In: Stephanidis, C. (Ed.): Proceedings of HCI International, Vol. 4: Poster Sessions, 2001. London : Lawrence Erlbaum Associates, S. 229-231

A Task- and Scenario-Based Analysis and Design Method for User-Centered Systems

Huberta Kritzenberger, Michael Herczeg

Institute for Multimedia and Interactive Systems, University of Luebeck, Seelandstrasse 1a, D-23569 Luebeck

ABSTRACT

Software development processes need to cover usability relevant information. Although several methods have been proposed how to collect usability relevant information, it is still an unsolved problem how to deal with this information and to ensure its availability and proper management at all relevant stages in the development process. This paper provides a framework to structure all relevant aspects concerning hypermedia learning systems and to link usability knowledge with domain knowledge. The framework also supports distributed design teams in framing, storing and communicating knowledge item. It is implemented with an XML-based database and a web-front-end.

1. INTRODUCTION

The process of lifelong learning changes the requirements and demands on usability of educational environments for different user groups and different usage contexts. Software developers have to find concepts and techniques to make software as flexible and interactive as to adapt to these altering demand levels. There has been a rapid evolution of learning technologies in the past decade. It involved instructional principles, learning theories, methods of framing the learning process or presentation principles for educational hypermedia. From the field of human factor analysis one basic idea of the development procedure (ISO 13407) is not to use standard requirements for all software products of some kind, but to derive requirements from individual analysis. Use cases and scenarios (Carroll, 1995; Carroll, 2000) may help to make requirements definition easier on the one hand. However, on the other hand a huge amount of informal data will be gathered from scenario description (Holtzblatt/Beyer, 1993) and usable tools for managing and adapting these data to different levels of usage are missing. As a consequence, the different stages in the development process (e.g. analysis of user and task (with scenario definition) and requirements definition and so on) are widely separated and therefore resulting in inconsistent conceptual models. To improve the development process and to overcome at least some of the problems described above, we propose a framework layer, which is based on the concept of object-oriented system design (OOA and OOD) (Herczeg, 1999; Herczeg, 2001). It is a generic model, which covers the specific characteristics of the user population as well as the situational factors and the process of using the software.

The experience of the authors is based on two projects concerned with the development of web-based course material for virtual universities: “Virtual University of Applied Sciences” (vfh) www.vfh.de (1998-2003) and “Distance Education in Medical Computer Science” (1999-2001). The authors are involved in the design of user-centered learning spaces and in the support of the design process (usability recommendations and quality management).

2. A FRAMEWORK FOR USER- AND USAGE-CENTERED DESIGN OF EDUCATIONAL HYPERMEDIA

The first step in the development of usable software is the analysis of the user and of the task (Herczeg, 1994). For the analysis of learning environments this means the definition of the pedagogic goals, the organizational context, the context of use and the characteristics of the intended user group(s) (learners). If this analysis is done accurately, a lot of data will be gathered. In order to use them in context, structured ways of description and storage will be necessary for which the following framework seems to be appropriate.

The described framework is a generic model (Herczeg, 1999; Herczeg 2001) which structures the main dimensions of user, task and context of use. The task analysis framework consists of a set of objects in order to build a model of the context as well as a functional model of the software product. It consists of the following entities: managed learning objects (defining the resources and modules), agents (defining users and their profiles), roles (agents may

be active in more than one role), tools (support the execution of tasks), tasks (define tools, have to be supported by those agents, who use the system according to their roles).

Managed learning object: The domain knowledge is stored in basic modules, so-called learning objects, which provide flexibility for presenting them to the users (learners) during run-time. So the selection of a sub-set of knowledge items at a certain stage of use depends on the characteristics of the learner. Different learner groups (Kritzenberger/Herczeg, 2000) might be distinguished in this context (see role), to whom different sub-sets of domain knowledge are presented.

Tasks: are defined by the work to be performed by the learners, that is by models of the learning process. There are higher levels of analysis, like “facts”, “exercises”, “rules”, “problem solving” and lower levels, which more precisely specify sub-tasks. Examples are the “principles for the elaboration of a sequence of knowledge items” and other steps to complete the learning process.

Agent: covers descriptions of specific target groups (learner groups), who may be diverse, if learning environments serve the purpose of lifelong learning. For example one user group might want to do basic studies and need all modules relevant for the curriculum. Another group might only want to solve some problem at hand and fill some knowledge gap. Many other groups and their respective characteristics might be defined for teaching and addressing the specific needs of these groups.

Role: specifies the context of use by the actual learner, her or his tasks and learning context, derived from the learning history. Therefore, the role realizes concrete user characteristics of the potential user groups defined in the entity “agent”.

The framework described above allows designers and developers to concentrate on relevant aspects of their work by defining the relevant design knowledge of learning environments and it has a potential to be used during run-time for task-driven adaptation of the learning environment to the user needs.

3. RESULTS

The framework described above is implemented with a relational database, XML-based data sets and a web-front-end (Kutsche, 2000). This technical basis is capable of meeting the requirements of using design specifications (user characteristics and usage information) for the adaptation of the learning environment during run-time. The task model is useful for designing more user- and task oriented interactions, because what was specified in the task model during the design process and keeps this information in a consistent way for later re-consideration. In this context the collection of hypermedia nodes presented to learners is a customized collection of knowledge objects fitting the specific user needs and the situation of use at hand. The criteria for the selection and presentation of the knowledge items to the learner was built-in meta information (e.g. specification of the learning task, teaching strategies relevant for learner groups and types of individual learners) linked together during the design process.

Furthermore, the framework allows distributed design teams to cooperate during the development process, as the implementation is web-based, uses XML-based data records and object-oriented techniques, which allow the handling of large data collections. The enhancement of object classes with special attributes for context of use information, e.g. describing user attributes, user goals or usability data (like information from scenarios), provides with higher flexibility for data selection. As the size and number of the usability data are normally hard to handle, in this model the task and the information derived from scenarios can be described at different levels of abstraction, using e.g. refinement or aggregation hierarchies, where only certain classes of information might be available for certain contexts and specifications (Hartwig/Kritzenberger/Herczeg, 2000).

Another helpful quality of XML-based relational databases for the development process is, that it can be easily translated into other SGML-type languages like XHTML or LaTeX in order to visualize and document the contents, if necessary. As the framework is web-based with server pages and the database is capable of serving a network, it is available for each of the members of the design team to insert data directly and can be used even for distributed teams.

4. CONCLUSIONS

As described above, the framework introduced here helps to solve several problems of knowledge management which occur in task- and scenario-based software-development processes in which normally design teams are involved. Among these problems design processes often suffer from huge data pools which cannot be used according to the requirements and design questions at hand. The framework and its implementation helps to re-produce design decision during later stages in the development process. The database supports the complete lifecycle of the course unit and makes all information and design rationales available for maintaining and updating the course.

Furthermore, the framework and its implementation is intended to bridge the gap between design knowledge and re-use of design knowledge for the implementation of the hypermedia learning environment. Although the implementation is still on-going, it already seems to be rather promising to produce user- and task-adapted views to the database.

REFERENCES

Carroll, J.M. (Ed.): (1995): Scenario-Based Design. Envisioning Work and Technology in System Development. New York: Wiley

Carroll, J.M. (2000): Making Use. Scenario-Based Design of Human-Computer Interactions. Cambridge, London: The MIT Press

Hartwig, R.; Kritzenberger, H.; Herczeg, M. (2000): Course Production Applying Object Oriented Software Engineering Techniques. In: Proceedings of ED-MEDIA 2000. World Conference on Educational Hypermedia, Multimedia and Telecommunications. 26th June – 1st July 2000. Montréal, Canada. AACE: Association for the Advancement of Computing in Education, pp. 1627-1628

Herczeg, M. (1994): Software-Ergonomie. Addison-Wesley-Longman and Oldenbourg-Verlag

Herczeg, M. (1999): A Task Analysis Framework for Management Systems and Decision Support Systems. In: Proceedings of AoM/IaoM. 17. International Conference on Computer Science. San Diego, California, 6th – 8th August 1999, pp. 29-34

Herczeg, M. (2001, to appear): A Task Analysis and Design Framework for Management Systems and Decision Support Systems. To appear in: International Journal of Intelligent Control Systems

Holtzblatt, K.; Beyer, H. (1993): Making Customer-Centered Design Work for Teams. Communications of the ACM 10, pp. 93-101

ISO 13407: 1999 Human-centered design process for interactive systems

Kritzenberger, H.; Herczeg, M. (2000): Completing Design Concepts for Lifelong Learning. In: Proceedings of ED-MEDIA 2000. International Conference on Educational Multimedia, Hypermedia and Telecommunications. AACE: Association for the Advancement of Computing in Education. Montréal, Canada, 26th June – 1st July, pp. 1374-1375

Kritzenberger, H.; Hartwig, R.; Herczeg, M. (2001, to appear): Scenario-Based Design for Flexible Hypermedia Learning Environments. To appear in: Proceedings of ED-MEDIA 2001. International Conference on Educational Multimedia, Hypermedia and Telecommunications. AACE: Association for the Advancement of Computing in Education. Tampere, Finland, 25th-30th June

Kutsche, O. (2000): Datenbank- und Web-basierte Unterstützung von Entwicklungsprozessen. Studienarbeit, Informatik, Medizinische Universität zu Lübeck, August 2000