Konferencija ELEKTRONSKO UČENJE NA PUTU KA DRUŠTVU ZNANJA 2010
Univerzitet METROPOLITAN

ELEKTRONSKO UČENJE NA DEPARTMANU ZA MATEMATIKU I INFORMATIKU – OD PRVIH KORAKA DO ELEMENATA WEB 2.0

E-LEARNING AT THE DEPARTMENT OF MATHEMATICS AND INFORMATICS – FROM THE FIRST STEPS TO ELEMENTS OF WEB 2.0

ZORAN BUDIMAC, MIRJANA IVANOVIĆ, ZORAN PUTNIK, ŽIVANA KOMLENOV, IVAN PRIBELA
Prirodno-matematički fakultet, Departman za matematiku i informatiku, Novi Sad,
{zjb, mira, putnik, komlenov, pribela}@dmi.uns.ac.rs

ALEKSANDRA KLAŠNJA-MILIĆEVIĆ, BOBAN VESIN
Higher School of Professional Business Studies, Novi Sad,
{aklasnja, bobanvesin}@yahoo.com

Rezime: *Ovaj rad prikazuje istorijski razvoj i trenutno stanje primene elektronskog učenja na Departmanu za matematiku, Prirodno-matematičkog fakulteta i Visoke poslovne škole u Novom Sadu. Elektronski kursevi, međunarodna saradnja u okviru više projekata, priručnici, te praktično korišćenje elektronskog učenja u svakodnevnoj nastavi informatike, zasluga su grupe profesora i asistenata okupljene u okviru i oko Katedre za računarske nauke Departmana. Ciljevi rada se ostvaruju korišćenjem otvorenog sistema za podršku elektronskom učenju Moodle, kao i upotrebom samostalno razvijenih alata ili dodataka i proširenja korišćenog LMS-a.*

Ključne reči: *Elektronsko učenje, Moodle*

Abstract: *This paper presents history of the development, and current application state at the Department of Mathematics and Informatics, Faculty of Science and Higher School of Professional Business Studies in Novi Sad. Electronic courses, international cooperation through several projects, booklets and manuals, and practical usage of eLearning in teaching of informatics can all be credited to a group of professors and assistants assembled in and around Chair of Computer Science at the Department. These goals are obtained through an open learning management support system Moodle, but also through self-developed tools and extensions of a used LMS.*

Keywords: *eLearning, Moodle*

1. INTRODUCTION

Ideas about the practical usage of eLearning at the Department of Mathematics and Informatics started during the year 2003. At the time, some of the members of the Computer Science Chair begin exploration of existing learning management systems, their practical capabilities, and possibilities for their inclusion in everyday teaching. Several learning management systems were investigated, same as scientific papers dealing with comparison of those, and LMS Moodle was chosen for testing. As an open source system, already translated to Serbian language, well supported by hundreds of developers, and tested by thousands of users, it seemed as an appropriate tool for start.

System was installed at the local server of the Department and some electronic material has been developed at first for the course in "Software Engineering". This course is developed as a part of internationally supported project with 9 countries participating [1], so it had strong both research and educational background, and seemed to be an appropriate course for starting the system [2].

After a successful start, several other undergraduate courses were added, like courses in “Operating systems”, “Ethical Aspects of Informatics” [3], “Introduction to e-Business”, “Object-oriented Programming”, or “Data Structures and Algorithms”.

Having a good experience with the Moodle system usage, and positive feedback from students, under three small projects supported by WUS-Austria [4], we developed electronic material for courses on “Software Project Management”, and "Methodic of Informatics I and II", but also produced user-manuals both for teachers and students, users of LMS Moodle.

At master studies,, as a part of another big international TEMPUS project [5], we have created complete eLearning environment for a whole set of around twenty master courses, such as: “Privacy, ethics, and Social Responsibilities”, “Software Engineering for Critical Systems”, “Formal Method Engineering”, "Software Testing" or “Component Based Development”.

Finally, as a part of bilateral project [6] conducted together with the Faculty of Electronics, Computing and

Informatics from Maribor, Slovenia, eLearning system has been further developed. Also, a large survey has been conducted about the students' satisfaction with the LMS Moodle, both in Serbia and Slovenia [7].

Along the way, several extensions and developments of LMS Moodle have been created, same as standalone tools for prevention of cheating, teaching material development and delivery, and similar. In the following sections, those will be presented in some details. Second section presents Moodle system structure as it is used at our Department, while the third section explains in more details changes and additions performed within Moodle. Section four deals with standalone applications developed at the Department and used in everyday teaching. Sections five and six present work performed by our colleagues from Higher School of Professional Business Studies, who are also part of research group of our Department. Finally, section seven gives some conclusions and overview of current state and possible future work.

2. SYSTEM STRUCTURE

Our eLearning system consists of several parts:

- eContent (repository of teaching materials),
- quizzes and testing facilities,
- live chats between students and teachers,
- specialized forums for students to exchange ideas, additional sources of information and ways to solve assignments,
- statistics on earned points and progress reports for teachers and students,
- eLessons (specialized electronic lessons for independent study and checking the progress)
- facilities to submit solutions to assignments with prevention of cheating,
- automatic checks of certain aspects of solutions to assignments (when solutions are computer programs).

Besides, it is worth mentioning that our eLearning system recognizes two different forms of eContent:

- most of the material is given in a form of "electronic lessons", created by students as a part of elective course "eLearning", and used as self-learning material.
- all of the content is given as a repository of teaching material, usually as PDF versions of presentations given to students.

Most of the courses have testing facilities in a twofold form. First, each eLesson is from time to time interrupted with questions, for keeping students "awake" and concentrated. Additionally, final quizzes are created within each section of eCourses. We decided to use these quizzes only as unofficial, self-testing facility for students. Besides this, we use Moodle's quiz module for creating repositories of questions and generating official tests.

Live chats between students and teachers, and forums for exchange of ideas and additional material, and for discussions about problems and solutions of assignments are in constant use.

All enrolled students, and only the enrolled students, had information about points gained at each of the tests/assignments.

Students of "Software Engineering" studied 2 electronic lessons by themselves, and were satisfied with those. In order to test self-learning, we conducted a test for those self-studied lessons, with very good results.

Finally, a self-developed application called "Svetovid" was created in order to prevent cheating and help in submission of solved assignments. It is used in several courses: "Object-oriented Programming", "Data Structures and Algorithms", "Operating Systems", "Compiler Construction"… for example, and it worked very well. The quality of original solutions received from students increased dramatically. Svetovid also has a possibility to help in solution submission, in a sense that it automatically checks certain simpler aspects of students' solutions – syntax, technical aspects, or easy-to-notice logical errors. Considering the relatively large number of students, this application proved to be of a great help to assistants in the process of checking assignments.

3. CUSTOMIZATIONS OF MOODLE

Our web-based eLearning course support software employs Open Source Course Management System Moodle [8] with customizations made by our project members according to our needs [9].

eCourses developed at the Department required more adaptivity concerning navigation through eLessons. For example: the course creators wanted more flexibility in their students' eLearning experience, in a way that they can explicitly choose different paths through lessons or can be directed to different parts depending on their answers. For that purpose, several navigational extensions of Moodle have been developed and most of them have already been put into practice.

Firstly, direct referencing (jumps) from eContents to other parts of eContents is a useful ability to reference parts of other eLessons within the course from any point of the current eLesson. There are at least two good purposes for such an extension:

1. Redirection of learners to other parts of eContents, for instance after giving the wrong answer to a question concerning already read material in the current lesson, or facts from previous lessons, it could be useful to go back to the explanations of that particular topic and go through, for example several pages dealing with the problematic issue.
2. Simple reminders - apart from complete redirection of learners to another page, there is a possibility to just remind them of certain pages, without disturbing their learning path.

These features have been fully integrated into existing structure of HTML editor used while creating eContents.

Secondly, extended jumps represent a possibility to move from one page in eLesson to the other, even from other eLessons providing direct reuse of the existing material without duplicating it, depending on student's current activity.

Thirdly, conditional jumps allow tracing where users came from when entering the current page and directing them further through the eContents in the most appropriate way, therefore designing more meaningful eContents, leading to greater success with students.

4. SVETOVID SUBMISSION SYSTEM

Submission system named "Svetovid" is a cross-platform software that helps instructors leverage the effort of practical exercises and exams. The system was created with the idea of close integration with the rest of our systems, but along the way we concluded that it is better to keep it as a separate facility.

The goals for the Svetovid submission environment were as follows:

1. Allow students to comfortably develop their programs.
2. Allow students to test their programs before submission.
3. Keep a log of student efforts.
4. Be flexible enough and usable for different courses including wide range of programming languages and project stage: coding, typing, program documentation...
5. Disallow students to share programs and solutions, intentionally or unintentionally.
6. Help instructors with marking student solution (i.e. program code).

The first goal was achieved by designing and implementing easy-to-use client program for program development, by incorporating most common IDE ideas. Some sample input and output data, prepared by instructor, for preliminary program test, are available in advance. Students can test their programs by running them against one data set on which they will eventually be tested, and under precisely the same conditions.

This ensures that the program will work the way student expects and that it satisfies some basic requirements, so the second goal was achieved.

The third goal was achieved by integrating an extensive logging mechanism to the system. Separate log files are kept for each day, and each course so that the lecturer can monitor student progress, or find potentially suspicious students.

The fourth goal is achieved by means of command scripts: by placing all necessary tasks, like compile, run, test, etc., in batch scripts provided by the instructor. By writing an appropriate command sequence in the batch script, the instructor can accommodate any programming language. We are using the system to automate compiling and running of Modula 2, Java, and Scheme programs; for courses in "Operating systems", "Object-oriented programming", "Programming languages", "Computer graphics", "Compiler Construction" and "Data Structures and Algorithms". Moreover, commands can test any aspect of files submitted by student. For example: programming style, line indentation, commenting, frequency of words, existence (or inexistence) of certain key words. With provided generality even assignments that are not programs can be tested. The system can be used for assessing quiz answers, checking program documentation, etc.

The fifth goal is achieved by means of password protected virtual directories placed on another (or the same) computer, and accessible to students only from the Svetovid client, but not via the standard file system.

The final goal is gained by the same command scripts for compilations and testing available to students. While command scripts provide automation of evaluation process, the concept of virtual directories provides security for instructor's data.

5. PROTUS – PROGRAMMING TUTORING SYSTEM

Protus is a tutoring system designed to help learners in learning basics of programming languages [10]. In spite of fact that this system is designed and implemented as a general tutoring system for different programming languages, the first completely implemented and tested version of the system was for introductory Java programming course. Java is chosen because it is a clear example of an object-oriented language and is therefore suitable for the teaching of the concepts of object-orientation. The environment is designed for learning programming basics for learners with no object-oriented programming experience. It is an interactive system that allows learners to use teaching material prepared within appropriate course and also includes part for testing acquired knowledge.

Two main roles exist in the system, intended for two types of system's users [11]:

- learners - they are taking the Java programming course and will be using the system in order to gain certain knowledge and
- teachers and content authors - the lesson and learner database administrator; they track learning process of learners and help them with their assignments.

Therefore, separated user interfaces are provided for learner and teacher. Teacher's interface helps in process of managing data about a learner and course material. Learner's interface is a series of web pages that provide two options: taking lessons and testing learner's knowledge. All data about learner and his progress in the course, as well as data about tutorials, tests and examples are stored on the system's server.

6. PERSONALIZATION IN PROTUS

Protus provides two general categories of personalization in system based on adaptive hypermedia and recommender systems.

Adaptive hypermedia techniques make it possible to inform learners that certain link lead to material they are not ready for, to suggest visiting pages the learner should consult, or automatically provide additional explanations at the pages the learner visits, in order to scaffold his/her progress [12]. Protus tracks every learner's path through the course text and thus system is provided with information whether for the current learner at this time a reference is a forward or backward reference. This allows authors to create multiple versions of the references and system can choose and present the appropriate one.

Recommender systems act as decision guides for users, aiding them in decision making about matters related to personal taste [13]. The recommendation module of Protus system is designed to [10]:

- recognize different patterns of learning style and learning habits through testing the learning styles of learners,
- form the clusters of learners based on learning style and then discover behavioral patterns for each learner by AprioriAll algorithm and
- carry out recommendation list according to the ratings of these frequent sequences, provided by the Protus system.

In the future, we plan to combine these different techniques with tag-based recommender systems, where data from learners' behavior and annotations/tags will be used on the fly for modifying and updating the learner model. Tag-based recommender systems allow learners to tag the resources they've studied by labeling them with specific labels – tags. The identification of tag collection may lead to a better recognition of learner's interests and knowledge level in different topics.

7. RESUME

ELearning is used at the Department of Mathematics and Informatics with undergraduate and graduate courses. Besides, the system is successfully used in international environment to exchange ideas and knowledge, especially via live chats and student forums.

The Svetovid system is used in undergraduate course in Operating Systems, Compiler Construction, Object Oriented Programming, and many other, where prevention of cheating and automatic checks is essential. Protus system was evaluated with experiments that were carried on an educational dataset. Students of Higher School of Professional Business Studies, Novi Sad University tested the system and provided usefull feedback.

Two manuals have been created for easier usage of Moodle CMS. The first one is designed to be used by students, and concerns the usage of the system from the student perspective. The other one is designed to be used by lecturers, so it covers some other points of interest of a person who wants to create a eLessons, or even whole eCourses, even if that person is not a professional in computer science.

A complete new translation of Moodle interface, in both Latin and Cyrillic version has been performed in order to improve the previous one. This translation was accepted by Moodle consortium as official Serbian language pack(s).

LITERATURE

[1] Joint Software Engineering course homepage. http:// www2.informatik.hu-berlin.de/swt/intkoop/jcse/

[2] Z.Budimac, Z.Putnik, M.Ivanović, K.Bothe, K.Schuetzler: *On the assessment and self-assessment in a students teamwork based course on software engineering*, Computer Applications in Engineering Education, early view, DOI 10.1002/cae.20249, 2009

[3] Putnik, Z., Ivanović, M., Budimac, Z.: "Extending The Notion of Computing Literacy ", 4th International Conference "Informatics, Educational Technology and New Media in Education", Sombor, Serbia, pp. 167 - 179, 2007.

[4] WUS Austria realized projects homepage, http:// www.wus-austria.org/rprojects/44/36.html

[5] TEMPUS project homepage for Joint M.Sc. Curriculum in Software Engineering http:// perun.dmi.rs/msc-se/

[6] Bilateral project "eLearning support system with application of data-mining and data security", 2008 – 2009, Joint project of Dept. of Mathematics and Informatics (Novi Sad) and Faculty of Electronics, Computing and Informatics (Maribor, Slovenia)

[7] Ivanović, M., Welzer, T., Putnik, Z., Hoelbl, M., Komlenov, Ž., Pribela, I., Schweighofer, T.: "Experiences and privacy issues - usage of Moodle in Serbia and Slovenia", Interactive Computer Aided Learning, ICL2009, Villach, Austria, pp. 416-423, 2009.

[8] LMS Moodle homepage. http://moodle.org

[9] Komlenov, Ž., Budimac, Z.: "On Creating Adaptive eLessons with Improved Navigational Features", XVII Conference on Applied Mathematics "PRIM 2006", 2006, Serbia

[10] Vesin, B., Ivanovic, M., Budimac, Z., & Pribela, I. . Mile-Multifunctional Integrated Learning Environment. In e-Learning, pp. 104-108, 2008.

[11] Klašnja-Milićević, A., Vesin, B., Ivanović, M., & Budimac, Z. Integration of recommendations into Java tutoring system. In The 4th International Conference on Information Technology ICIT 2009 Jordan.

[12] Devedžić, V.: *Semantic web and education* (1st ed.), Springer Science, New York. 2006.

Adomavicius, G., Tuzhilin, A.: Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6), pp. 734-749, 2005.