Classification of Usability Problems (CUP) Scheme: Augmentation and Exploitation

Sigurbjörg Gróa Vilbergsdóttir¹, Ebba Thora Hvannberg²
University of Iceland
Hjardarhaga 2-6, 107 Reykjavik, Iceland
¹svg2@hi.is, ²ebba@hi.is

Effie Lai-Chong Law
Eidgenössische Technische Hochschule Zürich
Gloriastrasse 35, CH-8902, Zürich, Switzerland
law@tik.ee.ethz.ch

ABSTRACT
Existing defect classification schemes are mainly used to characterize software defects. A few of them are specifically applicable to usability problems, but they have not been validated and their reliability has been assessed in a limited way. The aim of this study is to evaluate comprehensively the Classification Usability Problems (CUP) scheme. First, the reliability was evaluated with raters of different levels of expertise and experience in using CUP. Second, the acceptability was assessed with a questionnaire. Third, the validity was assessed with developers in a field study. Results show that some form of training is required for inexperienced evaluators to exploit CUP fully, but a simplified version of CUP may still be useful for developers and usability practitioners. The evaluation framework employed proved effective for revising CUP and may be applied to validate other related schemes.

Author Keywords
Usability, defect classification, think-aloud, user test, validity, reliability, acceptance.

ACM Classification Keywords
H.5.2 [Information Interfaces and Presentation]: User Interfaces -- Evaluation/methodology

INTRODUCTION
All too often, a method or a tool is suggested for developers of software, without prior understanding of their ways of working and lacking the validity of the method based on empirical knowledge. The same can be said of the work of designers and evaluators of user interfaces. In particular, better definition and understanding of usability problems is needed. Once the problem is understood, an enhanced knowledge about the abstraction level of the problem might help to motivate a solution. Finally, comprehension of the problem’s root cause in developer’s work is searched so that the reoccurrence of such a defect type may be prevented in the future. Several defect classification schemes have been suggested, but a few of them are tailored specifically for human computer interaction development, including Classification of Usability Problems, or CUP.

There are three major rationales underpinning the current work. (1) To expand the scale and scope of studying reliability of the classification scheme with usability practitioners of different levels of expertise in applying CUP and domain knowledge. (2) To assess the acceptability of CUP among its users. (3) To establish the validity of CUP by assessing its usefulness during the redesign of the system of interest.

RELATED WORK
The Classification of Usability Problems (CUP) scheme was developed by Hvannberg and Law [13]. The basic justification behind the creation of CUP is that defects (i.e. a list of usability problems) identified in usability tests will much enhance product quality and process improvement if precise and concrete information about these defects can be presented to the development team. The development of CUP was inspired by Freimut’s [9] proposed structure of a Defect Classification System (DCS). Other existing DCSs were also used as reference frameworks such as: Card [3], Chillarege et al. [4], Grady [10], Lezak et al. [16] and Mays et al. [17]. Hvannberg and Law [13] adjusted the granularity of existing attributes and their subsuming values and created new values to meet the specific needs of user interface developers. A number of attributes from other DCSs are not relevant because in principle when usability problems (UPs) are classified, the raters as usability engineers do not necessarily have access to the source code of the interface. To better understand the basis of CUP and its distinction from other DCS, three DCSs were selected for comparison to CUP. They are: Orthogonal Defect Classification (ODC) [4, 9], Root Cause Defect Analysis (RCA) [16] and User Action Framework (UAF) [2]. These three schemes were selected because we believe they have most relevance to CUP. An important factor that all four
schemes have in common is that none of them have assessed the validity of the scheme, but the current study of CUP attempts to do so as described in section Validity analysis. Only CUP and UAF have systematically evaluated reliability. The current study aims to evaluate CUP’s reliability more extensively as described in the section Reliability Analysis. None of the schemes have evaluated user acceptance. All the schemes except CUP estimated the mean effort for applying the scheme. We view this as an important dimension, which the current study aims to estimate.

CLASSIFICATION OF USABILITY PROBLEMS (CUP)

Goal
The goal of a defect classification scheme has to be clear and serve the needs of an organization to improve the quality of the system under development and the development process. Specifically, the goal of CUP is:

Classify Usability Problems (UPs) further in order to give the developers better feedback to improve their understanding of the UP, help them manage the maintenance of usability, enable them to find effective fixes for UPs as well as preventing such problems from reoccurring in the future.

Application of CUP

The CUP scheme can be divided into two parts, Pre-CUP and Post-CUP. Pre-CUP consists of nine attributes that describe in more detail UPs that are found in usability testing by representative end-users or usability experts. Those results are presented to developers. After some or all of the UPs have been corrected, the developers record four attributes in Post-CUP. The following section provides an overview of the attributes. It also discusses the two changes that were made to the original CUP scheme for this study before the reliability and validity analysis were carried out.

Attributes and Attribute Values

Table 1 shows short definitions of the CUP attributes and lists their values, with the four Post-CUP attributes highlighted in grey. In response to developers’ comments, two attributes were added to the original Pre-CUP scheme [13], Frequency and Context. The developers were interested in knowing how many users/experts experience/predict a particular UP. It provides the developer with an idea of how much different users/experts agree on that particular UP. In the Context variable, a screenshot from the user interface can be used for additional information and clarification. This was added to separate the context description from the UP description itself and to make it easier to group the UPs according to their context. The definitions of the Failure Qualifier values are as follows:

Missing: When the test participant fails to find something in the user interface that she expected to be present.

Incongruent Mental Model: When the user interface is unclear, because it does not match the test participant’s mental model or her previous experience.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Defect ID</td>
<td>Identification number</td>
</tr>
<tr>
<td>Frequency</td>
<td>Number of users/experts that experience/predict a UP</td>
</tr>
<tr>
<td>Trigger</td>
<td>Describes what an user is doing when s/he discovers the UP i.e. task scenario, heuristic, reflective question</td>
</tr>
<tr>
<td>Context</td>
<td>Describes in what part of the user interface the user/expert was when the UP occurred</td>
</tr>
<tr>
<td>Description</td>
<td>Concise description of the UP</td>
</tr>
<tr>
<td>Defect Removal Activity</td>
<td>Usability Evaluation Method e.g. user test and heuristic evaluation</td>
</tr>
<tr>
<td>Impact</td>
<td>Only for UPs found using user tests (UT), calculations based on the tasks scenario that the user was performing.</td>
</tr>
<tr>
<td>Severity</td>
<td>Indicates what effects the UP had on the user/expert: Severe, Moderate, and Minor.</td>
</tr>
<tr>
<td>Task Efficiency (%)</td>
<td>Only for UPs found using user tests (UT), calculations based on the tasks scenario that the user was performing.</td>
</tr>
<tr>
<td>Task Effectiveness</td>
<td>Only for UPs found using user tests (UT), calculations based on the tasks scenario that the user was performing.</td>
</tr>
<tr>
<td>Failure Qualifier</td>
<td>Describes how the user/expert experienced a UP: Missing, Incongruent Mental Model, Irrelevant, Wrong, Better Way and Overlooked.</td>
</tr>
<tr>
<td>Expected Phase</td>
<td>Indicates in which phase of a software development lifecycle the developer is expected to be able to fix the UP: Task analysis and context of use (TAN), Functional requirements (FUR), Quality attribute analysis (QAN), Conceptual modelling (COM), Dialogue design (DIA), Navigational design (NAV), Presentation design (PRE) and Implementation (IMP).</td>
</tr>
<tr>
<td>Actual Phase</td>
<td>The phase where the UP was fixed: Same values as Expected Phase.</td>
</tr>
<tr>
<td>Type of Fault Removed</td>
<td>Describes what in the user interface was changed to fix the UP.</td>
</tr>
<tr>
<td>Cause</td>
<td>Aimed to understand why the developers committed the error: Personal, Communication, Technical, Methodological, Managerial and Review.</td>
</tr>
<tr>
<td>Error Prevention Technique</td>
<td>Ideas on what can be done in the future to prevent the fault</td>
</tr>
</tbody>
</table>
**RESEARCH METHODOLOGY**

**User test**

The purpose of the user test, reported in this section, is to produce data that will be used in the two studies on validity and reliability.

**Application**

This study uncovers defects in a user interface, called the Owl Teaching Web Version 2.0, with a think-aloud user test. It is a university learning management system, which allows teachers to manage their courses online. The application has two user groups, teachers and students. Teachers can provide students with online course materials e.g. lecture notes and web links, send announcements to students, create online discussions and they can also maintain a calendar of events in their course. The student group has fewer functions available to them and mainly uses the system to access the provided learning resources.

The size of the application is roughly 11 thousand lines of code and the first version was released four years ago.

**Method and Procedure**

Eleven participants were recruited for the user tests (UT), five teachers and six students. The student participants were paid about €32 for participating in the tests. None of the participants had used the Owl Teaching Version 2.0 in their courses. The user groups had different tasks that they were required to perform but both groups had nine task scenarios covering the core functionalities of their user group. While performing the tasks, the participants were asked to think aloud. At all the sessions, an assistant UT tester was present and he recorded the test participant’s thinking aloud verbatim. The sessions were also recorded using Camtasia Studio [21] to capture screen activity and the audio of thinking aloud verbatim.

**Measurements**

The performance measures were mainly obtained through observations, including time on task, number of screens of online help consulted, and number of instances of expressed frustration. Primarily, the test participants reported the subjective measures in a think-aloud verbatim protocol, post-task and post-test questionnaires. The users experienced in total 112 usability problems (UPs). The UPs were filtered by consolidating those that were the same, i.e. the users experienced same types of problems in the same parts of the application. This is done by going over the list of UPs, one at a time, and checking whether a similar UP has come up previously on the list. After the filtering, there were 71 unique UPs, 46 in the teachers group and 25 in the student group. Of the 71 unique UPs there were: 26 Minor, 30 Moderate and 15 severe. The average number of UPs experienced per user was 10.2 (SD=4.0, N=11).

**Reliability and Acceptance Studies**

The reliability of the CUP scheme depends on its repeatability. It is only repeatable if there exists consistent shared understanding of the scheme. The classification of defects is a subjective exercise, and therefore it is possible that different raters classify the same defect differently. The credibility of defect data is unconvincing if such disagreement in classifications is dominant [13]. The reliability analysis only addresses the Pre-CUP attributes because they constitute the feedback that developers receive from the usability testing. Concerning the reliability and acceptance of the Pre-CUP scheme, we aim to answer the following three questions to meet the first, ((1) Reliability, RQ1, RQ2), and the second goal listed in the introduction ((2) Acceptance, AQ1)):

**RQ1:** How reliable is the Pre-CUP scheme among a group of novice raters?

**RQ2:** To what extent is the level of interrater agreement influenced by the raters’ expertise in applying Pre-CUP scheme and by their involvement in usability testing the application?
To answer these questions an experiment was conducted which is described in detail in the following section and illustrated in Figure 1.

**Selection of Usability Problems for Experiment**

There were in total 71 unique UPs identified in the user test of the Owl application. We developed a scheme to select a subset of UPs to use in the reliability experiment. When developing the scheme we kept in mind that the classification should not take more than about 90 minutes for the participants because otherwise they would get tired. We anticipated that the participants should be able to rate about 20 UPs in 90 minutes. We reviewed two selection schemes. A case study of Root Cause Analysis [16] used a rather complicated scheme where many items had to be selected, and in the User Action Framework [2] the selection of UPs was based on their real-world expected frequency. In considering the tradeoffs of these selection schemes, we have developed our own to select a subset of UPs to be examined in the reliability experiment:

i. Sort all the UPs according to the UT tasks where they occur.
ii. For each of these UT tasks first sort UPs according to their frequency, which is an objective parameter and then according to their severity. Two raters should perform this severity rating, to find out inter-rater reliability. One of the raters should have conducted the UT.
iii. Select the top half or 50% of the sorted list of UPs per task.
iv. Apply the Pre-CUP scheme to these extracted UPs. A rater who has conducted the UT performs the rating.

At step i, the UPs were classified according to 18 tasks. At step iii, 40 UPs were extracted. The rationale of step v was to maximize the number of types of UPs with different attributes and attribute values, thereby allowing the reliability test participants to work on various types of UPs. After step v, we ended up with about 30% of the total UPs or 21 UPs.

At step ii, two raters were involved that will be referred to as CUP expert and UT expert. The CUP expert has better knowledge about CUP than does the User Test (UT) expert but she conducted the UT. The UT expert carried out the selection process and prepared the data on the 21 UPs i.e. filled out the Defect ID, Context, Description, Defect Removal Activity and Trigger attributes. The agreement between the experts on the severity rating was only fair (kappa value $\kappa = 0.303$). The CUP expert received the same materials as the Pre-CUP experiment participants who all deal with so-called ‘second hand’ data about the UPs because they did not conduct the UT themselves.

**Participants**

Eight participants were recruited to participate in the reliability experiment and they were each paid about €64 for their participation. There were four master degree students, two bachelor degree students and two developers working in industry. Three of the master’s degree students were studying computer science and one was studying bioinformatics. The bachelor’s degree students were from computer science and industrial engineering. One of the software developers held a bachelor’s degree in computer science and the other a master’s degree in language technology.

The participants aged from 20-50 years and the majority of them were 20-24 years old. Half of the participants had work experience in either software development, in user interface design or in user interface testing. Four of the participants had used earlier versions of the Owl system in their studies at the university for 2-3 years, two for 1-2 years and two of the participants had not used the application before the experiment.

**Experimental Treatment**

The experiment (see Figure 1) involved two sessions, which the participants had to attend. A presentation about Pre-CUP was followed by a session where participants had to classify 21 UPs according to Pre-CUP (3 attributes).

The presentation about Pre-CUP covered the following items: explanation of what user tests (UT) are, the goal of Pre-CUP, information about the system where the UPs were identified, explanations about the Pre-CUP attributes and their values, example of a UP that had been classified.
according to Pre-CUP and instructions on what participants were supposed to do in the experiment.

All participants received the following materials during experimental treatment: a printout of the 27 slides that were used in the training presentation, a 5 pages booklet about Pre-CUP with an example of its application, the tasks that were used in the UT on the system and access to the part of the Owl system which was used in the UT.

**Experimental Task**

The information session about Pre-CUP was held by the CUP expert and the UT expert, and the participants joined the second session two or three days later. They were instructed to familiarize themselves with the materials about Pre-CUP so that they would be better at applying it to the UPs. They were also asked to try out the Owl system between the two sessions in order to know the domain better and have an enhanced understanding of the UPs. The participants were instructed to record the time they spent on these activities.

The second session of the experiment was conducted in a computer lab so that the participants could examine the system as they were classifying the UPs. The participants received, on paper, 21 UPs where the first five Pre-CUP attributes had been filled out for each UP. The attributes that had been filled out beforehand were: Defect ID, Context, Description, Defect Removal Activity and Trigger. The participants also received a printout and an electronic version of the seven screenshots that the Context attribute referred to. The participants rated three attributes for each UP i.e. Severity, Failure Qualifier and Expected Phase. The participants then entered their result into a provided Excel template where they also recorded the time they spent on preparation for the session and the time they had spent on the CUP classification. All participants filled out a questionnaire about their age, education, previous experience of the system and whether they had software development experience.

At the end of the second session the participants filled out a post-experiment questionnaire based on the Technology Acceptance Model (TAM) [6] which had been adapted to evaluate the CUP scheme to empirically examine the raters’ perceptions of the CUP scheme.

**Validity Study**

The validity study of the CUP scheme was carried out in a real software development environment, with the aim of answering the following three questions intended to answer the third goal in the introduction (3):

VQ1. Do developers think that the CUP scheme aids their understanding of a UP?

VQ2. Do developers find the CUP scheme useful when prioritizing what UPs to fix and when deciding how to correct UPs?

VQ3. How well do the developers understand the CUP attributes and their values? Do developers propose any changes to the CUP scheme?

The remainder of this section describes the environment in which the study was performed, the data that was collected and how it was analyzed. The results are presented in section Validity Analysis.

**Environment of Study**

Two software developers of Owl were involved in the study. One is the manager of the software development department and the other is the main programmer of the Owl system. They work together on the design of the application and decisions regarding it; one of them does the programming. They both perform testing on the Owl system, which is mostly unit testing and have never used the results of a UT. The programmer of the application has a B.Sc. in computer science and had worked for the company for 15 months when the study began. The department manager has worked for the company for nine years and he is self-educated.

**Data Collected**

The data was mainly collected in the form of semi-structured interviews and meetings with the developers of the system over a three month period. All the meetings and interviews were audio taped, transcribed and coded using a qualitative analysis tool called QSR N6 [19]. From the time when the developers received the CUP results from the UT, ten meetings were held with them, a total of 12 hours. Three meetings were spent on discussing particular UPs and their CUP ratings. These meetings provided the developers with a feel for the scheme and opened discussion for what they thought about it. We also examined the documentation available for the Owl system, which was limited because their software development process is mainly agile, and did not involve a great deal of documentation. The most relevant documentation was a list of 177 non-unique items that included comments and error reports from users of the system. The items on the list have been collected through the company’s help line and by running web surveys about the company’s services and applications. To motivate solutions to items/problems, the developer went through this list of items, which includes: suggestions for enhancements, errors, and comments about functionality, usability and performance. Then he categorized all the items according to a potential solution or change in design. Both developers estimated next how long it would take to implement the changes and prioritize them.

**RESULTS AND DISCUSSIONS**

**Reliability analysis**

**RQ1: Reliability of Novice Raters**

To evaluate reliability, Fleiss’ [8] generalized kappa was used as referenced in [15]. The generalized kappa statistic is an extension of the kappa statistic to include multiple
raters and it is useful to measure interrater agreement among three or more raters [15]. We used King’s Microsoft Excel spreadsheet to calculate the generalized kappa values [14].

The generalized kappa results are reported in Table 2. The reliability for the Severity attribute is the highest and it is the lowest for the Expected Phase attribute. The reliability results for the Expected Phase attribute suggest that its definition and values require refinement. This was confirmed by the post experiment questionnaire where the participants commented that the values for this attribute needed to be defined more clearly. There are probably too many values defined for this attribute. Another explanation may be that raters do not have adequate background in user interface development to be able to distinguish the difference between the phases, e.g. between navigational and dialogue design.

The Severity attribute has the highest reliability and part of the reason may be that it has the fewest values of the three attributes being examined. The fact that Severity is an ordinal scale (i.e. minor, moderate and severe) may have increased the understandability of the raters more than the other attributes, which are nominal scales.

**RQ2: Interrater Agreement and Influencing Factors**

Since the participants were all novices in CUP, we wanted to compare their ratings pairwise to the ratings of a CUP expert. The ratings of every participant were compared to the ratings done by the CUP expert. The CUP expert was as familiar with the UPs as the novice raters and received the same data on the defects. Second, we aimed to see how much knowledge about the UTs themselves helped the rater. Therefore, we compared the ratings of the CUP expert to the UT expert.

In this part of the reliability evaluation we used Kappa coefficient because it is the recommended statistical method for verifying interrater agreement for nominal scale data classified by two raters like this case [20]. Kappa values can be qualified into five levels of agreement ranging from “poor” to “very good” [1].

Table 3 shows the kappa results for the comparison between the raters in the Pre-CUP experiment and the CUP expert. The Severity attribute has the highest level of agreement, moderate (0.41-0.61) for three raters but five raters have poor (<0.2) level of agreement with the CUP expert. The Failure Qualifier attribute appears to have the most consistent level of agreement because it is fair (0.21-0.40) for seven raters. The Expected phase attribute has the lowest overall level of agreement because for half of the raters it is poor and for half of the raters it is fair.

If we consider the time that the participants spent on preparing themselves for the second session of the experiment then five of them spent less than an hour on preparation, two spent 1-2 hours and one participant spent 2-3 hours. The time that participants spent on the Pre-CUP classification itself in the second session ranged from 60-117 minutes and the mean time was 86 minutes (SD=17.01). In comparison, the CUP expert spent 50 minutes on the Pre-CUP classification, which is considerably less than the mean time and can probably be explained by the fact that she knows the CUP scheme well.

All, but one of the raters, have a poor level of agreement with the CUP expert for one of the three attributes. One rater has poor agreement for two attributes and that is rater 5. Rater 5 is also the only rater who has a negative kappa value, and that is for the Expected Phase attribute, which implies that these raters agreed less than would be expected just by chance. Rater 5 spent less than one hour on preparation and spent the least time of all the participants on the Pre-CUP classification i.e. 60 minutes. It is possible that Rater 5 rushed himself too much.

The raters who spent more than an hour on preparation were raters 1, 2 and 4. As shown in Table 3 raters 2 and 4 have the highest level of agreement for the Severity attribute and rater 1 has the highest level of agreement for Failure Qualifier. However, these three raters all have a poor agreement for the Expected Phase attribute.

The kappa results for the CUP expert and the UT expert are shown in Table 4. The CUP expert and the UT expert had a fair level of agreement for the Severity and Expected Phase attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>No. of possible</th>
<th>Kappa</th>
<th>Agreed cases*</th>
<th>Level of agreement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Severity</td>
<td>3</td>
<td>0.33</td>
<td>12</td>
<td>Fair</td>
</tr>
<tr>
<td>Failure qualifier</td>
<td>6</td>
<td>0.50</td>
<td>13</td>
<td>Moderate</td>
</tr>
<tr>
<td>Expected phase</td>
<td>8</td>
<td>0.22</td>
<td>7</td>
<td>Fair</td>
</tr>
</tbody>
</table>

*Note the total number of cases is 21
attributes and moderate for the Failure Qualifier. In case of Expected Phase the experts only agreed in 7 cases out of 21 and that is the lowest level of agreement.

Discussion on Interrater Agreement
The level of agreement between the 8 novice raters from the experiment and the CUP expert was overall best for the Failure Qualifier attribute. This concurs with the agreement between the two experts (see Table 4), but as was discussed above, the Severity attribute has the highest reliability among novices according to the generalized kappa results. The Severity attribute has only three possible ordinal values and is therefore easier for novices to learn and remember rather than the eight nominal possible values for Expected Phase. The results confirm that the Expected Phase attribute requires refinement. The level of agreement between the raters and the CUP expert on the Failure Qualifier attribute was quite consistent i.e. nearly always fair and that might imply that the raters understood part of the six values quite well and rated similar to the expert but perhaps lacked understanding and practice in the others. The value “Incongruent Mental Model” was the most commonly used by the raters and seven of them agreed the most times with the CUP expert on that value. The novice raters never agreed with the CUP expert on the value “Better Way” and half of the novice raters never agreed with the expert on the value “Irrelevant”.

If we examine some of the disagreements between the CUP expert and the UT expert then the most serious disagreement in the Failure Qualifier attribute is when one expert rates a UP as “Missing” and the other rates it as “Overlooked”. These two values have very different definitions because “Missing” is defined as: when the user fails to find something in the user interface that she expected to be present and on the other “Overlooked” is defined as: when the user overlooks an entity in the user interface. This disagreement comes up twice. Another important disagreement comes up once, which is when the CUP expert rates a UP as “Wrong” and the UT expert rates it as “Better Way”. These values are very different because in one case the user notices that something has gone wrong in the interface like a programming bug but the Better way value applies when the user actually suggests that something should be done differently in the UI. These disagreements imply that the descriptions of the UPs were insufficient or that the raters are interpreting the values differently. This issue is further explored in the following section.

In ratings for the Expected phase attribute, the experts never agreed on the following phases: TAN, FUR or QAN (see Table 1). The reason might be that it is hard to discriminate between these phases.

Regarding the comparison between the two experts there are two main factors that have to be considered. It is possible that the UT expert is better qualified to rate the Severity and Failure Qualifier attributes because she conducted the UT and therefore has better knowledge of what exactly happened. On the other hand with regard to the attribute Expected Phase it is possible that the CUP expert has better knowledge and qualifications to rate it because she has practical experience from software development whereas the UT expert has not. These factors might influence the level of agreement achieved by the experts.

When comparing the interrater agreement of this study with previous ones, we notice that the agreement between experts in this study is similar as in [13] for Severity, i.e. fair and moderate for Failure Qualifier in the case of UPs discovered in heuristics evaluation. The agreement is slightly worse than in [13] for Failure Qualifier in the case of UPs from UT where the agreement was good but in this study is only moderate. The agreement in [13] was moderate for Expected Phase but for this study, it was fair. We account this difference to experience in applying CUP.

**Consensus of Experts**

In a study by El Emam [7] two groups independently performed a software process assessment based on the same data. After their independent ratings, the groups would meet to reach a consensus in their ratings and as a result produce a consolidated rating which was the outcome of an assessment. Based on this idea and the fact that the kappa value between the experts was considered only fair for two attributes it was decided that the two experts would have a consensus session. The aim of the session was to go over UPs where they disagreed on some rating in order to find out why they disagreed and learn from those disagreements. After the consensus session, the kappa was calculated for the new ratings.

As shown in Table 5 the experts achieved a very good level of agreement (0.81-1.00: Very good) after their consensus session. The following were the main reasons for the experts’ disagreements:

- CUP expert thought the description for the UP was not precise enough and therefore misunderstood the problem.
- The description of the UPs contained two closely related usability issues and the experts were considering them from different perspectives. In these cases the UP might have been split into two separate ones.
- The attribute values overlap and therefore both experts were correct in their rating.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>No. of possible</th>
<th>Kappa</th>
<th>Agreed cases*</th>
<th>Level of agreement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Severity</td>
<td>3</td>
<td>1.00</td>
<td>21</td>
<td>Very good</td>
</tr>
<tr>
<td>Failure qualifier</td>
<td>6</td>
<td>0.94</td>
<td>20</td>
<td>Very good</td>
</tr>
<tr>
<td>Expected phase</td>
<td>8</td>
<td>0.88</td>
<td>19</td>
<td>Very good</td>
</tr>
</tbody>
</table>

*Note the total number of cases is 21

Table 5 Kappa for the CUP expert and the UT expert after consensus session
• The CUP expert thought a Context attribute value was unclear.
• Both experts acknowledged a mistake.
• Admittedly, the severity rating may be partly based on the UT expert’s opinion influenced by user tests rather than solely on the description.
• The CUP expert lacks domain knowledge of the application.

In the severity rating for the UPs, the UT expert made more changes after discussing the rating with the CUP expert. In the cases where the CUP expert changed her severity rating, it was because the description had not been precise enough. In the cases where the UT expert changed her ratings it was because she had made a mistake or that she changed her mind after discussing and reviewing it with the CUP expert.

In the Failure Qualifier ratings the CUP expert made more changes to her ratings than did the UT expert. In one case both experts made a mistake and after discussing it, they felt that no value matched the UP. In that particular case the CUP expert had rated it as “Wrong” and the UT expert rated it as “Better way”. They probably both felt inclined to rate the UP but after reconsidering it they realized that no value matched. The experts thought that the value “Incongruent Mental Model” overlapped with the values “Missing” and “Overlooked”. The reason for this overlapping is that the user may expect some entity to be present in the UI or misses some entity that is present because his mental model of the UI does not match with it. Suggestions for improvement on CUP will be given in a later section.

In the Expected Phase ratings the experts changed about the same amount of ratings. The experts reached an agreement for most of the UPs after discussing the UP and reviewing the attribute values. The experts both felt that there was some overlapping in the attribute values especially TAN, FUR and QAN (see Table 1). These three phases are strongly related and it is hard to distinguish between them.

Acceptance of CUP
This section will present and discuss the results from the post experiment questionnaire filled out by the raters in the Pre-CUP experiment.

The post experiment questionnaire had 19 questions, thereof 16 closed questions and 3 open-ended questions. The closed questions were in a randomized order and they were all measured on a 5-point Likert scale using the opposing statements format. Three perception-based constructs were used to evaluate the scheme and they are the following:
• Perceived Ease of Use (EASE): the degree to which a person believes that using the scheme would be free of effort. (Five questions)
• Perceived Usefulness (USEFUL): the degree to which a person believes that scheme would be useful. (Eight questions)

Table 6 Mean of raters’ responses

<table>
<thead>
<tr>
<th>Construct</th>
<th>Mean</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perceived Ease of Use</td>
<td>3.48</td>
<td>0.60</td>
</tr>
<tr>
<td>Perceived Usefulness</td>
<td>4.00</td>
<td>0.45</td>
</tr>
<tr>
<td>Intention to Use</td>
<td>3.04</td>
<td>0.81</td>
</tr>
</tbody>
</table>

• Intention to Use (INTENT): the degree to which a person intends to use the scheme. (Three questions) [18]

Table 6 shows the mean and standard deviation for the three constructs. Perceived usefulness has the highest mean and intention to use the lowest and this was confirmed by the responses in the open ended questions.

To answer the research question AQ1, a multiple regression analysis was performed. The relatively high multiple correlation coefficient (adjusted \( R^2 = 0.74 \)) suggested that EASE and USEFUL taken together could contribute to the prediction of INTENT. As indicated by the beta weight (\( b \)), EASE was a significant predictor (\( b = 0.66, t = 2.48, p = .05 \)) whereas USEFUL was not. The part Pearson correlation coefficients (\( r \)) are 0.48 and 0.22 for EASE and USEFUL, respectively. These statistics correspond to measures of effect size in terms of partial eta squared (\( \eta_p^2 \)), which are 0.55 and 0.21 for EASE and USEFUL, respectively.

The findings imply that ease of use is a critical factor in determining whether the novice evaluators will adopt the CUP scheme in their future work. This observation is not surprising because these evaluators are not involved in the entire process of applying the scheme (i.e. they have not worked on the Post-CUP attributes) and thus cannot see its possible usefulness. What concerns them most is whether they are able to understand and apply the scheme rather than whether it is useful to improve the system evaluated.

When asked in one of the open-ended questions how participants thought the scheme could be made easier to use, they answered that the definitions of the Expected Phase values should be explained more thoroughly and more examples provided. When asked what their major reasons for deciding to use or not to use the CUP scheme in the future they said that they thought the scheme took too long to use and that they would need more confidence in their own expertise to make a decision on intention to use the scheme.

Validity analysis

Results for VQ1
The results for VQ1 were positive because the developers thought that all the Pre-CUP attributes except for Severity and Task Efficiency improved their understanding of a UP. The developers focused mostly on the Description attribute and found them well written and to the point. The Context attribute supplements description where it is unclear. The screenshots that were provided with the textual descriptions were especially useful. The developer looked less at the Trigger. The Failure Qualifier was deemed useful to understand the description of the problem and the developer
said the following about it: “It [Failure Qualifier] can prevent misunderstanding. It helps with analyzing the UPs and to see the real problem.”

Results for VQ2
Interviews with developers revealed that they were not only concerned with the frequency and severity of problems as input to the prioritization task (VQ2), but also weighed the importance of the tasks and hence the Trigger attribute was useful as input to prioritization. Neither Task Efficiency nor Failure Qualifier is useful for project planning. When planning for release increments, the developers commented on the Context attribute that: “If there are many programmers working on the system the Context can be used to allocate UPs among them.”

The developers thought that only two attributes could help them to correct a UP (VQ2) and they are: Failure Qualifier and Expected Phase. The programmer made the following comment on Failure Qualifier: “[This attribute] helps you think about what has to be fixed; if it’s overlooked maybe something has to be made more obvious in the UI.”, and on Expected Phase: “It makes you think about how you will fix the UP because you are thinking of where in the development [lifecycle] you’re going to fix it. [It] gets you on the right track.”

Results for VQ3
The developers did not suggest any major changes to CUP (VQ3). They understood all the attributes quite easily except for Expected phase and Task Efficiency. The developers’ view that the attribute Expected phase is hard to understand concurs with the outcome of the interrater agreement in the reliability study. The developers felt that it had too many values, some of the definitions were too technical and they found it hard to distinguish between the values. They also found the Task Efficiency hard to grasp since it is difficult to benchmark it to another value and it does not have a specified range. On Task Effectiveness the developers found the completion rate useful but not the mean time-on-task. Again, it was because there was no baseline against which it can be compared.

The developers are not native English speakers and that may have affected their understanding of some of the more technical words in the CUP attribute definitions. It is interesting that the developers rely most on the Description attribute. However, the classifying parameters may help more as the Description is inferior or the usability problem is more complex.

Implications for the Improvements of CUP
This section will describe the main changes that we propose on CUP based on the results from the reliability and validity analysis. To improve the attribute Expected Phase we have taken the approach to simplify the structure of the values by organizing them hierarchically into three development stages: Requirement analysis, Design and Implementation. Three values belonging to Requirement analysis are: Task analysis and context of use, Functional requirements, and Quality attribute analysis. Design has four values: Conceptual modeling, Dialogue design, Navigational design and Presentation design. By structuring the values hierarchically, we hope that it is more accessible to rate the UPs; first, the rater selects one of the three stages and can then focus on the sub-phases. We did not want to merge values into fewer values because then the classification would be less accurate. The descriptive definitions of the Expected phase values are simplified in order to make them more understandable.

A defect classification scheme needs to take into account that many development teams are not using user interface lifecycles at all [11]. Thus, instead of presenting the engineer with a predefined lifecycle for user interface development for the Expected phase, an assessment of individual team’s current practices can derive a lifecycle that can then be used in later stages to reveal the root phase of a usability problem. The advantage of this approach is that the engineer will be familiar with the values of the Expected phase variable, but the disadvantage is that if the lifecycle is too simple, it may not give enough information about the root cause.

CONCLUSION
We have evaluated the reliability, validity and acceptability of the CUP scheme by conducting comparative experiments, qualitative studies and questionnaires. We believe that the contribution of the paper is twofold; one for developers and evaluators of human computer interaction and a second for researchers of work practices and tools. First, the results of the study serve as valuable insights into the improvement of a maturing usability tool, which can strengthen the feedback between evaluation and redesign. Second, the study provides a methodological framework for assessing reliability, validity and acceptance of a tool under development.

Analyzing reliability is useful to improve the scheme. Results on the reliability of CUP tend to indicate that the expertise and experience of raters is a critical factor or a determinant for applying this tool consistently and effectively. An expert CUP user could analyze the given UPs differently from a novice CUP user. The former is able to produce more accurate analysis than the latter. It implies that systematic training on deploying CUP is imperative. The inference for future work is to identify effective training strategies to enhance novice raters’ understanding. A comparison of two equal sized groups, i.e. of novices and experts would hopefully lead to a more valid study. In addition, we can further break down raters’ expertise and experiences in terms of three variables, namely the actual involvement in conducting the UEM of interest, the domain knowledge of the system under investigation, and the expertise of the CUP scheme. This can be a motivation for further studies into the systematic investigation of the variables’ effect on the results of the CUP analysis.
The validity analysis helped us understand whether the CUP attributes are relevant for developers’ work. The study showed, interestingly, that developers were more focused on creating intuitively holistic solutions to a set of problems, than on systematic individual problem corrections. This may suggest that grouping of the problems or their abstraction may indeed be useful. The study also showed that project management is of great concern and some of the attributes supported that task well. The unfamiliarity of this study’s developers in using the results of systematic tools such as UTs, applying problem reviews and problem tracking, may have had an affect on the results. The results on the Expected Phase attribute convinced us that tools used by developers need to be tailored to their working framework, knowledge and maturity. It will be a tradeoff between cost and benefit. More fine-grained attributes, such as Failure Qualifier and Expected Phase, can give richer data providing more insight into plausible origins of UPs, but may demand more expertise or initial training effort.

The results of the acceptability study shows that practitioners are concerned with the effort spent in applying any sort of tool. Whereas, our study has collected preliminary data on cost-effectiveness of CUP, which indicate that CUP is not time consuming and did not delay deadlines and releases of the Owl software, further data is needed to assess the direct benefits in terms of effort saved.

A triangulation study, like the one we have done, with experiments, qualitative work in the field, and a questionnaire is helpful in assessing and improving an intervention suggested for user interface developers. Our experience from this study shows that such an intervention, not only helps us evaluate the scheme in question, but also tells us a great deal about current practices and constraints of developers’ work.

REFERENCES