

Interaction Design Patterns for Classroom Environments

Henning Breuer¹, Nelson Baloian², Christian Sousa³, and Mitsuji Matsumoto³

¹ University of Applied Sciences Potsdam, Interaction Design Lab,
Pappelallee 8-9, 14469 Potsdam, Germany

² University of Chile, Department of Computer Science,
Blanco Encalada 2120, Santiago de Chile, Chile

³ Waseda-University, Graduate School of Global Information and Telecommunication Studies,
Waseda Bldg. 29-7, 1-3-10 Nishiwaseda, Shinjuku-ku, Tokyo, 169-0051 Japan
breuer@fh-potsdam.de

Abstract. In our research, we synthesize two lines of development that have been dealt with independently so far: 1) the development and evaluation of educational technologies to support problem-oriented and collaborative learning activities inside and outside of the classroom, and 2) interaction design patterns as a means to document and generate design knowledge. Primary contributions are software prototypes for enhancing classroom interaction through interactive whiteboards, multiple clients with pen-tablets and PDAs, and a basic layout of a pattern language for formal and informal learning environments.

Keywords: Interaction Design Patterns, Educational Technology, Learning Theories, Activity Theory, Classroom, Whiteboards, Pen-Tablets, PDAs, Gesture-based Interaction, Open Space.

1 Technologies in the Classroom and Some of Their Problems

In the last 10 years a variety of new, computerized media has moved into the educational settings. Students carry mobile phones, PDAs and laptop computers into classrooms, which are equipped with interactive whiteboards and wireless network connections. During class or at home they retrieve and also contribute information online, handling different interfaces and functionalities. However most of the applications of these media have been developed for other purposes mostly related to business and task-oriented activities. A reasonably consistent learner-centered interaction design across programs and devices may promise to ease the interaction and should flexibly enhance opportunities for learning. Within an ongoing research and development project we therefore try to elaborate upon, implement and evaluate interaction design patterns for formal and informal learning environments using computer technology in the classroom. The first software prototypes we developed address the following problems:

- Oftentimes interaction design principles for personal computing and desktop applications are being transferred to new devices without taking into account their specific properties, potentials and the contexts of their use.
- Interactive whiteboards are usually used in classrooms as presentation media with annotating features. Instead of encouraging active student participation

and contribution, teachers often tend to proceed from one slide to the next of a prefabricated show [4].

- Generating, structuring and documenting learning materials created on the fly during lecture in the class is not sufficiently supported by standard software (like SMART Notebook or PowerPoint).
- There is limited support for group learning activities or for enabling students to contribute to the generation of content on the whiteboard.
- Limited financial resources usually do not allow providing students with “fat” hardware clients like tablet PCs or laptops. Multiple low cost input devices like mice, pen-tablets accessing a single application synchronously on the other hand are not supported by “personal computing” machines.
- Even though PDAs have been discussed as an ideal means to support “learning outside the classroom”, a seamless integration with classroom technologies is still missing.
- Consistency in the interaction design principles and compatibility across devices, applications and environments is missing.

In order to solve these problems we developed various software prototypes (chapter 3). They enhance the interaction with whiteboards, allow students to work in groups and access whiteboard spaces through pen-tablets and PDAs. Since the problems and solutions are recurring for technology enhanced learning scenarios we then abstract from the individual implementations to describe our solution as interaction design patterns (chapter 4). Following this bottom-up approach we finally elaborate upon the emerging system or “language” of interrelated patterns (chapter 5) that should ensure consistency across devices and application scenarios.

2 Related Works

A distinguishing feature of our approach is to support individual and collaborative learning with the help of different devices. In the following paragraphs we give an overview on research on each of those devices and the discourse on patterns in design.

Electronic Board equipped Classrooms: Examples of early usages of electronic boards as an integral part of a computer-integrated classroom can be seen in the Hypercourse [16], in Taiwan secondary schools [24] and in the COSOFT project [2]. The computer-integrated classroom combines positive aspects of the classical chalkboard approach, particularly its flexibility in the spontaneous elaboration of ideas, with the potential of modern networked multimedia. The value added lies on the avoidance of discontinuities in representations (“media breaks”), e.g., when the solution that a student has individually elaborated on is copied again by hand to the chalkboard. Another example can be found in the European NIMIS project [10]. Target users for this version were young schoolchildren (4-8 years old) and it was aimed to develop children’s reading/writing skills.

Multiple pen-tablet input: Collaborative learning in computer enhanced environments needs to support collaborative content creation and presentation. LiveNotes connects tablet PCs by a wireless network [13]. However, a different challenge comes up when

multiple input devices access a single PC. The SDG Toolkit provides an independent layer for the development of applications with multiple keyboards or mice input. It was used in a learning application for developing countries where computers are scarce [18]. Another toolkit called TIDL [11] supports multiple mice and keyboards for legacy and custom applications.

Mobile Computing and Learning: Mobile devices are already been recognized as a powerful tools for supporting formal learning in the classroom situations and informal learning outside the classroom. Handhelds have been described as 'flexible tools that can be adapted to suit the needs of a variety of teaching and learning styles' [7]. In [19] authors provide a review of existing work in a number of areas which suggest possibilities for the future of research on mobile learning in informal science settings. However, like Scanlon et al, we also believe that it is necessary to take an integrative perspective on orchestrating technology enhanced educational scenarios. Our work is directly aimed at proposing such an environment.

Interaction Design Patterns: Interaction design patterns are standard solutions to recurring problems in interaction design. Originating from architecture theory they were adapted first to software development [9] and consequently to interaction design [3]. Pedagogical patterns have been proposed in order "to capture expert knowledge of the practice of teaching and learning in a portable, salient format" [20]. Usually they have been written and discussed independent of the technological development. Even a pattern language for web-based learning has been proposed [14]. Within HCI most work on design patterns has focused on web design and mobile devices. Using different notations and hierarchical levels pattern language have been proposed not only for Website design and shopping websites [23], but also for exhibitions [3]. Companies like Yahoo! and SAP have reported their experiences with introducing a design pattern approach into the organization.

3 Implementation

We developed a platform to support the programming of distributed applications running on different platforms. Due to the current stand of the technology, C# is the best alternative for programming applications for mobile devices and Java for desktop applications. Since we wanted to combine both worlds we developed a platform based on shared-object principle. Applications running on different platforms sharing an object will have automatically the status of the object synchronized. The architecture of this platform is decentralized, replicated and the information transfer is based in XML. It connects the DeepBoard Interface for interactive whiteboards with synchronous input through pen-tablets and PDAs. The following paragraphs describe the implementation status for each of these devices.

Interactive whiteboard application: Our DeepBoard application [4] implements a gesture-based interaction paradigm and a hierarchical semantic to store and retrieve learning created on the flight by free-hand writing. Drawing or performing gestures on the board may result in a flexible creation, structuring, presentation and documentation of various learning materials. The DeepBoard therefore supplies "depth" to the interaction with interactive whiteboards.

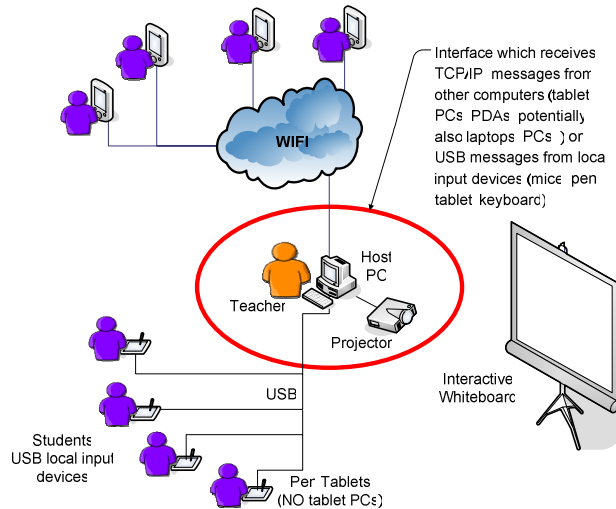


Fig. 1. General architecture of the platform: Mobile users use WiFi network to communicate among them and with the server controlling the interactive electronic whiteboard. Tablet users use USB connection directly to the server. Teacher uses e-board as input device.

Access for multiple pen-tablets: In order to enable several pen-tablets to interact in the same space within an application program in one PC, the DeepBoard application was modified. An intermediate software layer is responsible for receiving and interpreting the packets generated by the pen-tablets. These packets are analyzed and then are encapsulated to generate equivalent mouse events. Finally these events are sent to the modified application which can process them simultaneously.

Collaborative Learning with PDAs: MCSketcher is a system that enables face-to-face collaborative design based on sketches using handheld devices equipped for spontaneous wireless peer-to-peer networking. It may be used by students inside or outside the classroom to exchange ideas through sketches on empty sheets or over a recently taken photograph of the object being worked on, in a brainstorming-like working style. Pen-based designed human-computer interaction is the key to supporting collaborative work. This application was entirely written in C# aimed for being used mainly on mobile devices.

Nomad is an Electronic Meeting Support system for handhelds. The design principles applied for developing the system are aimed to help reduce the problems associated with having a small size screen to interact with. The human-handheld interaction is based on gestures and freehand writing, avoiding the need of widgets and virtual keyboards. The content of the generated documents are organized as concept maps, which gives more flexibility to reorganize and merge the contributions of the meeting attendees. The system is based on handhelds interconnected with an ad-hoc wireless network. This application has a module which allows the use of an electronic board in order to have a common display to show the content being produced during the working session.

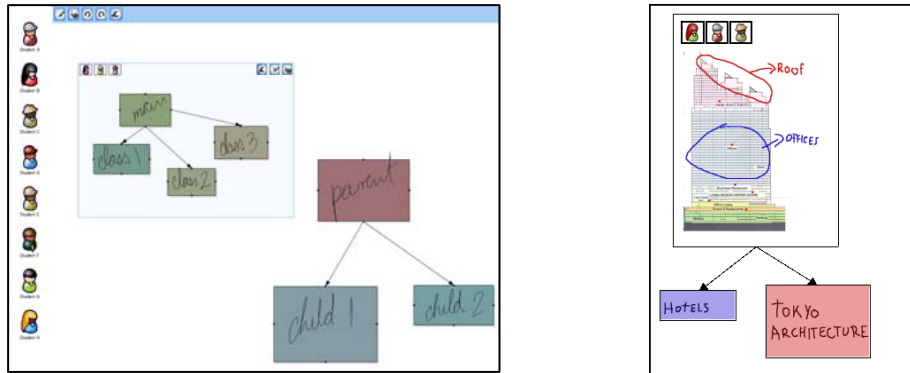


Fig. 2. The left image presents a screenshot of the whiteboard after the teacher and students have created some rectangular node objects for creating or importing content, and connected them by drawing from one node to the other. The right picture shows a sketch of the PDA interface designed according to the same interaction design patterns.

4 Interaction Design Patterns for the Classroom

In order to communicate our approach and to define and refine a reference for various kinds of formal and informal evaluation we documented our solutions in the form of Interaction Design Patterns (IDP). We propose a “generative pattern approach” combining the critical and the generative potentials of patterns: to document and optimize existing solutions and patterns, but also to critically reflect upon them in order to generate new design knowledge. In order to illustrate this approach we will present some of the patterns we abstracted from our design experience. In the context of this paper we focus on those middle-level patterns for design solutions that support student activities on different devices. Describing them we also want to show how individual patterns may be related forming an interaction design pattern language for learning environments.

While no officially standardized notation for interaction design patterns exists, each expresses a relation between a certain context, a problem, and a solution, as already proposed in [1]. We argue for a notation consisting of a name, a context and superior patterns, a problem statement (including the conflicting forces), a solution, an example with (ideally empirical) evidence and subordinate or (by the logical relations of AND, OR, or NOT) related patterns [5]. Optionally additional information may address the level of confidence in the respective patterns (referring to empirical evidence), the devices a pattern applies to, or further references on the subject field. The following sections provide examples of our current design patterns.

4.2 Name: Open (White) Space

- Context: Constructivist (or problem-oriented) didactics
- Problem: Problem-oriented or constructivist learning should emphasize students’ self-directed activities and start with students’ construction and discussion. Many

educational technologies in the classroom support a teacher centered approach by supporting instruction more than construction. Especially the presentation and annotation of prefabricated materials may not encourage student participation.

- **Solution:** Provide one or many open spaces for students to create and contribute content, and to put pieces into relation. (Use bright colors for these spaces to ensure contrast.)
- **Example and evidence:** Besides the prototype we presented the so called “Open Space Technology” [17] is moderation methods addressing diversity within an organization, a community, or even a classroom. Initially instructed by a facilitator who presents the general theme, the process and its guidelines, the participants move around, interact and contribute freely to designated areas driven by their interests. They engage in self-initiated or ongoing discussions as long as they may contribute or learn something, or disengage and go someplace else. Using various materials they take notes and document their self-organized activities.
- **Related patterns:** gesture-based interaction, multiple input (allows e.g. for multiple clients to access the whiteboard and contribute content), assign participants, contribute content, collect information, shared screen mode (PDA)s)

4.3 Name: Gesture-Based Interaction

- **Context:** Constructivist (or problem-oriented) in-class learning activities, learning with touch sensitive devices
- **Problem:** On small screen devices menus and widgets consume precious working space and are wearisome to hit. Other than desktop application no right mouse click is available to provide context dependant popup-menus. In order to support the focus of attention it would be desirable to keep user input and system response in the same space. On large screen devices spending too much time in activities, which are not directly related to teaching (typing long commands or queries, or searching for files) may interrupt the dynamic flow of the lecture and distract the attention of the audience.
- **Solution:** Provide for a gesture-based interaction that enables users create, import, edit, relate and delete pieces of content. With interactive whiteboards allow to perform all activities related to teaching on the whiteboard as the unique input and output device.
- **Examples** are the prototypes for pen-tablets, PDAs and the DeepBoard
- **Subordinate patterns** include the individual gestures, such as creating, deleting and linking nodes (pattern for semantic interpretation), also change mode (writing versus gesturing) and mode indication, related patterns include, group presence widget (represents the student group and their position in the context of learning activities), remote access, assign participants (to groups or granting access to open white spaces on the whiteboard).

4.4 Name: Change Mode

- **Context:** Gesture-based interaction, touch sensitive devices
- **Problem:** Gesture-based interaction provides a valuable alternative to selection from menus or widgets. Because the gesturing space is collocated with the space

for drawing (or content creation), user input and system recognition must differ between the two. The more gestures a system supports the more difficult it becomes to differ gestures from drawing within a single interaction mode. From a learner's perspective it makes sense to differentiate between working within some piece of content and applying a meta-perspective on different pieces of content.

- Solution: Provide for two distinct modes for gesturing and drawing. Allow for easy switches between the two modes (on small devices preferably using hard keys) and indicate which mode the user is working on (include pattern for mode indication).
- Examples are again provided by the prototypes we described, the icon on the DeepBoard interface and a hard key and the state indication of the PDA.
- Related patterns: mode indication, gesture, select, write and draw.

5 Towards a Pattern Language for Educational Technologies

The usually hierarchical network of these relations between patterns creates a so called pattern language. The pattern language for supporting classroom activities we intend to develop does not yet exist. Looking at learning environments from a psychological point of view we have to differentiate between the different codes, modalities and technical media being involved and view all of them in relation to the instructional methods they need to support [22]. While the instructional methods are ways to facilitate learning processes of the students, instructional media have been described as surface phenomena that solely deliver the methods [6]. While there has been some controversy whether the instructional methods and structuring of learning content should be considered primary to media attributes, Jonassen [12] proposed to move from an instructional- and media-centered towards a learner-centered design instead. The latter approach focuses more on supporting than controlling learning processes. The learner-centered approach suggests starting from student needs and activities in order to define system properties and design patterns for educational technologies.

With respect to instructional methods and their overarching influence on learning processes we expect patterns supporting them to be on higher levels than those addressing the properties of the media themselves. Trying to develop a framework for an emerging design pattern language for learning environments, we currently differentiate between five rough levels of potential learner support. For this differentiation of the three basic levels we draw from works on activity theory [15], which emphasizes the cultural and technical mediation of human activity under a developmental perspective. Activity theory differentiates routine operations, goal-oriented, conscious actions and meaningful activities and understands learning activity as expansion in various forms [8], e.g. from a problem to a context to defining a new, advanced activity. Considering the potentials of defining patterns on levels of increasing generality, as well as the institutional embedding of many learning activities the categories we propose include an additional level for curricula and long-time learning goals, and one for organizational and didactical contexts, educational technology may hinder or support.

- Organization and didactics: Basic theories and the practice of teaching and learning influence overarching design paradigms like goal oriented consistency across platforms and devices, methods to provide support for different learner types, also organizational support impacting all levels of interaction.
- Curricula and learning goals (or objectives) refer to time-bound bundles of courses and their contents. They include the instructional methods and are supposed to enable the achievement of learning goals that orient activities. On this level we propose to locate complete learning management or collaboration systems.
- Activities in our understanding are derived from a subject or unit in the curriculum, or may be derived from the students' interests. Interaction design patterns on this level may provide a context for these activities and include group features like a shared screen mode for multi-device environments, multiple input, a group awareness widget, but also educational games or learning environments.
- Tasks are necessary steps to fulfill an activity. Most individual and oftentimes device-specific procedures and applications supporting learning activities belong to this category. Examples may include modeling tools and simulators, achievement levels in games, specific training programs, or collaborative drawing.
- Operations: address the implicit reuse of previously gained knowledge. Interaction Design Patterns supporting operations include the gestures, also uses of different input and output modalities.

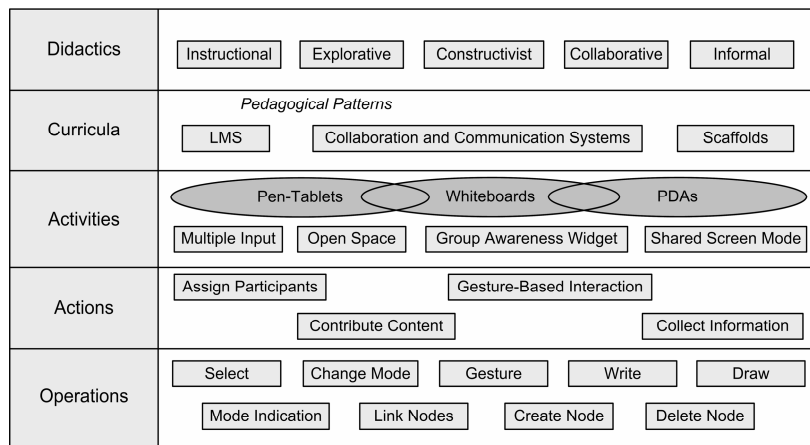


Fig. 3. Preliminary outline of an interaction design pattern language for formal and informal learning environments. Individual patterns are shown in rectangles; devices that are supported in our current implementation are shown in ovals.

Trying to match these hierarchical levels of didactics and student activities to technological properties we preliminary associate didactics to (technological) paradigms, curricula to systems, activities to environments, actions to applications and operations to functionalities. The following figure gives a preliminary overview of the emerging pattern language that needs to be further developed.

6 Conclusion

We developed various software prototypes to support collaborative learning in the classroom with interactive technologies. Abstracting from these implementations we described our solutions as patterns that may be applied (e.g. the pattern for Open Space) independent of or (e.g. the Shared Screen Mode for PDAs) tailored to specific devices. Reflecting the contexts and conflicting forces of the individual patterns already helped us with finding and refining design decisions. The formation and continuous adaptation of the emerging interaction design pattern language shifts the focus enabling us to see individual functionalities and systems in the context of overarching design principles matching learner-centered requirements. Still many descriptions of patterns and links between them are missing. An empirical evaluation of this language and its patterns is an ambitious project since many implementations must be evaluated to validate the appropriateness of even a single pattern. In this perspective a formative evaluation of the pattern language as a guideline and reference for design and empirical evaluation of systems and environments developed accordingly are a viable option.

Our long-term perspective is to contribute to the development of a patterns language for formal and informal learning environments that seamlessly integrates architecture, interaction and software design.

Acknowledgments. This work was partially funded by a research grant of NICT, the National Institute of Information and Communication Technology of Japan.

References

1. Alexander, C., Ishikawa, S., Silverstein, M.: *A Pattern Language*. Towns, Buildings, Construction. Oxford University Press, New York (1977)
2. Baloian, N., Pino, J.A., Hoppe, H.U.: *A Teaching/Learning Approach to CSCL*. In: 33rd Hawaii International Conference on System Sciences, Maui, HI, USA, pp. 1–10. IEEE Computer Society Press, Los Alamitos, CA (2000)
3. Borchers, J.: *A pattern approach to interaction design*. John Wiley & Sons, Chichester (2001)
4. Breuer, H. & Baloian, N.: *Augmenting Whiteboard Interaction in the Classroom*. In: Kommers, P., Richards, G.(eds.): *Proceedings of ED-Media 2005 Conference*. Montreal, Canada, pp. 1214–1221 (2005)
5. Breuer, H., Matsumoto, M.: *Interaction Design Patterns for Learning Environments – Basic Concepts and a Prototype*. IEICE Technical Report, vol. 106, no. 496, pp. 7–12 MVE2006-72 (2007)
6. Clark, R.E.: *Media will never influence learning*. *Educational Technology Research and Development* 42(2), 21–29 (1994)
7. Curtis, M., Luchini, K., Bobrowsky, B., Quintana, C. and Soloway, E.: *Handheld Use in K-12: a descriptive account*, In: *Proceedings of the IEEE International Workshop on Wireless and Mobile Technologies in Education*, IEEE Computer Society, Los Alamitos, CA (2002)
8. Engeström, Y.: *Learning by Expanding: An Activity-Theoretical Approach to Developmental Research*. [Internet]. Available from: <http://communication.ucsd.edu/MCA/Paper/Engestrom/expanding/toc.htm> (1987/Feb. 2007)

9. Gamma, E.H.R., Johnson, R., Vlissides, J.: *Design Patterns. Elements of Reusable Object-Oriented Software*. Addison Wesley: Reading, Mass (1995)
10. Hoppe, H.U., Lingnau, A., Machado, I., Paiva, A., Prada, R. & Tewissen, F.: Supporting collaborative activities in computer-integrated classrooms - the NIMIS approach. In: *Proceedings of CRIWG 2000 International Workshop on Groupware* pp. 94–101 (2000)
11. Hutterer, P., Close, B., Thomas, B.: TIDL: mixed presence groupware support for legacy and custom applications. In: *ACM International Conf. In: Proc. Series vol. 169. In: Proceedings of the 7th Australasian User Interface Conf. (Internet) (2006)* <http://wearables.unisa.edu.au/mpx>
12. Jonassen, D.H., Campbell, J.P., Davidson, M.E.: Learning with media: Restructuring the debate. In: *Educational Technology Research and Development* 2(42), 31–40 (1994)
13. Kam, M., Wang, J., Iles, A., Tse, E., Chiu, J., Glaser, D., Tarshish, O., Canny, J.: Livenotes: a system for cooperative and augmented note-taking in lectures. In: *CHI 2005. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Portland, Oregon, USA, pp. 531–540. ACM Press, New York (2005)*
14. Leeds Metropolitan University: *Web-based Learning Pattern language [Internet]*. (February 2007) Available from <http://www.lmu.ac.uk/ies/comp/research/isle/patterns/ByCat.htm>
15. Nardi, B.A. (ed.): *Context and consciousness: Activity theory and human-computer interaction*. MIT Press, Cambridge, Mass (1996)
16. Norman, K.: Hypercourseware for assisting teachers in the interactive electronic classroom. In: Willis, J., Robin, B., Willis, D. A.(eds.): *Procs. of Fifth Annual Conference of the Society for Technology and Teacher Education, Washington, D.C., USA, pp. 473–477 (1994)*
17. Owen, H.: *Open Space Technology: A User's Guide*, 2nd edn. Berrett Koehler Publishers, San Francisco (1997)
18. Pawar, U., Pal, J., Toyama, K.: Multiple Mice for Computers in Education in Developing Countries. In: *1st International Conference on ICT and Development, UC Berkeley (2006)*
19. Scanlon, E., Waycott, A.J.: Mobile technologies: prospects for their use in learning in informal science settings. In: *Journal of Interactive Media in Education (Portable Learning: Experiences with Mobile Devices. Special Issue 2005/25 (2005)*
20. The Pedagogical Pattern Project [Internet] (February 2007) <http://www.pedagogicalpatterns.org/>
21. Tidwell, J.: *Common Ground: A Pattern Language for Human-Computer Interface Design [Internet] (1999)* Available from: <http://www.mit.edu/~tidwell>
22. Weidenmann, B.: Multicodierung und Multimodalität im Lernprozess (Multicoding and multimodality in the learning process. In: Issing, L.I., Klimsa, P. (eds.) *Information und Lernen mit Multimedia*, 2nd edn, pp. 65–83. Psychologie Verlags Union, Weinheim (1997)
23. van Welie, M., van der Veer, G.C.: *Pattern Languages in Interaction Design: Structure and Organization*. In: *Proceedings of Interact 2003. International Human Computer Conference, Zuerich, Switzerland (2003)*
24. Welie, M. v.: *Patterns in Interaction Design: Web Site Designs [Internet]*. (February 16, 2003) Available from <http://www.welie.com/patterns/index.html>
25. Wu, C.-C., Lee, G.C.: ICT Integration at a Taiwan Secondary School In *Proceedings of the Computers and Advanced Technology in Education (CATE), Mexico pp. 131–151, (2002)*