Development-oriented eLearning Tool Evaluation:
the Edukalibre Approach

Luca Botturi – University of Lugano
Vania Dimitrova, Julika Matravers, Chris Tebb, Drew Withworth – University of Leeds, UK
Jutta Geldermann, Isabelle Hubert – University of Karlsruhe, Germany

Abstract. The Edukalibre development approach described here offers libre educational tools based on open development model. The paper presents a development-oriented evaluation approach comprising three phases. Phase 1 focused on rapid and distributed usability inspections producing development-oriented feedback for designers, and was built around a usability inspection framework. Phase 2 was conducted through student feedback, and confirmed the results of Phase 1, along with indications about the reasons behind some issues and about the actual use of the system. Phase 3 was aimed at a broader holistic evaluation that considers the views that developers and other parties have of the system. The evaluation approach proposed in this paper was applied to two eLearning tools – ConDOR and GISMO - and the results so far have provided fast and constructive feedback into the next round of development. This paper arose from the recognition that a lot more work is required to address the issue of suitable and complete evaluation for libre software development for education. The evaluation approach proposed here offers a significant step towards addressing these emerging needs.

Introduction: The Evaluation of eLearning Tools

The increasing amount of Open Source eLearning solutions has recently provided the educational sector with access to flexible alternatives to the traditionally expensive commercial products which many institutions cannot afford. With the rapid invasion of these technologies into educational institutions, the evaluation of these systems has become increasingly important. Many of these technologies follow the Open Development Model which relies on the constant improvement of software products through the modification by a wide range of developers. The provision of constructive feedback is crucial to the success of such a novel development mentality that is *highly iterative* with notably *frequent software releases* and a *geographically distributed development team*. The development of a suitable framework for the comprehensive evaluation of eLearning tools that follow the open development model remains a challenge.

Typically eLearning tools have been evaluated on the basis of particular aspects of a system such as their costs, their technical specification, the features they provide and their usability (Britain & Liber, 2000). The importance of usability evaluation, in particular, is frequently stressed in the literature. Magoulas et al (2003), for example, propose the use of heuristic evaluation within a layered evaluation approach for usability testing within Adaptive Learning Environments. Research also seems to have given priority to the evaluation of system usability against the background that the evaluation of the effectiveness of a learning environment remains difficult (Strother 2002). Furthermore, it can be argued that good usability is an essential feature of a successful eLearning system. The level of a system's usability may not only influence the user acceptance of a system but may also greatly influence the learning process (Balacheff & Kaput, 1996). Whilst it needs to be recognised that good system usability is crucial for the user acceptance of an eLearning system, it needs to be acknowledged that usability is a necessary, but not a sufficient, condition for an effective learning environment (Pulichino, 2004). A further important and difficult issue, for example, which is often not directly addressed by usability evaluation is the evaluation of functional requirements (including the identification of missing requirements) for an Elearning system from the range of system users (O'Droma et al 2003).

It is the objective of this paper to propose an evaluation approach for eLearning applications that not only recognizes the need for a high level of a system's usability, but at the same time opens up scope for a more *holistic evaluation approach* that accounts for, and supports, the *rapid evolution of Open Source Software* and is tailored to the development process of libre educational applications. Such an evaluation approach addresses both the behavioural aspects and the system's architecture, thereby aiming to support the rapid development and refinement of eLearning systems by producing design-oriented feedback for developers.

The evaluation approach presented in this paper was developed within the framework of the Edukalibre project (Gonzalez-Barahona et Al., 2005; Edukalibre, 2005), as a method for assessing the tools produced in the project and at the same time to develop refinement guidelines and indications for the developers. For this reason the next section is devoted to a short introduction of the project and to the description of two eLearning tools that were evaluated. The third section will describe the approach in its general outline, and will then provide details for each phase, along with the results for the selected tools. Finally, a summary will provide some conclusions and outlooks for further work.

The Edukalibre System

Edukalibre is an EU funded project that examines the connection between libre software development and creation of open content for education. The project involves developers and educationalists from several European countries (Spain, Portugal, Germany, UK, Switzerland, and Check Republic) working together to study both technological and pedagogical aspects of the successful deployment of the libre idea in university teaching. Until recently, technical challenges made it very difficult to support truly open, dynamic, educational resources constructed collaboratively by large groups of teachers and even students. New systems are needed that effectively support the collaborative construction of open educational resources on the web (Noel, 2004). One such system, composed of several key tools and based on the libre methodology, has been developed within the framework of the Edukalibre project.

The core of the system is the repository manager, written in python, which uses the subversion version control system. The system accepts documents in OpenOffice, LaTeX or docbook formats, automatically version control the documents, and provides output versions in multiple formats (currently xhtml, docbook XML, text and PDF). This repository manager has been designed to be independant of the user facing applications, allowing us to use openoffice as our interface (working directly with the webdav repository) or web based collaboration software. We have created two web based systems, one simple document management tool called COLLAB and COnDOR, a groupware tool which has been created as a module of the open source learning environment Moodle[1]. Moreover, a visual user-tracking tool was developed, again for the Moodle environment, called GISMO (Mazza & Milani, 2004). The evaluation presented in this paper focuses mainly on the use of ConDOR and GISMO, which are outlined below.

ConDOR: Construction of Dynamic Open Resources. ConDOR is a bespoke groupware application which runs as a modular component of the open software learning management environment Moodle. ConDOR was designed as a tool to be used by students or teachers for the collaborative construction of learning resources. The groupware application allows easy "explorer style"' access to the document repository, allowing intuitive navigation of the file and folder structures, and simple tools for uploading files and creating new folders. Figure 1 shows a screen with the ConDOR interface. To prevent excessive repeated navigation of large folder structures, a *My Resources* section is provided within the groupware resource area to allow authors to bookmark documents deep within a resource tree of any repository. This allows for a simple list of documents the user has an active interest in.
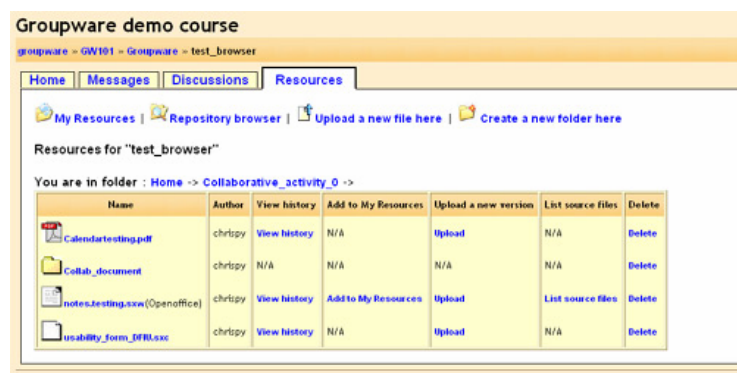
From either the main *Resources* browser or the *My Resources* bookmark area, authors can download a document in any supported format (see above), upload a new version of the document, browse the history of the document and add/remove documents from their "My Resources" list. Extra features such as format conversion are only available for OpenOffice and LaTeX documents, but trial users have been using the repository

---

[1] See http://moodle.org

Figure 1 - Screenshot of ConDOR – tool for collaborative construction of dynamic open learning respurces

to manage many kinds of file.

GISMO: Graphical Interactive Students MOnitoring. Student tracking data provided by Moodle is a valuable source of data that can be used by the instructor not only to check students activities, but also to improve quality of the materials. For instance, an instructor may check which part of the course's materials are the most and least accessed by the student, and then perform further investigations to understand whether the students found these parts difficult to understand or not. Furthermore, the students activities in ConDOR are recorded and can be used to analyse the collaborative work of a group of students or teachers who are constructing resources using the Edukalibre system.
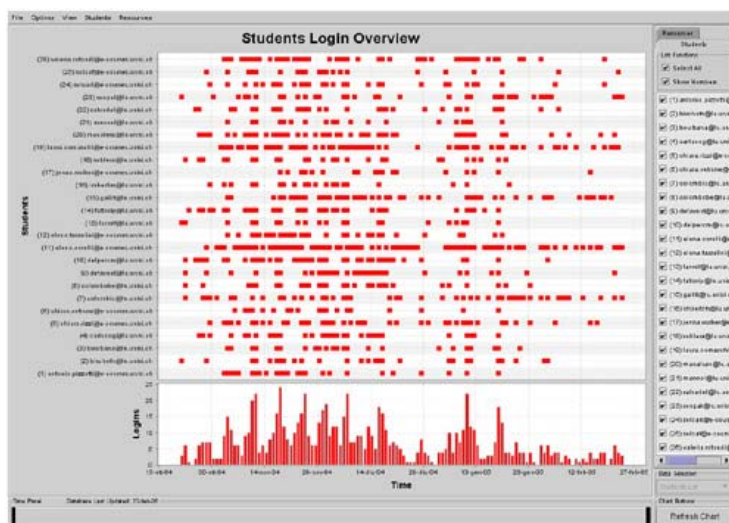

Student tracking data is complex and is usually organized in some form of a tabular format, which is in most of the cases difficult to follow and inappropriate for the instructors' needs. GISMO uses the students' tracking data as source data, and generates graphical representations that can be explored and manipulated by course instructors to examine social, cognitive, and behavioral aspects of both distance learning and conventionally taught students. Instructors can investigate the users' reading of course materials, which – at a glance - can give feedback on the quality of collaboratively constructed resources and their practical use by students.

Figure 2 - Screenshot of GISMO – tool for monitoring the students' work in both Moodle and ConDOR.

The Edukalibre Evaluation Approach

Challenges to the Edukalibre evaluation

The specifics of the open development model brought some challenges that had to be dealt with during the evaluation. In the first place, the whole design and development cycle was iterative, including frequent release of software that was sometimes not fully tested. Moreover, the development, as well as the evaluation, involved several geographically distributed teams. The advantage of this is the quick deployment in various settings and the wide dissemination. The downside is that the effort has to be distributed and the work properly synchronised. As a result, several small scale evaluative studies appear to be more appropriate than one large evaluation. Furthermore, the development was done asynchronously at times with several developer teams working independently in several countries (Spain, Leeds, Switzerland, and Portugal). This led to the need to be able to conduct evaluation in a flexible way to enable examining different tools within the appropriate context. These constraints had to be taken into account in conducting the Edukalibre evaluation, as discussed in the following sections.

Another issue that challenged the definition of an evaluation approach for Edukalibre was the relationship between software application design and development and the evaluation itself. A complete evaluation that looks for comprehensive data, requires real users in real setting and stable tools in their final version. Such an approach produces sound *ex-post* results which might be used to promote the tools or to improve the design in the future. Edukalibre, given its libre nature, had different needs: on the one hand, developers

needed short term input and guidance relating to their development work in order to improve their tools; on the other, the project team felt the need of a wide evaluation of the impact of the tools in a real higher education setting in order to evaluate the usefulness and effectiveness of their product.

Once a fairly stable version of the system was produced, it had to be evaluated. It was very important to consider all aspects:

- *Flexibility* – the tools should be useful for a wide range of users, e.g. teachers or students.
- *Effectiveness* - the tools should be applicabile in learning scenarios, and required several iterations to ensure that applications complied with the tasks they were intended for;
- *Efficiency* - the users should be able to perform their tasks quickly, cumbersome operations and long navigation menus should be avoided;
- *User satisfaction* – the tools should be subjectively pleasing.

Evaluation outline

Based on these goals and taking into account the specific needs of the libre development model, we developed a three-phase evaluation approach.
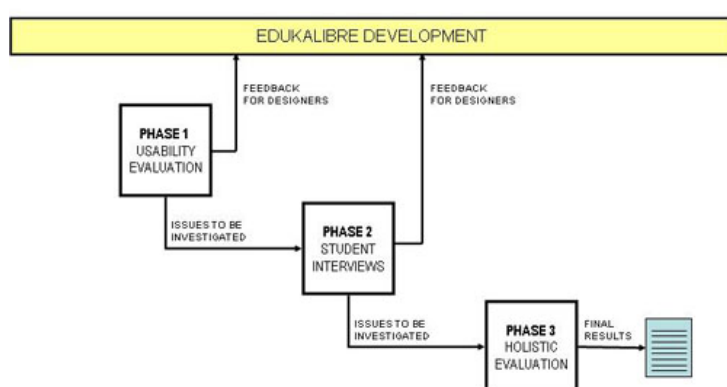

Figure 3 - Evaluation approach outline

Phase 1, conducted between October and November 2004, included several studies involving experts and users from different partner institutions and focused on usability. Phase 2, conducted in December 2004, focused on student feedback, in order to verify the results of phase 1 and to investigate the reasons for the problems detected during the usability inspection. Phase 1 and 2 were then complemented with a broader evaluation in Phase 3 to match the expectations of both developers and users. This phase has commenced and will complete at the end of the project (end of 2005).

The three phases are discussed in detail in the following sections. For each phase, a general outline is followed by some examples of the kind of feedback produced, along with its impact on the project.

Phase 1: Rapid Usability Inspection

The MiLE Usability Method. Usability evaluation is a narrow-scope assessment of how satisfactorily users can achieve specific goals with a web application. With the Edukalibre framwork, usability evalaution was selected as (a) it can be conducted in a short time, so to keep the pace with the needs of the developers; (b) it produces design-oriented feedback useful for the refinement of the application; and (c) it can be conducted in a uniform way by different teams in a libre fashion, if a specific methodology is followed. In order to meet the conditions expressed under (c), the MiLE method was selected. MiLE (Milano-Lugano Evaluation method) is an experience-based evaluation framework for web applications that strikes a healthy balance between heuristic evaluation and task-driven techniques (Triacca et Al., 2004). MiLE can be used flexibly at different levels of granularity. It offers reusable tools and procedures to carry out both expert reviews (called *inspections*) and user testing within budget and time constraints.

MiLE uses scenario analysis as the driver for usability evaluation. Scenarios are then combined with selected usability heuristics in order to provide analytic feedback on the different aspects of the application design.

The MiLE method moves through five stages (see Figure 4). Stage 1 is analysis of the application requirements, in order to get information for constructing goals, scenarios and user types. Stage 2 defines macro scenarios (i.e., broad "stories about use" - cfr. Carrol, 2002) and user types (i.e., profiles of typical users of the application). Stage 3 defines scenarios (i.e., smaller-scope use cases), goals for each scenario, and tasks for each goal. The next two stages form the evaluation itself. Stage 4 is the expert inspection, guided by scenarios, goals, task and MiLE heuristics, which provides the identification of main issues. It can be combined with user testing, in order to confirm and further investigate the issues. Finally, in Stage 5 produces reporting and guidelines development.
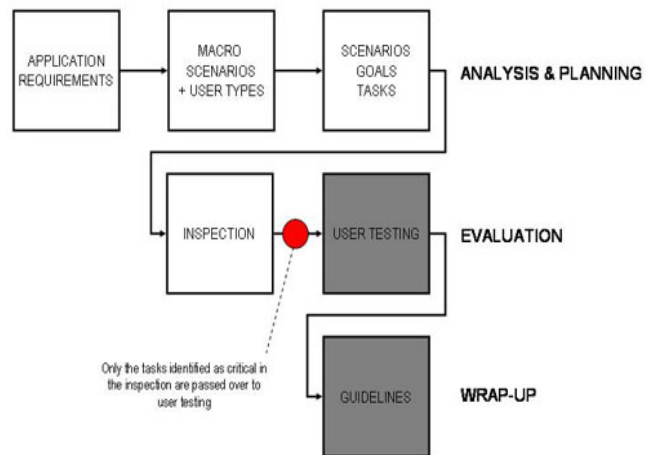

Figure 4 - The MiLE inspection method

The outcome of the steps in the analysis and planning phases make it possible to define a usability protocol that can be replicated by different teams, so that the work can be distributed and then the results collected and merged. Although originally developed for hypermedia web applications (not specifically for Elearning), MiLE is now attracting increasing interests for its application to Elearning environments.

MiLE for Edukalibre. Given the goals for the Edukalibre project expressed above, MiLE was slightly adapted. The usability framework (i.e., application requirements, scenarios, goals and tasks) was developed from the application requirements expressed by the developers' documentation (which were in turn gathered from user requirements). Scenarios were developed both for teachers and students, see Figure 5.
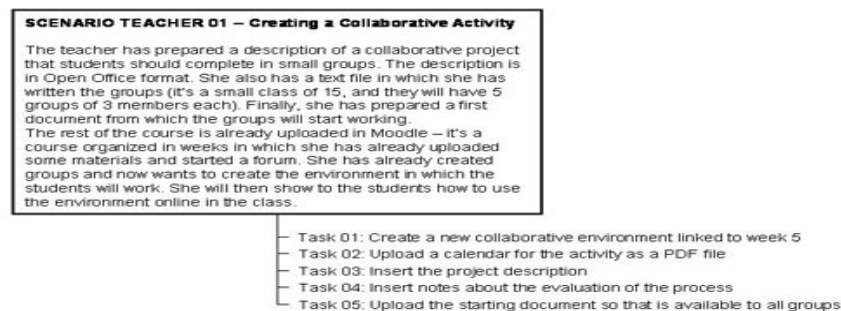

Figure 5 Sample of scenario and tasks

Usability involves a number of different issues. For this reason MiLE proposes to articulate usability judgments around dimensions. The following set were considered for the tools described above:

- *Content*: is the information provided by the software complete, clear and accurate concerning the operations that the user is trying to accomplish?
- *Orientation*: does the user understand where s/he is, at what step in a given process or in what position in the application?
- *Navigation*: can the user easily reach the tool/page/content s/he is looking for? Can s/he identify the right path to it?
- *Predictability*: is the user able to know what is going to happen when s/he clicks on a link/button? Does s/he get surprises?
- *Layout*: is the layout consistent? Can users identify functional areas that support easy interaction?
- *Legibility/Graphics*: can the user read the content? Are the graphics useful for working with the application or are they confusing?

The usability framework and the usability dimensions were put together in a single document, which was then distributed to all project partners, along with indications about how to conduct the inspection. Each partner was in charge of assessing the usability of its module and of one other. The results were collected with a standard usability matrix (one mark on a 5-point scale for each task/dimension), to be sent to the Lugano team, which was in charge of collating the data and producing the final guidelines. The Leeds team set up Moodle courses that reflected the scenarios and were used for inspection. GISMO was evaluated by 3 inspectors (a teacher scenario) and ConDOR by 6 inspectors (3 teacher and 3 student scenarios).

Results. In order to provide the flavour of the results obtained with MiLE in Edukalibre, and their impact on design, two sample results are reported below.

Inspecting GISMO, an issue emerged concerning orientation and navigation for Task 02: See who logged in during last week. Actually, in order to see logins, the teacher should select *draw chart* and then *login overview*, which is not very straightforward.

The identified solution is to organise menu semantically (i.e., following the teacher's perspective), and not operationally (i.e., following the programmer's perspective). Figure 6 shows the outline of the GISMO main menu before and after the usability inspection.
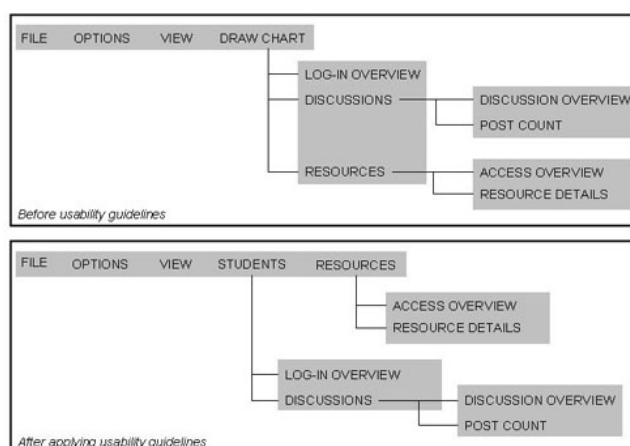

Figure 6 Usability improvement for GISMO menu

For COnDOR, an issue was identified concerning Task 08: Identify the least and most productive contributors of Group 1 (a group in the scenario), concerning the content dimension. Actually, the COnDOR interface is document-oriented, and in order to track the activity of single students, the teacher had to browse and count messages/posts manually. This was indeed a common task among teachers, which was not supported by the tool. It was therefore included in the plan for next development to provide summary information about the activities of each group member (including the number of posts and file contributions, the date of the last contribution, etc.).

Usability evaluation provided useful indications for further development – yet it is by nature not complete, as it only focuses on a specific feature of the system, and it does not consider its educational impact and its effectiveness. This is the reason why Edukalibre moved forward to the next two phases.

Phase 2: Student Feedback

In order to verify the results of the rapid usability inspection (phase 1), student feedback was gathered in phase 2. At the University of Karlsruhe, a seminar was organized with 14 German students in the advanced study period of Industrial Engineering. During four months, the students elaborated an environmental business game, which comprised computer supported case studies for location planning and technique assessment. Four groups of three students worked on a specified topic, representing fictitious companies faced with strategic questions and the subsequent decisions. In addition, two individual topics were related to business games and open source software (Geldermann et al., 2004).

During the preparation of the business game, the students were invited to use the Edukalibre platform that was specifically set up for them. The platform comprised of the general moodle platform (with news forum, calendar, documents etc.) plus the developments of Edukalibre up to that date: one groupware tool for each group as well as a direct link to the conversion platform. The students were very familiar with Microsoft applications and had some experience with programming, but were only vaguely aware of the concept of

open source software. The web and e-mail suite used was Mozilla (Firefox) - open source programme well known among the users, who however were unaware of the open source concept behind the software.

Data collection. For producing development-oriented feedback, a questionnaire was used with altogether 15 main questions, developed according to the issues identified in Phase 1. Some questions served to evaluate the background of the students, while for some questions, the 7-point Likert scale was used. Finally, open questions gave the students the opportunity to make specific comments.

Results. Among the 14 users of the platform, 8 logged in weekly and 6 hardly ever. All the users seldom used the groupware and rarely downloaded a file. The groupware was mostly used to download documents, sometimes to upload files, to try to communicate within their groups or for the chat. However, apparently the students used the features of Moodle (calendar, news forum, download files) more than the groupware, largely due to the Moodle based organisation of the class.

The system was not too difficult to use for most of the students. The uploading time was not judged too long either. The best features of the system according to the students are the calendar, the news session, the repository browser (however extremely hidden – a finding that emerged also in Phase 1) and the multiple download formats offered by COnDOR. The worst features cited by the students are the messaging that apparently did not work all the time. Moreover, a lack of feedback by the platform was found faulty. Accordingly, the missing feature was an e-mail notification of the messages sent or received or when getting an answer to their own posts.

It can be seen from the features that were not discovered by the user group that they apparently had little initiative to search for what they needed. The students did not seem very keen about having to use an Internet platform for the purpose of the assignment they had to make. Maybe the type of course used to test the platform was not an ideal case, but the Edukalibre platform is being designed for such a kind of seminar. Indeed, the groups are made of only three students and they prefer to meet regularly instead of using a platform. In a way, the students seemed to be conservative. Although they are eager to try different kinds of learning such as a "business game seminar", they seem unsettled when the learning experience differs from their routine (e.g. an Internet platform and Open source against MS Office).

In addition, the terms resources or shared resources were probably understood as "documents provided by the teachers" more than "my resources" signifying "our group documents" in the resource browser of the groupware. When almost all the students agree about the importance of the shared resources, they are more cautious about the idea of collaborative authoring, especially the fact that other people can modify what they write. Although they like to see what others have written, they are not really enthusiastic about people reading and modifying their work (see Table 1).

Table 1: Attitude towards shared resources

	1 – Strongly disagree 4 – Neutral 7 – Strongly agree						
	1	2	3	4	5	6	7
The idea about having shared resources is great		1	1	1	1	5	4
I like to see what others write	1		2	2	3	3	3
I like others to see what I write	1	1	1	4	3	1	2
I am happy to have others make changes to what I have written	1		5	2	3	1	1

In conclusion, the students of this first test group were not totally convinced of the benefits of any Elearning platform mainly because they can meet regularly and attend "traditional lectures". They did not use the platform regularly, which explains most of the lukewarm answers found in this first questionnaire. More documentation such as a tutorial for the Edukalibre-platform would be beneficial. Yet, the idea is growing and the acceptance and extensive use of such a platform by the students could arise soon.

Phase 3: Matching the expectations of developers and users

The main idea for Phase 3 is to investigate the developers and find out if and how their assumptions match

with those of the users of their applications – students and teachers. Why is it necessary to investigate the developers? Although powerful for delivering results during development, we believe that standard HCI recommendations for usability testing are inherently limited in terms of what they can reveal. HCI can ask how well "objective" usability standards are being maintained, how accurately the needs of users are being assessed and how effectively the insights gained are translated into system functionality. However, usability problems are, at heart, conflicts between the assumptions of the developers and the needs of the users. These may emerge because the needs of users have not been adequately researched: here, traditional HCI (as summarised above) can help. However, they may also emerge because of inherent differences in how different groups perceive a system, and this can apply both to the specifics of a system (functionality), and generalities (such as why the system was created in the first place). It is not a case of lacking the right answers; evaluators may not even be asking the right questions. In the first place, many usability evaluations do not take place in the full learning context (Lim, Benbasat & Todd, 1996). Secondly, task-driven evaluation is limited to assumptions made by the designer so will struggle to test unexpected and creative use of the system. Nor do these methods consider technologies as being dynamic, existing in a constant state of evolution (here see Star 1999).

In an initial survey as a first step in Phase 3, developers were asked what, in their opinion, would constitute "success" for eLearning generally and Edukalibre specifically. The answers were as follows:

- In general, a new level of adaptability to users and flexibility; for Edukalibre, achieving project goals and fulfilling the set tasks.
- for Elearning in general: complementary to face-to-face teaching; further enhances learning; enables better engagement of students; interactivity; gives opportunities for immediate feedback from the teacher; for Edukalibre: If we manage to deploy it in real courses.
- Involve learners constructively; good popularity among instructors; add new benefits not given by face-to-face modality; at least 20 courses with materials collaboratively produced with this tool.
- Here is no general rule, but I would list: easy to use, simple to explain to the students, answering a direct need, stable and having support. The same for Edukalibre.
- Success in Elearning projects is when the users of the system (teachers and students basically) can use the system easily and take all the advantages of the system without any extra effort to learn the normal use of the system.
- Elearning in general: If it will be generally used; for Edukalibre, if a functional system is created.
- Success in Elearning is wide adoption of the application, an improvement in the quality of teaching, and a reduction in the time teachers spend doing admin. For Edukalibre, it is wide adoption of the tools we have produced.

There is considerable variety in these answers. Some refer to specific pedagogical issues, but some are quite general ("enhances learning", "improvement in quality of teaching") and some are not pedagogical at all ("adoption of the system"). Taking this latter point as symptomatic of the issue, a desire to have a new technology adopted by a number of institutions is a very different motivation for creating that technology than enhancing teaching. It may also require developers to write the technology in particular ways. Certain technological elements may have been included in Edukalibre because of the desire of developers to "play" with certain technologies and innovate in order to create something substantially different from the competition. In their own cognitive environment, this is understandable - it is through engagement with new technologies that developers acquire status, job satisfaction, and so on. But simply including new technologies is not directly addressing the educational needs of students (or teachers). Any teaching which uses the system may be forced to change to fit the system, instead of the reverse - pedagogical needs driving technological development.

This is not meant to tar Edukalibre with the brush of commercialism: it is merely an example of the potential ways in which Elearning technologies, being complex and significant to a wide variety of stakeholders (Benson, 2002), can become an arena which different interests and perspectives compete to define. The "social shaping of technology" thesis (Mackenzie & Wajcman, 1985) suggests this.

The ideal is first that assumptions are revealed by self-reflection; second, that they can if necessary be challenged and changed. This has been termed "double-loop learning" (Argyris, 1999). Ideally, this takes place in a continuous cycle of evaluation then subsequently, implementation. The more user needs can be

included and underlying assumptions revealed, the more participatory the construction of the environment will become. This is a driving force behind the project in the first place (thanks to its basis in open source software and collaborative content) so the method is appropriate to the system.

Conclusion

Open Source eLearning applications have become increasingly common in today's educational environment. However, the lack of suitable evaluation approaches - that extend beyond the assessment of a system's usability - to provide developers of eLearning applications with constructive feedback remains an issue. This issue now extends further into a world where the Open Development Model has become a dominant development mentality. It was the objective of this paper to propose an evaluation approach for eLearning systems that follow the Open Development Model. It was argued that such an evaluation approach would have to be holistic in the sense that it offers feedback beyond issues of usability. Furthermore, it should be applicable within a development environment that was highly iterative with frequent releases, and that potentially involved a high number of geographically dispersed developers of various backgrounds (including students and teachers).

It was not the intention of this paper to provide a recipe for the complete evaluation of eLearning systems. However, this paper recognizes that software development projects normally have an ambitious time frame, and results of ex-post evaluation can hardly be taken into account in full. The evaluation approach proposed here supports a holistic evaluation of Open Source Software that accounts for these factors. The approach was used with fairly stable versions of a learning system that was developed following the Open Development Model, and we were able to trial our approach within real higher education teaching situations. Both students and teachers were highly motivated to use and test the software, because they were conscious of the significance and immediate relevance of their feedback. This paper arose from the recognition that a lot more work is required to address the issue of suitable and complete evaluation for Libre Software Development. The evaluation approach proposed here offers a significant step towards addressing these emerging needs.

References

Argyris, C. (1999). *On Organizational Learning.* Oxford, UK: Blackwell.

Balacheff, N., & Kaput, J. (1996). Computer-based learning environments in Mathematics. In A. J. Bishop et Al. (eds), *International Handbook of Mathematics Education*, Kluwer, Amsterdam, 469-501.

Benson, A. (2002). Using Online Learning to Meet Workforce Demand: A Case Study of Stakeholder Influence. *Quarterly Review of Distance Education*, 3(4), 443-452.

Britain, S., & Liber, O. (2000). *A Framework for the Pedagogical Evaluation of Virtual Learning Environments*. JTAP Report No. 041.

Carroll, J. (2002). *Making Use-Scenario-based design of Human-Computer Interactions.* Cambridge, MA: MIT Press.

Edukalibre (2005). Edukalibre's Website. Retrived on February 23rd, 2005, from www.edukalibre.org.

Geldermann, J., Treitz, M., & Rentz, O.(2004). Teaching Integrated Technique Assessment based on Open-Source Software. *EUNITE Conference: European Symposium on Intelligent Technologies, Hybrid Systems and their implementation on Smart Adaptive Systems, Aachen, Germany, 10.-12.06.2004,* PP 503-508.

Gonzalez-Barahona, J. M., Dimitrova, V., Chaparro, D., & Mazza, R. (2005). Producing educational resources in the "libre" way: The Edukalibre project. Edukalibre technical report.

Lim, K. H., Benbasat, I. & Todd, P. A. (1996). An experimental investigation of the interactive effects of interface style, instructions and task familiarity on user performance. *ACM Trans. Computer-Human Interaction*, 3(1), 1-37.

Mackenzie, D., & Wajcman, J. (Eds.) (1985). *The Social Shaping of Technology.* Buckingham, UK: Open University Press.

Magoulas, G. D., Chen, S. Y., & Papanikolaou, K. A. (2003). Integrating Layered and Heuristic Evaluation for Adaptive Learning Environments. In S. Weibelzahl and A. Paramythis (eds.), *Proceedings of the Second Workshop on Empirical Evaluation of Adaptive Systems*, held at the 9th International Conference on User Modeling UM2003, Pittsburgh, 5-14.

Mazza, R., & Milani, C. (2004). GISMO: a Graphical Interactive Student Monitoring System for Moodle. Paper presented at the *UK and International Moodle User Conference - MoodleMoot 04*, Oxford, UK.

Noel, S. & Robert, J.-M.n (2004). Empirical study on collaborative writing: What do co-authors do, use, and like? Computer Supported Cooperative Work. *The Journal of Collaborative Computing, 13* (1), 63-89.

O'Droma, M. S., Ganchev, I., & McDonnell, F. (2003). Architectual and functional design and evaluation of Elearning VUIS based on the proposed IEEE LTSA reference model. *The Internet and Higher Education*, 6, 263-276.

Pulichino, J. (2004). *Usability and eLearning*. eLearningGuild Report.

Star, S. L. (1999). The Ethnography of Infrastructure. *American Behavioral Scientist*, 43(3), 377-391.

Strother, J. (2002). An Assessment of the Effectiveness of Elearning in Corporate Training Programs. *International Review of Research in Open and Distance Learning*, 3(1).

Triacca, L., Bolchini, D., Botturi, L., & Inversini, A. (2004). MiLE: Systematic Usability Evaluation for Elearning Web Applications. *EDMEDIA 2004, Lugano, Switzerland,* 4398-4405.