International Journal of Engineering Education Vol. 27, No. 1, pp. 1–7, 2011
Printed in Great Britain

0949-149X/91 $3.00+0.00

Devising Instruction from Empirical Findings on Student Errors: A Case in Usability Engineering Education*

ELENA GARCÍA-BARRIOCANAL, SALVADOR SÁNCHEZ-ALONSO and DANIEL RODRIGUEZ-GARCÍA
University of Alcalá, Polytechnical School, Ctra.Barcelona km. 33,700, Alcalá de Henares, 28871 (Spain).
E-mail: {elena.garciab, salvador.sanchez, daniel.rodriguezg@uah.es}

Problem-based learning relies on the use of problems as the fundamental activity driving the learning process, focusing on the application of knowledge to realistic settings. Problems requiring students the design or evaluation of artifacts are a fundamental ingredient of engineering education in diverse fields. In those settings, the effectiveness of instructional design critically relies on the quality of the problems used, which should emphasize the aspects that students usually find difficult to master, so that relevant domain knowledge is exercised during learning activity. The analysis of the errors in student's solutions to problem assignments can be used as an empirical source of information for the instructional design of problem collections. In this approach, problem design is driven by findings on the kind and frequency of errors. This paper reports the use of such an approach in the domain of heuristic usability evaluation in the context of an introductory Human Computer Interaction course, using the 3C3R model as a framework. The method for data elaboration and the resulting approach to devising problems can be transferred to other domains in which similar high-level design analysis is required.

Keywords: usability engineering; heuristic evaluation; student assignments; problem-based learning; evidence-based education; 3C3R model

1. Introduction

Problem-based instruction [1] has been subject to significant research interest in recent years Particularly, evidence that problem-based learning (PBL) leads to better outcomes for the learning styles of engineering students has been reported in recent studies [2]. However, the superior effectiveness and efficiency of problem-based learning approaches has also been questioned in previous meta-studies, e.g. Colliver [3] criticized the credibility of claims about the ties between PBL and educational outcomes in some particular domain context. Particularly, it has been pointed out that ineffective problems could affect whether students acquire sufficient domain knowledge, activate appropriate prior knowledge, and properly direct their own learning [4–5]. In consequence, the design of problems [6] can be considered a critical precondition for the effectiveness of problem-based learning, as it determines at least to some extent the subsequent outcomes of PBL. In that direction, Hung [7] proposed the 3C3R method, a systematic conceptual framework for guiding the design of effective and reliable problems for PBL. The 3C components (content, context, and connection) address the static properties of a problem in terms of intended content to be learned, the specific and unique contextual factors to be considered, and the conceptual connections of the problems within the curriculum, while the 3R components (researching, reasoning, and reflecting) deal with dynamic properties of the problem by analyzing the problem's cognitive processing requirements for the students.

Problems can be defined as 'questions rose for inquiry, consideration, or solution'. Instructors facing problem design for their courses attempt to craft problems to support competency development, thus trying to anticipate the 'problems in solving the problems', i.e. facing students with exercising most of the potential difficulties they would find in performing realistic work activities. This can be approached from the experience and sound knowledge of the problem designers about the topics to be taught, self-reflecting on potential difficulties and pitfalls. In some disciplines, it is easy to find problem collections in textbooks or on-line learning resources that can be used as a form of reliable source of practical experience in proposing and evaluating problems. However, there are topics or disciplines for which few or fragmented problems collections are available, and in general problem collections do not provide hints on the kind of learning difficulties and detailed topic coverage of each of the problems they contain. In those cases, an empirical approach would be helpful to continuously improve and evolve problem collections.

Empirical data on problems faced by students can be gathered from the assignments in previous

* Accepted 2 September 2010.

courses. The careful examination of assignment solutions given by students is the best source of evidence for devising new problems or updating existing ones, if we assume that student cohorts are reasonably homogeneous from one year to the next. The 9-step process of the 3C3R model includes step 4 '*select/generate PBL problem*', which using an empirical approach can be guided by evidence of the performance of students with past problem collections [7]. Then, affordance analysis and conduct correspondence analysis (steps 5 and 6 respectively) serve as an evaluation that the main problem elements have actually be included in the new problem collection.

This paper describes the use of an empirical approach to devising problem collections that can be integrated in the 3C3R model. The method has been applied to courses in Human Computer Interaction, and concretely to heuristic evaluation, a form of open evaluation of interaction designs that requires high-level thinking skills and application of principles and guidelines.

The rest of this paper is structured as follows. The second section provides background information on HCI and usability evaluation as part of the curriculum of engineering disciplines. Then, the third section reports the result of the empirical study on student's assignments used as the case study for the approach described. The fourth section describes the results and how they can be applied to a problem design model. Finally, conclusions and outlook are provided in the fifth section.

2. Background and approach

Human computer interaction (HCI) is a multi-disciplinary area of study that is essential in the education of engineers in every discipline that deals with the construction of man-machine interfaces [8]. One of the key competencies to be acquired in HCI is usability evaluation. Usability evaluation requires general analysis and critical thinking skills, as usability problems require judging interaction designs on the basis of empirical data or heuristics. Also, knowledge on usability is described in terms of generic rules (often called 'usability guidelines') or principles that require practice to be mastered. There exist various techniques for evaluating usability depending on available resources (time facilities and resources), evaluator experience, ability and the stage of development of the software under review [9].

Human-Computer Interaction is also one of the fourteen knowledge areas specified in the computing curricula recommendations elaborated cooperatively by the Association for Computing Machinery (ACM), the Association for Information Systems (AIS) and the Computer Society (IEEE-CS). As such, it is a required topic for all the computing disciplines, namely, Computer Science, Computer Engineering, Software Engineering, Information Systems and Information Technology. Consequently, graduates in these disciplines are expected to have acquired practical skills on the engineering of interactive systems. This is reflected in the 2008 Computer Science curriculum update in the as a requirement for graduates to be able to 'apply the principles of human-computer interaction to the evaluation and construction of a wide range of materials including user interfaces, web pages, multimedia systems and mobile systems' [10]. The ACM/IEEE/AIS curricula recommendations for Computer Science include 8 core hours of Human-Computer Interaction, which is concerned with the required skill of 'knowing how to create a usable interface and testing the usability of that interface'. In the detailed topics related to HCI, the recommendations include 'evaluation without typical users', including guidelines, heuristics and expert-based analysis. While user testing is considered the most reliable way of evaluating user interfaces, teaching guidelines and heuristic evaluation have the benefit of not requiring students to be provided with an observational setting, so that distance students are able to exercise that kind of evaluation. Further, such kinds of evaluations do not rely on the availability of users for testing, but on the application of theoretical elements and guides. If these elements are used for summative assessment of students, the student's responses can be evaluated rather objectively by instructors.

It seems apparent that problem-based approaches to instructional design may be adequate for teaching usability evaluation. However, such approaches require a carefully devised set of problems that provide the required progressive scaffolding [11]. Skov and Stage [12] reported a study comparing problems found by students using a conceptual tool with students not using it and with the evaluation outcome of the teachers. The use of the tool resulted in more problems found by students, which supports the idea that additional scaffolding elements are required for usability evaluation. The elaboration of case collections for usability engineering has been approached by Carrol and Rosson [13]. These are comprehensive cases that fit project-based education, but there is a need to understand and learn to apply the concrete guidelines and heuristics related to expert-based usability evaluation, as these can provide detailed insights on how students face usability problems.

Heuristic evaluation is a problem-oriented usability evaluation method [14]. In its initial proposal by Nielsen and Molich [15], it was found that it served to identify 55 to 90 percent of the known usability problems user interfaces, concluding that heuristic evaluation was a cheap and intuitive method for evaluating the user interface. Heuristic evaluation has an additional interesting property in the educational context: it forces students to classify usability problems, assess their importance and argument why they qualify as such. Consequently, the analysis of records of student heuristic evaluation has the potential to uncover underlying false assumptions, misunderstandings or in general difficulties in acquiring user interface evaluation abilities. In general, finding problem collections for usability evaluation is difficult, and there are not classified problem sets available, as can be found in other computing topics as programming or databases.

3. Method and data gathering

The context of the present case was an elective course on Human Computer Interaction at the last year of a four-year degree in Computer Science at our University. The authors had been teaching the course since 2004 following a continuous assessment method. The course starts with an HCI fundamentals module (where usability attributes, principles and some guideline collections are introduced), followed by a module on user interface design and then a usability evaluation module. In that module, students are taught about user testing, but also other methods including heuristic evaluation. One of the assignments included in the continuous assessment presents the students with concrete Web sites for heuristic evaluation, following Nielsen's heuristics and rating scales. Students have previously exercised the technique at a heuristic evaluation lab, and they are in principle equipped with knowledge on guideline-based assessment as a supplementary tool, concretely, they have been introduced at the beginning of the course to guideline-based usability analysis using the research-based guidelines elaborated by the U.S. Department of Health and Human Services [16].

The users have to report on problems found as exemplified by the Table entry (Table 1).

Categories for problems are the ones first described by Molich and Nielsen [17] that have reached widespread used in practice: *visibility of system status* (VSS), *match between system and the real world* (MSRW), *user control and freedom* (UCF), *consistency and standards* (CS), *error prevention* (EP), *recognition rather than recall* (RRR), *flexibility and efficiency of use* (FEU), *aesthetic and minimalist design* (AMD), *help users recognize, diagnose, and recover from errors* (HURE) and *help and documentation* (HD). Hvannberg, Lai-Chong and Larusdottir [18] found no significant differences between using Nielsen's heuristics and the cognitive principles of Gerhardt-Powals, and no difference was also found in either using a web tool or paper, so we have initially not considered alternative interaction categories from other authors.

The students are asked (both at the lab and as an evaluation assignment) to develop a heuristic evaluation report, including providing a description and rationale for each problem found as reported in Table 1. Experience has shown that different people find different usability problems [14]. Therefore, we can reasonably expect that the collections of the problems found individually by all the students in a group would cover all the problems found by the instructors. Differences in the problems found by students and instructors are indicators of deficiencies in their understanding of some aspect of usability that deserve attention. From another viewpoint, the analysis described below serves as an assessment of the problem selection or design done by the instructors, as it reveals if all the relevant problem categories were actually included in the selected problems.

The analysis of assignments requires a detailed representation of problems found versus actual problems (as identified by the instructors), an analysis of the appropriateness of the assessment of severity, and the qualitative examination of justifications in the problems found. The data analysis was done by creating a database following the schema depicted in the Fig. 1.

Table `Students` recorded information on the students evaluated, the academic year in which they took the HCI course, the site they evaluated heuristically and their overall grade in the course

Table 1. Example fragment from a student's heuristic evaluation report

	Severity					
Category	**0**	**1**	**2**	**3**	**4**	**Error description**
VSS		X				The Web site is too much interactive
UCF					X	Some pages linked from the homepage have no option to *go back* or this option is difficult to find.

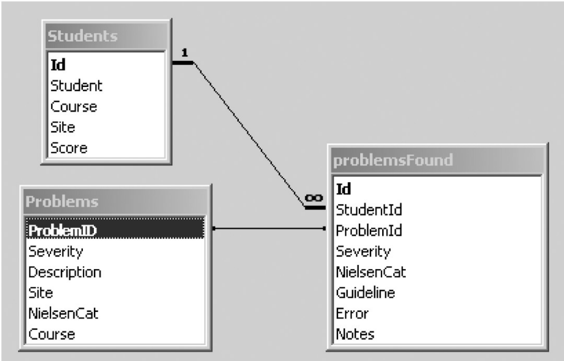

Fig. 1. Database schema used for data collection.

(`score`). Then, each of the problems identified by students were recorded in the `problemsFound` table, including severity assigned, Nielsen category selected, and the related guideline(s) identified (if any). Note these problems found by students are not unique, but each identification of an error (obviously, many of these errors reported correspond to the same 'real' error). The instructors filled the `Error` field, classifying entries in which students failed to justify the reason of the problem, or simply reported a situation that was not actually a usability problem. Problems found that were actual problems are related to the `Problems` table, which stores the 'correct' problems, categories and severity ratings. Matching problem found descriptions was done by the instructions, even though matching problems has been found to be controversial across matching techniques [19], here the group and criteria are homogeneous, and they are built by the same matcher.

The heuristic evaluation assignments for thirty students from two the 2008–2009 academic year and of 55 students from the academic year 2007–2008 were analyzed and data was recorded in the form depicted in Fig. 1. The instructors, student profile, materials and instructional approach were the same in both years. Table 2 reports the overall Figures related to the data gathered. The column *Students* refers to students that submitted their solution to the heuristic evaluation assignment only.

Correct problems found are defined as problems that coincide with the ones found by the instructors and that are described correctly.

4. Results and discussion

In this section, a discussion of the data gathered for the study is reported, with an analysis of the interpretation of data gathered according to the concrete topic of heuristic evaluation. Then, the integration of the method used for the data gathering and analysis in the 3C3R model is reported.

4.1 Evaluating problems and problem design

The first important evaluation aspect is the degree to which students performed the task. The high percentage of coverage of problems identified by instructors represents the degree in which the group of students as a whole identified collectively all the problems actually present (i.e. actually identified by the instructors). This is coherent with models on the finding of usability problems reported elsewhere [20]. The low figures of correct problems found are in principle also consistent with those previous models, as a single evaluator rarely finds more than 60% of the total problems. However, the variance of errors found by students is high, which points out that there is some barrier for some students even to identify some prominent problems. This suggests general misunderstandings of usability problems, as confirmed by the qualitative analysis of justification reports. For example, the student with Identifier 5 reported a total of 31 problems, of which only 11 matched actual problems. Nonetheless, from the other 20 problems, 10 of them were incorrect because they were considered redundant statement (see below), that is, very related or identical to problems already detected. For example, the student detected that there is no a link to the homepage in four different pages and reported them as four different errors. This is also evidenced if we take some 'prominent' usability errors that are found by the majority of students, but remain unnoticed by a significant percentage of them. This initial analysis reveals problems in student's performance. An additional check of that poor achievement can be done by hypothesis testing with the two variables: *problems-found* (number of problems correctly identified and explained by each student) and *score* (the grade resulting from the evaluation of the assignment). In our case, Spearman's rank correlation coefficient evidenced the

Table 2. Overall figures of the data gathered in the case study

Course	Students	Problems identified by students (correct or not)	% Correct problems found (%CPF)	Problems found only by students (PFOS)	% Coverage of problems identified by instructors (%CPII)
2007–2008	55	520	39%	5	86%
2008–2009	30	328	42%	2	90%

positive relation of problem founds with the score for a significance level of 0.25 in both student's cohorts.

The in-depth analysis of problems found requires a categorization of mistakes. In the data discussed above, errors were classified in three categories: incorrect, redundant and vague statements.

- *Redundant* statements are considered as the repetition of the same problem found in several sections of the application (Web site, as the students were asked to evaluate that kind of application). This adds only noise to the heuristic evaluation report and should be avoided. This frequent error category reveals a methodological mistake, which can be avoided easily by emphasizing the unique nature of problem entries in the report. In this case, a clarification about redundancy was included in the heuristic evaluation report given to students, and an example emphasizing that aspect was also included in the case presented at the lab, which is discussed with the tutors.
- *Vague* statements in contrast represent a very broad category, including statements that are to some extent correct, but fail in accurately describing the problem form a usability perspective or even use clichés and stereotypes. Examples of these statements are 'the page is too interactive' (supposedly referring to the impossibility of skipping some animation, which represents a lack of user control) or 'the consistency of contents and interfaces is poor' (which fails in detailing the concrete elements that are related to that inconsistency, e.g. navigation structures, general appearance, etc.).
- Those reported errors that don't reflect real usability problems are considered incorrect statements. Examples of incorrect statements are 'The site is not WAI compliant'—where WAI refers to the Web Accessibility Initiative of W3C—or 'In page X there is no link to homepage' when that claim is not true.

Table 3 reports the overall Figures of errors in student's assignments.

An additional element to be taken into account is the coincidence in severity assessments for problems correctly identified by students with the severity indicated by the instructors. Severity assessments are critical as they determine the final recommendations for improvement, which are typically prioritized by severity. However, a degree of subjectivity is present in severity assessment even in the professional context. Table 3 provides the rate of coincidence of severity assessments (computed over the problems correctly identified). An strict coincidence reveals low coincidence; however, this can be attributed to the subjectivity of judgments about severity. However, taking an 'approximate' view of coincidence (two severity ratings are considered equal if they differ in +/– 1 in the scale) the coincidence is much higher. As severity judgments can be hypothesized to be relative to the subset of problems found by an individual, this figure seems a reasonable adjustment, so that no further inspection of possible causes was initiated.

In the case of system status (see summary in Table 4), the cause for the hypothesized errors was the absence of an explanation of the concept of system status as the important characteristics that the user needs to be aware of when changes occur. The case

Table 3. Error categories in the case study

Course	Redundant	Vague	Incorrect	Rate of severity assessment coincidence (strict)	Rate of severity assessment coincidence (approximate)
2007–2008	10%	22%	68%	30%	63%
2008–2009	4%	31%	65%	32%	59%

Table 4. Main error categories according to heuristics and principal connections identified to the contents of the course

Error category	Frequent errors	Notes
Visibility of system status	Confusion with issues related to interface layout (e.g. font size) or to the match of the system with the real world.	The notion of 'system status' from the perspective of the user was not properly understood.
Flexibility and efficiency of use	Descriptions of lack of flexibility and efficiency are vague statements in most cases.	
Consistency and standards	Non-compliance	The inspection of the errors reveals that there is confusion between accessibility and compliance to W3C WAI guidelines.

of flexibility and efficiency of use points to the inability of many students to connect the heuristic with objective measures of efficiency as an attribute of usability. While early in the course, the students are exposed to the different aspects of usability [21] and the different measures and measurement instruments for each, these are not put in relation to the heuristics presented. The case of consistency and standards come from a possible overemphasis on W3C WAI guidelines and automatic assessment of accessibility.

Once the problems in student's assignments were evaluated, the coverage of the problem was evaluated by inspecting the usability categories of the problems found by the instructors. In our case, all the Nielsen's categories were represented, except *error prevention* in the 2008–2009 course.

4.2 Devising instruction from the analyzed problems

Evidence gathered from student assignments can be used for developing new assignments and also for the development of guided or tutorial problems that are targeted to raise difficulties found by students in the past. Fig. 2 summarizes the overall process followed and the concrete steps considered in the case study on heuristic evaluation.

The analysis of error patterns can be considered as retrospective content correspondence analysis using the framework of the 3C3R model, but exercised over an empirical base of evaluation (could be considered a problem *calibration* process also). For example, the analysis done to reveal the problems with the *flexibility and efficiency of use* heuristic is a kind of connection analysis, as it reveals the interconnection of the content related to usability attributes and their measurement with the content of usability heuristics. The latter is basically a more concrete realization of the former, put in the context of the practice of expert evaluation. The inspection of coverage of the problem is related to goal setting, also in retrospection. In this case, that analysis attempts to trace back if all the important content elements are actually exercised when dealing with the problem.

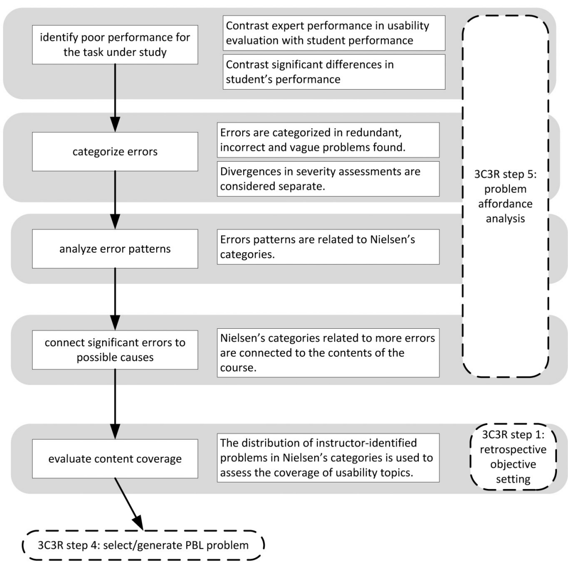

Fig. 2. Overall process for problem analysis (left), concretization in the heuristic evaluation context (middle) and mapping to 3C3R steps (dashed rounded rectangles).

Once the errors reported by students have been analyzed, teachers can fine tune the problems (examples and lab practices) to reflect in usability evaluation case studies most common misunderstandings, that is, a re-elaboration of the problems (related then to step 4 in 3C3R). In this case, the main changes suggested to the material include: (a) provision of examples that contrast the VSE heuristic with other related ones, (b) making an explicit connection of the heuristics to attributes, guidelines and principles introduced before in the course and (c) an improved explanation of the role of tools and recommendations in the case of accessibility standards. Further, the analysis revealed a methodological error related to the redundancy in problem descriptions, this was introduced as a warning in the problem statement.

5. Conclusions and outlook

Problem-based approaches to instruction require a careful design of problems or cases that exercise all the required content and skills that are desirable as objectives of the learning process. Some disciplinary areas lack reliable, mature problem collections that could be used by instructors as a point of departure. In these cases, evidence can be gathered form student assignments involving problem solving, so that new or revised problems can integrate aspects known to have been difficult to master or sources of common errors in their statement and task design. This paper has described a method for doing such analysis, and it has discussed its framing in an existing model for problem design in PBL settings. The paper has also reported the application of the method to gaining insight on student's difficulties and pitfalls when confronting heuristic usability evaluation, a high-level analysis task that requires exercising a considerable amount of principles, guidelines and rules in the field of HCI. The procedure entailed a detailed analysis of problem report entries submitted by students and their categorization, leading to three broad categories of error that entail different kinds of update of teaching material.

Future work will continue mainly in two directions. On the one hand, the error analysis for the specific setting reported will be used to improve the concrete course and to gain further insight in the main cognitive pitfalls behind heuristic evaluation of user interfaces and the supporting knowledge required for effective usability evaluation. On the other hand, the method described here will be used in other domains that present similar complexity and openness as heuristic evaluation in order to test the degree to which the method presented can be transferred to other domains.

Acknowledgements—The research is within the framework of the activity of the educational innovation project UAH/EV243, from University of Alcalá.

References

1. M. D. Merrill, A Task-Centered Instructional Strategy. *Journal of Research on Technology in Education*, **40**(1), 2007, pp. 33–50
2. C. Hsieh, L. Knight, Problem-Based Learning for Engineering Students: An Evidence-Based Comparative Study, *The Journal of Academic Librarianship*, **34**(1), 2008, pp. 25–30.
3. J. A. Colliver, Effectiveness of problem-based learning curricula: Research and theory, *Academic Medicine,* **75**(3), 2000, pp. 259–266.
4. P. A. Kirschner, J. Sweller and R. E. Clark, Why minimal guidance during instruction does not work: An analysis of the failure of constructivist, discovery, problem-based, experiential, and inquiry-based teaching, *Educational Psychologist,* **41**(2), 2006, pp. 75–86.
5. P. A. O'Neill, The role of basic sciences in a problem-based learning clinical curriculum, *Medical Education*, **34**, 2000, pp. 608–613.
6. Hung, W, The 3C3R model: A conceptual framework for designing problems in PBL, *Interdisciplinary Journal of Problem-based Learning* **1**(1), 2006, pp. 55–77.
7. W. Hung, The 9-step problem design process for problem-based learning: Application of the 3C3R model. *Educational Research Review,* **4**(2), 2009, pp. 118–141.
8. E. P. Rozanski and A. R. Haake, The many facets of HCI. In *Proceedings of the 4th Conference on information Technology Curriculum* (Lafayette, Indiana, USA, October 16–18, 2003). CITC4 '03. ACM, New York, NY, 2003, pp. 180–185.
9. H. R. Hartson, T. S. Andre and R. C. Williges, Criteria for evaluating usability evaluation methods, *International Journal of Human–Computer Interaction* **13**(4), 2001, pp. 373–410.
10. ACM/IEEE *Computer Science Curriculum 2008: An Interim Revision of CS 2001*. Available at: http://www.acm.org/education/curricula-recommendations
11. K. Simons and J. Klein, The Impact of Scaffolding and Student Achievement Levels in a Problem-based Learning Environment, *Instructional Science*, **35**(1), 2007, pp. 41–72.
12. M. B. Skov and J. Stage, Supporting problem identification in usability evaluations. In: *Proceedings of the 17th Australia Conference on Computer-Human interaction* (Canberra, Australia, November 21–25, 2005). OZCHI, **122**. Computer-Human Interaction Special Interest Group (CHISIG) of Australia, Narrabundah, Australia, 2005, pp. 1–9.
13. J. M. Carroll and M. B. Rosson, A case library for teaching usability engineering: Design rationale, development, and classroom experience, *J. Educ. Resour. Comput.* **5**(1), 2005,
14. J. Nielsen, Finding usability problems through heuristic evaluation. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Monterey, California, United States, May 03–07, 1992). P. Bauersfeld, J. Bennett, and G. Lynch, Eds. CHI '92. ACM, New York, NY, 373–380.
15. R. Molich and J. Nielsen, Improving a human-computer dialogue, *Communications of the ACM*, **33**(3), (March), 1990, pp. 338–348.
16. Usability.Gov (consulted in 2010) *Research-Based Web Design & Usability Guidelines*. Available at http://www.usability.gov/guidelines/guidelines_book.pdf
17. J. Nielsen and R. Molich, Heuristic evaluation of user interfaces, *Proceedings of the Conference on Human Factors in Computing Systems (CHI 90)*, Seattle, WA, 1990, pp. 249–256.
18. E. T. Hvannberg, E. Lai-Chong and M. K. Larusdottir, Heuristic evaluation: Comparing ways of finding and reporting usability problems, *Interacting with Computers*, **19**(2), 2008, pp. 225–240.
19. K. Hornbaek and H. Frokjaer, Comparison of techniques for matching of usability problem descriptions, *Interacting with Computers*, 2008, **20**(6), pp. 505–514.
20. J. Nielsen and T. K. Landauer, A mathematical model of the finding of usability problems. In *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems* (Amsterdam, The Netherlands). ACM, New York, NY, 19193, pp. 206–213.
21. M. Welie, G. van der Veer and A. Elins, Breaking down usability. *Proc. International Conference on Human-Computer Interaction INTERACT99*. IOS Press, 1999, pp. 613–620.

Elena García-Barriocanal obtained a university degree in Computer Science from the Pontifical University of Salamanca and a PhD from the Computer Science Department of the University of Alcalá. She joined the Computer Science Department of University of Alcaláand currently she is associate professor at that department. She is a member of the Information Engineering Research Group of this University. Her research interests mainly focus on topics related to the role of knowledge representation in fields like human-computer interaction, learning technologies and Semantic Web; concretely she actively works on ontological aspects both in e-learning and in usability and accessibility.

Salvador Sanchez-Alonso obtained a university degree in Computer Science from the Pontifical University of Salamanca (Spain) in 1997, and a PhD. in Computing in 2005 from the Polytechnic University of Madrid (Spain). He has worked as an assistant professor at the Pontifical University of Salamanca from 1997 to 2000 and again from 2002 to 2005. He has also worked as a software engineer at a software solutions company during 2000 and 2001. He is currently an associate professor of the Computer Science Department of University of Alcalá (Spain). His research interests include Learning objects reusability, Metadata, Ontologies, Web Engineering and Object-Oriented Technologies.

Daniel Rodríguez-García is currently a lecturer in the Computer Science Department at the University of Alcalá, Madrid, Spain. Daniel is also a seasonal lecturer at the Master degree of the University of Reading covering the Data Mining and Information Retrieval module and a consultant at the Universitat Oberta de Catalunya (UOC) for the Compilers module. He obtained his degree in Computer Science at the University of the Basque Country and PhD degree at the University of Reading in 2003. His research insterest include Software Engineering in general and the application of data mining techniques to Soft Eng in particular. He is also a member of IEEE and ACM and associations. He is also secretary of the Spanish Chapter of the ACM Special Interest Group in Computing Education (SIGCSE).