Introducing Metrics for Evaluating Usability Aspects
of E-Commerce Sites

Karen Renaud, Paula Kotzé & Tobias van Dyk
University of South Africa
Email: {renaudk, kotzep, vandykt}@unisa.ac.za

ABSTRACT
The developers of e-commerce applications have a problem in gauging the knowledge and expertise of end-users. The developer must therefore design the interface to the system so that the use thereof is intuitive enough to require the minimum of background knowledge. Developers need to enhance the usability of their web sites so that incidental users will find the site sufficiently rewarding and enticing enough to encourage them to explore further, this being one of the primary aims of e-commerce sites since it increases the likelihood of online sales.

This paper will explore the role of feedback as a valuable tool in enhancing the interpretability of e-commerce applications. We discuss firstly how applications may be designed to make use of an enriched model of application feedback, and secondly how developers may evaluate their sites so as to gauge the efficacy of the currently provided feedback.

We propose a mechanism for analysing the purchasing phase of the e-commerce experience. We then introduce an evaluation method for determining whether a particular site provides adequate feedback or not. Three sites were evaluated using the proposed model, and the results of this evaluation are analysed.

Keywords
Feedback, Metrics, Evaluation, E-Commerce.

INTRODUCTION
E-Commerce (EC) has tremendous potential and can benefit both seller and buyer. Buyers are no longer restricted to shopping at certain times or within certain geographical locations [32]. Jahng et al. [21] track the changes in retailing all the way from the small family store to the current shopping malls with plenty of choice and limited service. They claim that EC is the next step in this natural progression. According to Rohn [51], there are two important things an EC site should do in order to be successful:

- attract additional customers, and
- reduce the workload for the sales force.

Keeping current customers might be even more important [20]. EC offers such a wide range of choice that EC purveyors can never become complacent. EC customers can defect whenever they feel that they can get better prices or service elsewhere [19, 35]. Customer loyalty is of prime importance in EC [21], but it is not easily earned. EC could possibly become the new retailing paradigm, offering customers the levels of service last seen in the days of the family stores — one-to-one service whenever it suits them — while giving them the level of choice generally offered by modern shopping malls.

Developers of EC applications necessarily target a much wider audience, albeit an untrained and unknown one, than was previously the case [31, 51]. Developers therefore have to develop new skills and take different needs into account when developing these systems [19, 36]. Researchers have started trying to define this new shopping paradigm and also to understand EC customer behaviour [7, 27, 33, 52, 55, 60, 63]. One fact that comes to light is that those customers who choose to do their shopping online are attracted by the convenience of this new mechanism [4, 20, 22, 60, 64]. Research has indicated that online customers will seldom stay around to investigate a site that puzzles them, or is difficult to learn to use. They tend to simply switch to another page in another site [35, 41].

Many researchers rate ease of use as being of critical importance to the EC process [33, 60, 65]. Poor usability is often blamed for the failure of sites [20, 51]. Helander argues that the success of EC is directly related to ease of use [20]. Liu et al. [29] found that the success of EC sites can be related to the quality of information they provide, system use, playfulness and system design quality — essential usability features. Jahng et al. [21] argue that the uncertainty and ambiguity of badly designed sites
make users feel uncomfortable, something which is unlikely to result in a sale. Usability can be tested in three ways:

1. **asking users.** This approach is followed by Tilson et al. [60], Liu and Arnett [29], and Kirakowski and Cierlik [26]. There are always problems with conducting user surveys, however, as pointed out by Nielsen [39], because people do not always tell the truth.

2. **scrutinising logs.** This approach is applied by Rosenstein [52], but logs are notoriously difficult to interpret [35]. Nielsen [40] argues that the acid test of EC sites should be the number of successful transactions completed. However, in the light of what has previously been said about people’s behaviour when confronted with a badly designed site, one wonders whether this type of retrospective analysis will be timely enough.

3. **applying usability metrics.** We argue here that this approach should be followed as a first step in ensuring usability of a site. In traditional user interfaces one is able to involve the end-users in the development of an interface, but, in the case of EC sites, one usually has to publish the site and risk losing customers before knowing about problems [12]. Tieman et al. [59] emphasise the importance of first impressions to users. Kavanagh [24] cites a study which found that 80% of users fail to return to a site.

To minimise the risk of losing users permanently due to poor usability, one should apply usability metrics and *only then* use a combination of the previous two methods to keep tabs on the usability of the site.

In this paper we have chosen to discuss one particular aspect of web-site usability, namely that of feedback. We will advocate the extensive use of feedback to increase the interpretability of systems, thus enhancing the ease of use of these systems. The following section will explore the nature of feedback in EC applications. It is necessary, however, to convert this discussion about the merits of feedback to a methodology for evaluating proposed EC web site pages. Before such a methodology can be provided it is necessary to understand the nature of the EC shopping experience, and this will be done next. We then propose a feedback evaluation methodology for EC systems. This methodology is evaluated by applying it to three large EC sites.

### FEEDBACK

The human-computer interaction feedback perspective has an interest in ‘the exchange of information between participating agents through sets of channels, where each has the purpose of using the exchange to change the state itself or one or more others’ [57]. The feedback thus concentrates on the method and type of interaction, the participants in the interaction, their purpose, and the interface between the human and the computer [56].

Feedback therefore serves a behavioural purpose in interaction, so that the computer fulfills the same role as a *conversational participant* [47, 58]. Only by means of feedback can participants in a conversation detect faults in the understanding of what is said [13]. The success of human-computer ‘conversation’ then, will depend on the human being able to gauge the ‘knowledge’ of the application. This is especially true in the case of the conversation being interrupted. Feedback must therefore make the ‘knowledge’ of the application, based on previous inputs, tangible and accessible in order to fulfill its role adequately in the face of an untaught and unreliable user population.

Feedback is therefore a valuable tool in the hands of an application developer, who needs to communicate the application’s ‘knowledge’ (what has already been provided) and ‘expectations’ (what the application needs the user to provide) to facilitate the application’s role as conversational participant.

There are two different schools of thought with respect to the motivation behind user actions. The artificial intelligence branch of computer science is based on the existence of underlying plans influencing user actions [30]. The alternative to this rationalistic perspective, as proposed by Suchman, is that action is inherently *situated* — with plans having a limited prescriptive effect on user actions [58]. Suchman’s *situated action* view is that users react to their circumstances, with an objective in mind, rather than slavishly following some set of plans.

The conversational model of user interaction, with respect to the current computer usage paradigm of recognition rather than recall [11], leans towards the Suchman’s situated action perspective, rather than the plan-based mode of operation. Users thus react according to the way they interpret the state of the system. Because of this, the user’s understanding of the EC application will not be based on the model as designed by the system developer, but rather on the
image portrayed by the system as the user makes use of it. The designer thus has a difficult task in making this system image explicit, intelligible and constant [42].

The quality of the feedback provided by the system can go a long way towards narrowing the gulf of evaluation — thereby enabling an understanding of the state of the system. Feedback becomes very important when the system is prone to long response times, which often happens in highly distributed systems.

**USE OF FEEDBACK**

The previous section motivated feedback, and discussed issues pertaining to the role of feedback in the human-computer conversation. It is necessary, before deciding on the type of feedback to be provided, to consider the purpose of any feedback, and the way a user can be expected to make use of such feedback as is provided. The OED [54] defines feedback as:

1. signifying a response,
2. modifying the behaviour of the user, and
3. promoting understanding.

The traditional role of feedback in human-computer interaction is often seen exclusively as pertaining to the first use. The extension of the feedback concept to include all the features will enable EC sites to give better and adequate feedback to users.

It is difficult to provide a general rule about the exact nature of feedback since it is directly dependent on the nature of the task. Some feedback is fairly standard, such as the highlighting of text to indicate selection. However, other feedback needs are not nearly as straightforward, and the developer may not have existing guidelines to be followed. An excellent example of this is the diverse treatment accorded to error management.

It has been noted by various researchers that a discourse typically has an incremental quality about it [6, 28, 31]. Dix [11] argues that it is difficult for users to manage and visualise this ‘sense of history’ in their interaction with the computer, especially since the current interface is based more on recognition than recall. This historical need was also noted by Tweedie [61], who argues that past input and output should be linked so that all historical input and output relationships can be explored directly. This is echoed by Shneiderman [33]. Often the application’s only concession to a user’s need for this is the provision of an undo facility. This provides a type of historical function but the user often needs to reverse the state of the system in order to see what happened before. When a user is interrupted the undo function is useless in assisting the re-establishment of context.

In addressing the question of the type of feedback to be provided, it is therefore appropriate to consider the need for the portrayal of previous system states, so that the user can refer to it in order to understand the present state of the system. Researchers have noted the usefulness of the historical functions routinely provided by web browsers such as a history of sites visited and bookmarks [66] but we feel that this history should be more finely grained than that. The user needs to have a history of their interaction within particular web sites. Therefore, we believe that there is a need to keep a history of the user’s actions, and relate it to the system’s response. This type of information could be referred to as archival feedback, as opposed to immediate feedback which communicates the present state of the system. Archival feedback provides the facility often used in conversation when a person refers to a previous statement, and then builds on it.

We postulate that good feedback should involve giving the user both immediate and archival feedback. The following section will take a look at the difficulties of providing such feedback.

**Difficulties in Providing Adequate Feedback**

While the provision of historical details with respect to sites visited is routinely done by web browsers, the more detailed historical details will have to be provided by the EC sites themselves since it is too finely grained, and linked to the current state of the EC transaction, to be provided by the browser. In providing this level of feedback one encounters several undeniable difficulties:

1. Developers of components, e-commerce systems and web interfaces are often technically oriented and this can make them unrealistic about the abilities of their end-users [25]. Norman [43] puts it very succinctly by stating that ‘there is no return to innocence’. There is a mismatch between developer assumptions and actual user knowledge. This gulf has to be bridged effectively if feedback needs are to be met.
The developer will often make assumptions which are unfounded, in total innocence. An example of this is the frequent use of words like *cookie* and *secure server* on most EC sites. It is unreasonable to expect the average EC user to understand what these words mean [64]. It is far better to simply tell the user that their interaction with the system will be secure — there is no need to inform the user about the facilitating technology.

2. The distributed nature of EC systems makes the goal of a trouble-free and enjoyable EC shopping experience that much harder to attain. There is always the possibility that some parts of the system will not be available [2, 34], either because of a network problem or because one of the computers has crashed. Users will be puzzled by sudden absences of trusted sites and need to be apprised of the reasons for them.

3. The nature of their users. Most users of EC systems will not have been trained in their use. The user interface will therefore have to be designed with great care. The user must be able to discover everything about the system, based on the *perceptible system state*. The designer of the user interface must be sure to bestow rational behaviour on the application — ensuring that the application behaves in a way that is reasonable and intelligible. When a user makes use of an application, the application must give the impression of being responsive to user actions, in the same way as humans are responsive to other humans' actions [58]. Friedman and Millett [15] found that people, even computer literate people, attributed social attributes to computer technology. The user must therefore be able to understand the motivation behind the system's actions — as being directly in response to his or her own actions.

**Recommendations**

It is necessary to translate the discussion of this section into some set of recommendations so that developers have guidelines to follow in order to ensure that adequate feedback is provided. The section has thus far introduced the idea of a classification of feedback according to the timeliness thereof, and talked about the problems of providing such feedback. To end off this section we conclude that the developer needs to provide the following:

- **Immediate feedback** — signal a response to each user action, explain unusual occurrences such as delays, display relevant system state clearly and don't use jargon.
- **Archival feedback** — provide a historical function which allows users to check on previous actions without changing the state of the system. Always provide the facility for users to check on their progress through any process which has specific stages.

The reader will note that this combined feedback satisfies the OED definition of the purpose of feedback. However, in working towards a set of usable guidelines for developers, the next step in this research is to develop a model of the EC shopping process. Only once such a model has been derived can we arrive at an EC-specific set of guidelines for determining feedback quality. The next section provides such a model, and uses this model and the findings of this section to derive the required guidelines.

**Analysis of the E-Commerce Purchasing Experience**

Researchers have examined the e-commerce 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 feedback which should be provided. Guttman *et al.* [18] identify six stages of customer purchasing behaviour: need identification, product brokering, merchant brokering, negotiation, purchase and delivery, and service and evaluation. O'Keefe and McEachern [45] propose a model with only five processes: need recognition, information search, evaluation, purchase, and after-purchase evaluation. Singh *et al.* [55] break up the EC process into three activities: indentifying and finding a vendor, purchasing and tracking.

Miles *et al.* [33] have examined the early phases of these models in detail, focusing on the decision making processes which lead a user to decide on a specific product and merchant. We would also like to examine only one of these processes — namely the one that everyone refers to as the *purchase* task. This task can be split up into two distinct stages, as shown in Figure 1:
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 because the user is looking at and assimilating information continuously. The system merely attempts to support the searching and browsing process so that it renders the products the user is looking for. This stage has the following substages which can be traversed iteratively and in varying sequences:
   - Welcome;
   - Search;
   - Browse;
   - Choose.

Heland and Khalid [20] refer to Johnson et al. [23] who argue that a cost can be allocated to this phase. Customer welfare increases as the cost reduces. Jarvenpaa and Todd [22] also emphasise the importance of easing this process. Since most customers are primarily interested in convenience this process should be streamlined, efficient and informative.

2. **Checkout.** This stage ends the shopping session. When the user triggers this stage he or she has made a choice of offered products and decided to purchase one or more products. Users 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. In contrast to the LSD stage, this stage is system-driven. The system needs to get certain information from the user in order to process the transaction, and this phase changes the paradigm of the interaction process from user-driven to system-driven. This is one of the biggest reasons why feedback is of critical importance during this stage — users who feel that they have lost control can simply leave the site without any embarrassment — unlike a user standing at a checkout till in a supermarket.

This stage is typically composed of at least the following steps, which should be navigated in a serial fashion:

(a) User? — user identification.
(b) Where? — specification of an address for delivery and invoice.
(c) How? — specification of the desired delivery method.
(d) Payment? — credit card, cheque or other fund transferral.
(e) Sure? — final confirmation from the user that the transaction should proceed.
(f) Done — a visual confirmation of the transaction.

Only one of the above steps has elicited much research interest — number 2d above — one finds many studies which have commented on consumer mistrust of security mechanisms on
the internet [21, 60, 63]. Since this risk is more perceived than real, Jarvenpaa and Todd [22] suggest alleviating these fears by providing customer testimonials and emphasising security measures. Some EC sites offer to carry all costs should security problems occur (Figure 2). In conclusion, once a customer trusts any site enough to purchase a product the process should be made as smooth as possible.

Amazon.co.uk Safe Shopping Guarantee

We guarantee that every transaction you make at Amazon.co.uk will be 100% safe. This means you pay nothing if unauthorised charges are made to your credit or debit card as a result of shopping at Amazon.co.uk.

Learn More

Figure 2. Payment Guarantee

Many customers simply like to browse through the available products without necessarily buying each time they connect to a site [60]. The ease with which they can do this will affect their decision to proceed to the following stage [20]. When customers do decide to proceed to the following stage, and check the item(s) out, poor usability could easily cause them to reverse their decision. However, the feedback needs of these two stages are sufficiently different to require them to be considered separately, as will be done in the following section.

METHOD

The two phases mentioned in the previous section have very different operating paradigms. The feedback and information requirements of a user during the LSD phase are very different from those required by the checkout stage.

- The LSD stage is often unstructured, iterative and information intensive. During the LSD stage, the user will often navigate principally by means of point and click. The user thus gathers information about various products and makes a choice of those products. This stage is also characterised to a large extent by a situated-action paradigm of navigation. The user will react directly to what each search reveals, and according to some inner set of norms with respect to product preference and price.

- The Checkout stage must perform be structured in a particular way, is linear, and requires far more inputs from the user. This phase requires a user to provide various information, in a pre-determined and structured manner, and signifies the end of the shopping process. This stage is so similar to the checkout process in a normal retail store that customers usually have no problems with the general concept of an EC checkout — each set of required inputs is so predictable as to be non-threatening. The manner in which the checkout stage is implemented, bearing in mind customer concerns about security and the fact that trust needs to be earned in EC, is often less than optimal.

The differences in operation between these phases makes it beneficial to develop two different strategies for evaluating web pages making up each of these stages separately since some sites will support one stage far better than the other and one single evaluation process is unlikely to suit both stages.

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 feedback of their EC site. In deriving such an evaluation method it was necessary to combine the findings of various researchers. Raven and Johnson [49] propose a checklist-based evaluation mechanism which rates interfaces from various perspectives such as visual clarity, consistency, compatibility, explicitness, appropriate functionality, informative feedback, flexibility and control, error prevention and correction and finally user guidance and support.

Some web design guidelines appear on the web — most notably those of the European Usability Support Centres (EUSC) [16] and IBM [17]. The EUSC guidelines are generalised web-site design guidelines, and provide no specific e-commerce 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 e-commerce sites as a whole. They do not address feedback as a separate issue and do not apply different guidelines to the different stages identified in the previous section — the one exception being the guidelines to be applied specifically to the purchase transaction.

We have selected suitable elements from most of Ravden and Johnson’s categories in order to set up one complete feedback evaluation mechanism, for each stage, which will ensure that an EC page provides adequate and complete feedback. The evaluation metrics for the LSD phase are shown in Table 1. The questions are paired with references to papers published about this aspect of human-computer interaction or EC systems in particular. The last column indicates the purpose of the feature in the interface as follows:

1. R — signifying a response,
2. M — modifying the behaviour of the user, and
3. U — promoting understanding.

The metrics for the checkout phase are somewhat different, reflecting the linear and structured nature of the checkout process. The metrics are given in Table 2.

The following section will describe how these metrics were applied to three EC sites, and comment about the efficacy of the proposed evaluation mechanism. In order to evaluate EC web pages, a score is given for each of the above 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 where 100% indicates a site giving a user perfect feedback and sites scoring 0% might as well give up and close shop. 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.

The evaluation criteria used were equally-weighted. In certain cases it may be advantageous to prioritize some of the criteria through a heavier, unequal, weighting. Examples of how to do this may be found in Levi and Conrad [9]. They describe the application of Nielsen and Mack’s [1] usability guidelines to the evaluation of a set of Web pages. After the evaluation they then 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 Scalin [3] 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 [62]. These criteria are based on the type and number of research experiments that may support, or discount, the specific criterion. W3.org [8] prioritizes in terms of guidelines that ‘must be applied’, ‘should be applied’, and ‘may be applied’.

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

EVALUATION

In order to test the efficacy of the proposed evaluation metrics we chose three sites to apply the metrics to. In choosing the sites we tried to find sites which sold similar products so that the evaluation would be comparatively meaningful. Booksellers like Amazon (www.amazon.ac.uk) are the pioneers in this field and we felt that their site would be a good one to evaluate. We therefore chose two other bookseller’s sites to compare it to — namely Books Online (www.uk.bol.com) and Kalahari (www.kalahari.net). We purchased various products from each EC site, and evaluated the process using the tables given in the previous section. Our final scores for each site, arrived at after discussion and consensus of all the authors are given in Tables 3 and 4.
<table>
<thead>
<tr>
<th>Evaluation of the System Feedback Quality — LSD Stage</th>
<th>Ref</th>
<th>Feedback Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is it clear what a user must do to search for a product?</td>
<td>[20, 33]</td>
<td>U</td>
</tr>
<tr>
<td>If the search engine fails to find a match, does it offer alternatives so as to prolong the interaction process?</td>
<td>[21]</td>
<td>M/R</td>
</tr>
<tr>
<td>Does the system inform the user of the reasons for delays?</td>
<td>[38, 37]</td>
<td>U/R</td>
</tr>
<tr>
<td>Are different types of information clearly separated?</td>
<td>[5]</td>
<td>U</td>
</tr>
<tr>
<td>Was information always readily available?</td>
<td>[20, 46]</td>
<td>U</td>
</tr>
<tr>
<td>Is it clear what the user must do to select a product?</td>
<td></td>
<td>U</td>
</tr>
<tr>
<td>Can the user easily undo a product selection?</td>
<td>[38, 37]</td>
<td>U</td>
</tr>
<tr>
<td>Is it clear what the user must do to make the transition from LSD to Checkout?</td>
<td></td>
<td>U</td>
</tr>
<tr>
<td>Does the system allow users to check on previous searches?</td>
<td>[50]</td>
<td>M</td>
</tr>
<tr>
<td>Is jargon and terminology user-centric?</td>
<td>[38, 37]</td>
<td>U</td>
</tr>
<tr>
<td>Is there a help facility?</td>
<td>[60, 46]</td>
<td>M/U</td>
</tr>
</tbody>
</table>

Table 1
EVALUATION METRICS FOR THE LSD STAGE

<table>
<thead>
<tr>
<th>Evaluation of the System Feedback Quality — Checkout Phase</th>
<th>Ref</th>
<th>Feedback Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Are instructions and messages displayed by the system concise and positive?</td>
<td>[35]</td>
<td>U</td>
</tr>
<tr>
<td>Are possible actions clear?</td>
<td>[35]</td>
<td>U</td>
</tr>
<tr>
<td>Is it clear what a user must do to take an action?</td>
<td>[20, 35]</td>
<td>U</td>
</tr>
<tr>
<td>Is the required format of user inputs clearly indicated?</td>
<td>[46]</td>
<td>U</td>
</tr>
<tr>
<td>Is it clear what changes in the system have taken place as a result of a user action?</td>
<td></td>
<td>R</td>
</tr>
<tr>
<td>Is there always an appropriate response to user actions?</td>
<td></td>
<td>R</td>
</tr>
<tr>
<td>Does the system inform the user of the success or failure of their actions?</td>
<td></td>
<td>R/U</td>
</tr>
<tr>
<td>Does the system inform the user of the reasons for delays?</td>
<td>[38, 37]</td>
<td>U/R</td>
</tr>
<tr>
<td>Do error messages indicate:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>What errors are?</td>
<td></td>
<td>U/M</td>
</tr>
<tr>
<td>Where errors are?</td>
<td></td>
<td>U/M</td>
</tr>
<tr>
<td>Why they have occurred?</td>
<td></td>
<td>U/M</td>
</tr>
<tr>
<td>What the user must do to recover?</td>
<td></td>
<td>U</td>
</tr>
<tr>
<td>Does the system indicate the current stage in the checkout process?</td>
<td>[60]</td>
<td>U</td>
</tr>
<tr>
<td>Was information always readily available?</td>
<td>[20, 46]</td>
<td>U</td>
</tr>
<tr>
<td>Can the user easily back out of the process?</td>
<td>[38, 37]</td>
<td>U</td>
</tr>
<tr>
<td>Does the system ensure that the final purchase is confirmed by the user?</td>
<td></td>
<td>R</td>
</tr>
<tr>
<td>Does the system allow users to check on inputs provided during the checkout process?</td>
<td>[50]</td>
<td>U/M</td>
</tr>
<tr>
<td>Is jargon and terminology user-centric?</td>
<td>[38, 37]</td>
<td>U</td>
</tr>
<tr>
<td>Is there a help facility?</td>
<td>[60, 46]</td>
<td>U/M</td>
</tr>
</tbody>
</table>

Table 2
EVALUATION METRICS FOR THE CHECKOUT PHASE
<table>
<thead>
<tr>
<th>Evaluation of the System Feedback Quality — LSD Stage</th>
<th>Amazon max 9</th>
<th>Kalahari max 9</th>
<th>BOL max 9</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1</strong> Is it clear what a user must do to search for a product?</td>
<td>6</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td><strong>2</strong> If the search engine fails to find a match, does it offer alternatives so as to prolong the interaction process?</td>
<td>9</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td><strong>3</strong> Does the system inform the user of the reasons for delays?</td>
<td>5</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td><strong>4</strong> Are different types of information clearly separated?</td>
<td>7</td>
<td>5</td>
<td>9</td>
</tr>
<tr>
<td><strong>5</strong> Was information always readily available?</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td><strong>6</strong> Is it clear what the user must do to select a product?</td>
<td>9</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td><strong>7</strong> Can the user easily undo a product selection?</td>
<td>9</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td><strong>8</strong> Is it clear what the user must do to make the transition from LSD to Checkout?</td>
<td>6</td>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td><strong>9</strong> Does the system allow users to check on previous searches?</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>10</strong> Is jargon and terminology user-centric?</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td><strong>11</strong> Is there a help facility?</td>
<td>9</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td><strong>Percentage</strong></td>
<td><strong>73%</strong></td>
<td><strong>57%</strong></td>
<td><strong>64%</strong></td>
</tr>
</tbody>
</table>

Table 3  
EVALUATION METRICS FOR THE LSD STAGE

Discussion

One notices from Tables 3 and 4 that some metrics scored markedly well or badly. A low score should wave a red flag at the developer and can be used to indicate a problem area. A high score shows that the developer has done a good job in providing adequate feedback for that particular feature of the site. A good score does not mean that the developer can become complacent, however. The developer should continuously monitor the performance of the site to ensure that the good features are well maintained. It should also be borne in mind that it is far harder to proportionally improve good aspects of a site than bad aspects. This section will discuss the bad, good and interesting feedback features of the three sites.

1) Low Scores

Two criteria stand out particularly: the lack of a historical facility and inadequate reasons for long or unexpected delays. None of the evaluated sites allow users to remind themselves of previous search criteria. Users in the LSD stage navigate in a situated-action manner, and therefore respond to what is on the screen. A user searching for a specific type of book may type in many different search parameters and may easily forget which parameters have been tried before, especially after a period of time has elapsed. It would be helpful to have a drop-down menu which can be activated by the user in order to see previous search criteria.

In the same vein, there is also a need for the user to be reminded, as they progress through the checkout stage, of their previous inputs. Some sites do provide this but it is often not done consistently.

The other controversial score is the one allocated to the question: Does the system inform the user of the reasons for delays? All browsers give observable feedback on page fetch delays and anticipated completion times. However, many sites, including the ones evaluated, seem to rely completely on this facility rather than providing the user with some sort of site-specific indicator of site access (hit-rate). A user who is given access to such an indicator will perhaps be more patient when sites take a long time to respond. A user should also not be kept waiting when a search is taking some time. It would be better to display a message if the search time exceeds some boundary so that the user knows that the system is still working on the search. The authors did find one site that displayed a message to the effect that the site was busy loading (www3.gartner.com), so that although the site took a little longer than expected to load the user does not become impatient.

Other questions also scored lower than is acceptable. We referred to error messages previously, and the variability of the treatment of errors in these sites confirms our previously-held conviction about the low priority given to this crucial feature in EC sites.
<table>
<thead>
<tr>
<th></th>
<th>Evaluation of the System Feedback Quality — Checkout Phase</th>
<th>Amazon max 18</th>
<th>Kalahari max 9</th>
<th>BOL max 15</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Are instructions and messages displayed by the system concise and positive?</td>
<td>12</td>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>Are instructions clear and prompts unambiguous?</td>
<td>12</td>
<td>6</td>
<td>12</td>
</tr>
<tr>
<td>3</td>
<td>Are possible actions clear?</td>
<td>11</td>
<td>5</td>
<td>11</td>
</tr>
<tr>
<td>4</td>
<td>Is it clear what a user must do to take an action?</td>
<td>13</td>
<td>5</td>
<td>13</td>
</tr>
<tr>
<td>5</td>
<td>Is the required format of user inputs clearly indicated?</td>
<td>15</td>
<td>7</td>
<td>10</td>
</tr>
<tr>
<td>6</td>
<td>Is it clear what changes in the system have taken place as a result of a user action?</td>
<td>13</td>
<td>6</td>
<td>12</td>
</tr>
<tr>
<td>7</td>
<td>Is there always an appropriate response to user actions?</td>
<td>12</td>
<td>6</td>
<td>13</td>
</tr>
<tr>
<td>8</td>
<td>Does the system inform the user of the success or failure of their actions?</td>
<td>14</td>
<td>6</td>
<td>11</td>
</tr>
<tr>
<td>9</td>
<td>Does the system inform the user of the reasons for delays?</td>
<td>11</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>10</td>
<td>Do error messages indicate:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.1</td>
<td>What errors are?</td>
<td>12</td>
<td>6</td>
<td>13</td>
</tr>
<tr>
<td>10.2</td>
<td>Where errors are?</td>
<td>6</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>10.3</td>
<td>Why they have occurred?</td>
<td>6</td>
<td>1</td>
<td>13</td>
</tr>
<tr>
<td>10.4</td>
<td>What the user must do to recover?</td>
<td>10</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>11</td>
<td>Is it clear what the user has to do to complete the task?</td>
<td>11</td>
<td>5</td>
<td>14</td>
</tr>
<tr>
<td>12</td>
<td>Does the system indicate the current stage in the checkout process?</td>
<td>17</td>
<td>3</td>
<td>15</td>
</tr>
<tr>
<td>13</td>
<td>Was information always readily available?</td>
<td>9</td>
<td>6</td>
<td>12</td>
</tr>
<tr>
<td>14</td>
<td>Can the user easily back out of the process?</td>
<td>10</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>15</td>
<td>Does the system ensure that the final purchase is confirmed by the user?</td>
<td>18</td>
<td>9</td>
<td>15</td>
</tr>
<tr>
<td>16</td>
<td>Does the system allow users to check on inputs provided during the checkout process?</td>
<td>9</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>17</td>
<td>Is jargon and terminology user-centric?</td>
<td>12</td>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td>18</td>
<td>Is there a help facility?</td>
<td>16</td>
<td>9</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>Percentage:</td>
<td>66%</td>
<td>59%</td>
<td>72%</td>
</tr>
</tbody>
</table>

Table 4
EVALUATION METRICS FOR THE CHECKOUT PHASE
A feature which we had considered to be essential and basic to good practice, namely that of indicating the substage throughout the checkout stage, was almost absent in the Kalahari site. The user becomes disoriented because the checkout stage encompasses various substages and they have no way of knowing where they are in the process. In defence of Kalahari, it must be said that their checkout phase has only three substages and the developers might have decided that an indicator was superfluous. We do not agree. When one uses Amazon and BOL one is struck by how helpful this indicator is.

The ability to back out of anything is a well-known user-interaction principle [10]. One would expect web pages making up the checkout stage to provide the user with a graceful way out. Users may need to browse further, or need to continue with some other work, or may have forgotten their credit card and thus would want to continue the checkout process the following day. The whole idea of EC is to make things easier and more convenient for the user and not to force them into ‘your’ way of finishing the process. Developers should not expect the universal ‘Back’ button on browsers to do this for them. Users should not have to use the back button to navigate through many different windows in order to continue shopping.

Another bad habit which all sites have fallen prey to is the use of jargon. Users do not need to know about cookies and servers.

Some sites seem to only consider one way of operating and don’t consider that users may need to navigate in a different manner. BOL, for example, does not allow the user to correct delivery addresses during the checkout phase. The user has to back out using the ‘Back Button’ before an icon is available which allows the user to correct addresses.

2) High Scores

Some positive features should also be mentioned. The e-mail confirmation sent out by all three sites evaluated is a very good feature. The scores allocated to help facilities were high for all the sites — and this is particularly good with respect to site usability. What is good for an EC user is automatically good for the site too.

Another universal feedback feature is the requirement that the user positively confirm their transaction. This eliminates possible errors later when users find that they have made a mistake and not have picked it up in time. All sites offer a final page which displays all choices made for the transaction, allows the user to check these choices, and waits for the user to confirm before processing the order. All sites also send the user an e-mail confirming the order so that the user can still exercise a form of recovery, via e-mail, if he or she wishes to cancel the order.

Amazon always attempts to offer alternatives when a search does not produce any results. Another feature provided by both Amazon and BOL is their usage of a customer-driven rating system. Although this is not feedback in the traditional EC sense it is undeniably valuable to potential customers.

3) Interesting and Noteworthy Features

The use of cookies can make the EC process much smoother and make the user feel more at home. Amazon and BOL use their cookies to remember user identities and previous session details. Kalahari remembers only user identities. All three sites keep a history of previous completed transactions and allow users to check on the status of orders.

Kalahari has an interesting feature with both good and bad aspects. They display a summary of the user’s basket on the search page (Figure 3). In terms of feedback this is a very good feature but it must slow down the page fetch time. This is unfortunately one of the problems that must be addressed in providing feedback — one must always play off a good feature against the cost thereof.

Figure 3. Basket Summary

BOL did not score as well as Amazon but they do have a very interesting feature which activates during the checkout stage. Users can link directly to a customer service desk for online help. This is a smart move by BOL because this is the stage during which customers have to be treated well if the site is not to lose the sale. BOL has obviously realised the folly of losing users at this stage and attempt to offer
help by providing personal service. Unfortunately BOL has not exploited their innovation very well because they restrict their online help to daytime hours, and assume that users will be in the same time zone by not explicitly linking their times to GMT. It is also unfortunate that they lapse into jargon when the online chat feature is activated. The activated window instructs the user to java-enable his or her browser, but doesn’t give any instructions on how this may be achieved. It is also a pity that when one of the authors connected to ask a question at midday no one replied. The whole idea of EC is to free users from restrictions of times and place and if BOL is to have any chance of challenging other large EC sites they will have to improve on this very innovative facility.

FUTURE WORK

We would like to refine this evaluation method and also use a similar method to rate other usability aspects of EC interfaces such as robustness, enjoyability and appropriate functionality. The importance of positive engagement for the organisation can not be overemphasised and this aspect of EC sites is also of particular interest to us.

CONCLUSION

Feedback can be considered to be ‘making visible that which should be visible, and hiding what is irrelevant’ [43]. This is not merely a matter of common sense, as is abundantly obvious to any user of EC applications, but rather an issue which should be given due consideration. Feedback, in general terms, can be used to assist the user in understanding the functionality and requirements of an EC application. Feedback can be effectively harnessed to ensure that users do not simply give up on sites but rather stay and spend their money — ensuring the continued presence of the EC site and perhaps the continued existence of the organisation itself.

Most developers would agree with the advisability of feedback provision but find it difficult to measure the feedback quality of their sites. We have therefore provided an evaluation mechanism which can be used by developers to flag problem feedback areas.

References


