potential for costly problems arising in later phases of development.
4. Task analysis makes it possible to design and allocate tasks appropriately and efficiently within the new system.
5. The functions to be included within the system and the user interface can be more accurately specified [23].

It is important to note that Diaper and Addison’s [9] claim that a task will be strongly dependent on the user’s mental model of the computer and (for this study, on-line), documentation. Task decomposition, knowledge based techniques and entity-relationship based analysis are three different but overlapping approaches to task analysis [11] and of these, the first two are especially relevant to this study. Knowledge-based descriptions of tasks are important because they identify:

1. the plan for carrying out the task,
2. the knowledge or concepts required, and
3. the interaction between different kinds of knowledge.
Figure 4: Task Analysis based on a Simple Iterative Computer Operating Model (adapted from [40])

The designer therefore gains understanding as to the commonalities that exist between tasks in terms of their knowledge requirements and their plans for execution [21]. The focus of this part of the study will be on identifying differences between the nature of the task during the LSD (which includes search-and-browsing, selection and the shopping cart), and the Checkout phases of the EC purchase interaction process.

4.1 A Model of Interaction for Task Analysis

An oft-encountered, and simple computer operation model, is presented in Figure 4\cite{40} — this specific example being adapted from Woodson, Tillman and Tillman\cite{40}, which is in turn based on the set of USA Military User

\footnote{The following should be noted about Figure 4 when specifically considering the EC task:

1. The S7 → S4 feedback loop labeled ‘automatic’ corresponds to Johnson’s\cite{20} notion of ‘procedures’ or well-practiced behaviours.

2. The error feedback loops S7 → S2 and S7 → S1 are included, as they are especially appropriate viewed within the unreliable communication context of Internet-based interaction. The ‘normal’ assumption of system deactivation only on completion of the task[s] is thus often untrue.

3. The original term used by Woodson et al.\cite{40} was entry. This was replaced by the term action in S4 to correspond to a five-level hierarchy composed of projects, tasks, subtasks, activities and actions — a set of these [i.e. multiple loopbacks] would be an activity.

4. The S8 → S2 loop represents the execution of different tasks within single system activation — it corresponds to the notion of a task-procedure or project.

5. S2 could be re-labeled as ‘goal adjustment’ if it is reached via an error feedback loop.

6. The S3 → S9 and S5 → S9 loops represent abandonment of the site — probably due to difficulty in understanding the system in the first case, or site response in the second case.}
Interface Design Guidelines. This type of model is eminently suitable to be used as the basis for formulating a strategy for task analysis. It is noted that Norman’s classical execution-evaluation model (as discussed by Dix et al. [11]), has stages effectively similar to this model.

4.2 Task Analysis for the LSD and Checkout phases

The nature of the shopping task during the LSD (Look-See- Decide) and Checkout phases are significantly different. The LSD phase is in essence a user-driven iterative browsing and selection task with (usually) less well-defined goals and a larger number of possible actions. The Checkout phase is a system-driven pre-defined, linear task with well-defined goals and sub-goals, and with a smaller number of predefined actions. During the LSD phase there will be three types of goals:

- browsing (search), for the shopping object,
- categorizing (search-result), the shopping objects, and
- specifying (selecting) a shopping object for the shopping cart.

The nature of the interaction is such that the customer should be kept interested in the results of the search-type goals, and keep them on-site — the focus is on discouraging user dropouts through abandonment of the goal or by linking them off-site [10]. System errors and poor response times during this phase are perceived to be less serious by the user (but not by the shop owner), but may result in shopper abandonment.

The checkout phase has a single goal — completing the financial transaction (as defined by the contents of their shopping cart and their shipping preferences) as quickly and securely as possible with the minimum of disruption and roadblocks [7]. Accordingly this phase has a set of linear and intentionally fairly rigid, sub-goals. The focus here is not on user engagement but on completing the transaction rapidly and securely — before shoppers change their minds about their shopping carts and the related cost [7]. This implies that response times, and clear feedback on reasons for delays, are more important here than during the LSD phase. Because errors could have a more serious (security and financial), impact, a well-designed user help function and clear explanatory sub-system are required. It is also easy to provide the user with obvious and intuitive navigation clues as to where they are in the process by using progress or stage indicators. The trend should be to strive for the minimum number of pages or stages — rather have the user scrolling moderately than clicking through to a larger number of small pages [7]. This is in stark contrast to having as much as possible of the relevant information immediately visible in the LSD stage [38]. Simplifying this process will ensure that there will be a smaller incidence of user dropout and shopping cart abandonment during this phase — provided additional costs such as shipping are shown as soon as possible. The effect of an error at this stage will affect the sequence of the sub-goals, and will make this phase non-linear (i.e. ‘loopy’).

When these aspects are applied to the model as presented in Figure 4 the following are obvious for the two phases:
1. More effort will be required for system activation (S1) in the LSD phase when compared to the checkout phase — for example, the customer has to have an established an Internet session. The customer also needs to know about the site. This part is well understood by marketing professionals and sites are often well advertised in the media. Unfortunately this level of attention is often not paid to other aspects of the EC experience.

2. Goal formulation (S2) will be less clear in the LSD phase as compared to the checkout phase — the customer may want to re-evaluate options and re-formulate goals based on the range, price, and availability of the shopping objects during the outcome of a set of search results, or promotional material.

3. The intermediate stages (S2 — S6) are less proscribed for the LSD phase, and there will be a natural tendency to loop back to S2 during this phase — for example, if the response (at S5) takes too long.

4. The S3 stage is often trickier for the user to formulate in the LSD phase. The user may have some vague idea of an item he or she needs, but have difficulty formulating a query. For example, the user may have heard about a popular autobiography by an Irish teacher who grew up in Limerick. The user types in many different search criteria — “Irish”, “teacher”, “Limerick” before perhaps finding the book Angela’s Ashes by Frank McCourt by browsing through the list of available autobiographies.

5. Interpretation of the response (S6) will be more difficult during the LSD phase when compared to the checkout phase — the customer may be presented with a range of shopping objects from which to choose compared to the linear progression during the checkout phase.

6. Measuring the success of the task at S5 will be more difficult for the LSD phase — the customer is dealing with a goal achievement based on an electronic description rather than confirmation of a familiar financial transaction as in the checkout phase.

7. The result of an error (which may be at S7 → S1, or S4 → S2) will be deemed to be less serious during the LSD phase than during the completion of a transaction in the checkout phase.

8. The transition S7 → S2 may be traversed during the LSD phase without an error having been made — it could happen as a result of a reformulated goal.

9. The transition S7 → S8 could be made as a result of abandonment even though the goal has not been achieved — since the site may not stock the required object.

10. The activity distance S2 → S7 should be as short as possible for the checkout phase with achievement of S7 always clearly visible, perhaps by means of stage indicators.

11. The S3 → S9 and S5 → S9 loops should be minimised by ensuring good site usability.
5 Task-Weighted Evaluation Metrics

We previously reported on a set of suitable evaluation metrics for measuring feedback-related usability of three EC Web sites [34]. The evaluation criteria used were equally weighted³. In certain cases, however, it may be advantageous to prioritize some of the criteria by means of a selective, unequal, weighting. Examples of how to do this may be found in Levi and Conrad [3]. They describe the application of Nielsen and Mack's [27] usability guidelines to the evaluation of a set of Web pages. After the evaluation they modify the list based on feedback from their two different (HCI and Web developers) evaluation teams, and produce a new list by assigning severity ratings to each usability violation found on a five-point scale. In addition they also prioritize on the basis of the frequency of occurrence of the usability problem. Their scale varies from 0: Not a usability problem, 1: Cosmetic, 2: Minor, 3: Major, to 4: Catastrophic problem. They produce a list of usability violations, which contains both frequency and severity information. Along the same lines, Bastien and Scapin [1] refer to the ‘the amount and importance of usability problems found’. Another technique applies a ‘strength of evidence’ scale to a set of evaluation criteria [26]. These criteria are based on the type and number of research experiments that may support, or discount, the specific criterion. W3.org [41] prioritizes in terms of (accessibility) guidelines that ‘must be applied’, ‘should be applied’, and ‘may be applied’.

The previous section discussed the differences between the two different phases of the EC shopping experience. It is fitting for the two phases to have different evaluation metrics as well — as befitting their different paradigms and needs — and for these metrics to be weighted according to their impact on usability. One suitable weighting approach, which we believe to be novel, is to assign a task weighting to each of these previously unweighted criteria scores. This task weighting has two components:

1. A task repetition component. The task repetition component is an indicator of how often this task or activity will be encountered during the interaction, and a weight of 0.1 will indicate it to be of low occurrence — implying that it only happens in exceptional cases, whereas a factor of 0.2 means that it happens very seldom, and a weight factor of 1.0 will indicate that this type of activity occurs regularly during the interaction. A value of 0 would indicate absence of the activity — indicating that it does not play a role in the calculation of the overall score.

2. A task complexity component. The task complexity component reflects the inherent degree of difficulty in executing the task or activity. A weight factor of 1.0 would imply that the activity is a highly complex task, requiring extensive background and operational knowledge, or indicating that it requires a high degree of complex interaction. A weight factor of 0.1 will indicate that the task is simple, with low interactivity. A weight value of zero is not possible because it would imply that no interaction is required.

Each of these two components can either amplify or attenuate the contribution of a specific usability feedback criterion to the overall score, and then yield a value, which we believe to be a truer reflection of the Web site’s overall usability than that rendered by our previous metrics.

³ It should be noted that in effect we did prioritize — we made a careful selection of the criteria used from the larger Ravnseth set, and then we used a binary weighting based on ‘good’ and ‘red-flagged’ results.
5.1 Evaluation of EC Sites

We chose the same three sites to apply the metrics to as were used in the previous paper [34] so as to enable us to effect a comparison. We purchased various products from each EC site, and evaluated the process using the metrics given in Tables 1 and 2. A score is given for each of the questions as follows:

- Never (0) — the feature is never available.
- Sometimes (1) — the feature is seldom there.
- Mostly (2) — the feature is usually there.
- Always (3) — the feature is universally available.

The scores are then determined per stage, and per site in the form of a percentage. The scores per feature in each stage were calculated by adding up the score for each page making up the stage and awarding a total for each particular feature. The scores for each feature were then totalled to arrive at a percentage per site per purchasing stage to arrive at a raw score.

The results of applying this task-based weighting to the previously-reported results are shown in Tables 1 and 2, for the LSD and Checkout phases respectively. In these two tables the original (raw) score is weighted by the two task components which is shown in brackets as a value between 0.1 and 1.0. The final ratio value shown is the upper normalized percentage ration to the site with the highest score — the previous, unweighted or raw values are shown in brackets [34]. Of particular interest in these two tables, would be criteria with associated task or activities that have either high combined (i.e. repetition and complexity (R+C)), weight factors, or very low combined weight factors.

<table>
<thead>
<tr>
<th>Evaluation of Feedback Quality - LSD Stage</th>
<th>Amazon Max 18</th>
<th>Kalahari Max 18</th>
<th>BOL Max 18</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Is it clear what a user must do to search for a product?</td>
<td>6(0.8+0.5)=7.8</td>
<td>6(0.8+0.5)=7.8</td>
<td>7(0.8+0.5)=7.8</td>
</tr>
<tr>
<td>2 Does the search engine offer alternatives if a search fails?</td>
<td>5(0.8+0.5)=6.5</td>
<td>5(0.8+0.5)=6.5</td>
<td>6(0.8+0.5)=6.5</td>
</tr>
<tr>
<td>3 Does the system inform the user of the reasons for delays?</td>
<td>5(0.8+0.5)=6.5</td>
<td>5(0.8+0.5)=6.5</td>
<td>6(0.8+0.5)=6.5</td>
</tr>
<tr>
<td>4 Are different types of information clearly separated?</td>
<td>7(0.8+0.3)=11.2</td>
<td>6(0.8+0.3)=10.2</td>
<td>7(0.8+0.3)=11.2</td>
</tr>
<tr>
<td>5 Is it clear what needs to be done to select a product?</td>
<td>5(0.8+0.5)=5.4</td>
<td>5(0.8+0.5)=5.4</td>
<td>6(0.8+0.5)=5.4</td>
</tr>
<tr>
<td>6 Can the user undo a product selection?</td>
<td>9(0.1+0.5)=5.4</td>
<td>6(0.1+0.5)=5.4</td>
<td>9(0.1+0.5)=5.4</td>
</tr>
<tr>
<td>7 Is it clear what must be done to make the transition to Checkout?</td>
<td>6(0.1+0.5)=3.6</td>
<td>0(0.1+0.5)=0</td>
<td>9(0.1+0.5)=5.4</td>
</tr>
<tr>
<td>8 Does the system allow users to check on previous searches?</td>
<td>0(0.1+0.5)=0</td>
<td>0(0.1+0.5)=0</td>
<td>0(0.1+0.5)=0</td>
</tr>
<tr>
<td>Percentage: (Raw) Task-weighted</td>
<td>(73%)</td>
<td>(57%)</td>
<td>(64%)</td>
</tr>
<tr>
<td>Normalized Percentage Ratio: (Raw) Task-weighted</td>
<td>46.9/144=33%</td>
<td>31.5/144=22%</td>
<td>35.1/144=24%</td>
</tr>
</tbody>
</table>

Table 1: Results for the task weighted feedback evaluation: LSD Stage

The results from Table 1 show that:

1. Applying the task weighting has emphasised the usability difference between Amazon and BOL/Kalahari.

---

4Amazon ([www.amazon.com]), Books Online ([www.uk.bol.com]) and Kalahari([www.kalahari.net])
<table>
<thead>
<tr>
<th>Evaluation of Feedback Quality - Checkout Stage</th>
<th>Amazon Max 36</th>
<th>Kalahari Max 18</th>
<th>BOL Max 30</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Are instructions and messages concise, clear and unambiguous?</td>
<td>12(1+0.8)=21.5</td>
<td>10(1+0.8)=18.6</td>
<td>11(1+0.8)=19.8</td>
</tr>
<tr>
<td>2 Are possible actions clear?</td>
<td>12(0.8+0.8)=20.2</td>
<td>12(0.8+0.8)=20.2</td>
<td>12(0.8+0.8)=20.2</td>
</tr>
<tr>
<td>3 Is the required format of user inputs clearly indicated?</td>
<td>15(0.5+1)=22.5</td>
<td>7(0.5+1)=8.5</td>
<td>10(0.5+1)=15</td>
</tr>
<tr>
<td>4 Are user actions linked to changes in the interface?</td>
<td>13(0.8+0.5)=18.9</td>
<td>10(0.8+0.5)=18.0</td>
<td>12(0.8+0.5)=19.6</td>
</tr>
<tr>
<td>5 Is there always an appropriate response to user actions?</td>
<td>12(0.8+0.8)=19.2</td>
<td>10(0.8+0.8)=18.0</td>
<td>13(0.8+0.8)=20.8</td>
</tr>
<tr>
<td>6 Does the system inform the user of the success or failure of their actions?</td>
<td>14(1+0.5)=21</td>
<td>11(1+0.5)=16.5</td>
<td>11(1+0.5)=16.5</td>
</tr>
<tr>
<td>7 Does the system inform users of the reasons for delays?</td>
<td>11(1+0.5)=16.5</td>
<td>8(1+0.5)=9.5</td>
<td>5(1+0.5)=6.5</td>
</tr>
<tr>
<td>8 Do error messages indicate the what, where, and why, and how to recover?</td>
<td>9(0.5+1)=13.5</td>
<td>4(0.5+1)=6</td>
<td>11(0.5+1)=16.5</td>
</tr>
<tr>
<td>9 Is it clear what the user has to do to complete the task?</td>
<td>11(0.5+0.8)=14.3</td>
<td>5(0.5+0.8)=6.5</td>
<td>14(0.5+0.8)=18.2</td>
</tr>
<tr>
<td>10 Does the system indicate the current stage?</td>
<td>17(0.3+0.5)=13.6</td>
<td>10(0.3+0.5)=9.4</td>
<td>15(0.3+0.5)=16.5</td>
</tr>
<tr>
<td>11 Can the user easily back out of the process?</td>
<td>19(0.1+0.5)=13.9</td>
<td>10(0.1+0.5)=5.4</td>
<td>15(0.1+0.5)=9</td>
</tr>
<tr>
<td>12 Is final purchase confirmed by the user?</td>
<td>19(0.1+0.5)=13.9</td>
<td>10(0.1+0.5)=5.4</td>
<td>15(0.1+0.5)=9</td>
</tr>
<tr>
<td>13 Can users check on inputs provided during the process?</td>
<td>9(0.1+0.5)=5.4</td>
<td>6(0.1+0.5)=3.6</td>
<td>3(0.1+0.5)=1.8</td>
</tr>
<tr>
<td>Percentage: (Raw) Task-weighted</td>
<td>88%</td>
<td>59%</td>
<td>72%</td>
</tr>
<tr>
<td>Normalized Percentage Ratio: (Raw) Task-weighted</td>
<td>0.92</td>
<td>0.82</td>
<td>1.0</td>
</tr>
</tbody>
</table>

Table 2: Results for the task weighted feedback evaluation: Checkout Stage

2. High (R+C) criteria include activities associated with the presentation of information, and instruction-oriented actions.

3. Low (R+C) criteria include product selection and de-selection actions.

The results from Table 2 show that:

1. The task weighting has improved the score of Amazon, but decreased that for Kalahari, both relative to the BOL score. The extent of this is more obvious when studying the normalized percentage ratio values in Table 2.

2. High (R+C) criteria include user guidance, appropriate responses and the clarity of interaction messages and information presentation.

3. Low (R+C) criteria include the confirmation of the purchase and error indicators.

The results and especially the approach adopted, namely that of prioritizing certain criteria over another set is believed to be the correct approach. On an intuitive level, it is clear that repetition of a task should make it more important (i.e. increase its weight), that the level of interaction required should also increase its weight, that the task duration should increase its weight, and the level of knowledge required for the task should also increase its contribution to the website’s overall usability score. A more formalized analysis for task repetition values could be obtained through a frequency count of the Knowledge Representation Grammars (KRGs) and Sequence Representation Grammars (SRGs) produced from a Task Analysis for Knowledge Description (TAKD) analysis [8]. In contrast the method used here was to simply count the occurrence of these during a typical (shopping and browsing) interaction session.
6 Conclusion

This paper extends previously reported work on EC Web site evaluation metrics. We report on a task-based weighting scheme for evaluating these sites. This extension refines our previous evaluation metric scheme and provides a more finely tuned mechanism for assisting developers to improve usability of EC Web sites.

Our guidelines will be more helpful than many of those published on the Web, since we also considered the user’s task in the formulation of the guidelines. In conjunction with this, we have made use of a novel usability metric prioritising scheme to yield information which is noticeably different from unweighted values.

References


