FIRST STEPS TOWARDS A CONCEPTUAL MODEL FOR A VIRTUAL UNIVERSITY

Cristina Fierbinteanu * **Mihai Buneci** **

* *Titu Maiorescu University*
Calea Vacaresti 187, sect.4, Bucharest, Romania
** *National Institute for Economic Research*
Calea 13 Septembrie 13, Bucharest, Romania

Abstract: In this paper we present the first results of our work aiming the implementation of a virtual university campus. Two years ago we began using Moodle, a package of software tools for producing Internet-based courses, in an attempt to offer support to distance learners. Soon we realized that we needed a different concept of student group than that supported by the system. The definition of this concept was the starting point of a conceptual model for a virtual university, which we are currently developing.

Keywords: e-learning, distance learning, virtual learning environment, intelligent tutoring systems, content management systems

1. INTRODUCTION

The European e-learning strategy includes the requirement that all universities should offer online access for the members of their communities (Patriciu and Podaru (2005)). E-learning (i.e. the use of multimedia and internet technologies to improve the quality of distance learning, exchanges and collaboration), while opening unprecedented possibilities, also requires new skills and unveils new challenges (Brut (2006)). Probably the most popular open-source content management system, Moodle is a package of software tools which can be used by professors to produce Internet-based courses. The system is helpful for distance learning as well as for enhancing classical, face-to-face teaching. Several types of courses (weekly, subject oriented or discussion oriented) and several types of activities within a course (forums, journals, tests, assignments, quizzes, chats, workshops) are available. The users of the system are divided into three categories: administrators, tutors (who can be course editors or just teachers) and students. The functionality of the system, as well as its graphical interface, differ according to the user's category and rights. Moodle supports a constructivist learning system, which means, among others, the promotion of learning in cooperation. With this in view, the system provides means to define groups of students within course units (see moodle.org). However we encountered the need to define global groups of students, that is groups transcending course units. In this paper we present the implementation of global groups using MySQL and Java.

2. MOTIVATION

SCORM (Sharable Content Object Reference Model) is the most popular standardization initiative in the field of instructional technologies (Serb (2005), www.telscenter.org) but it doesn't support tracking and sequencing of student group's actions, as well as individual students' actions in the context of group work. We have approached the design of the concepts and tools necessary to support learning in cooperation.

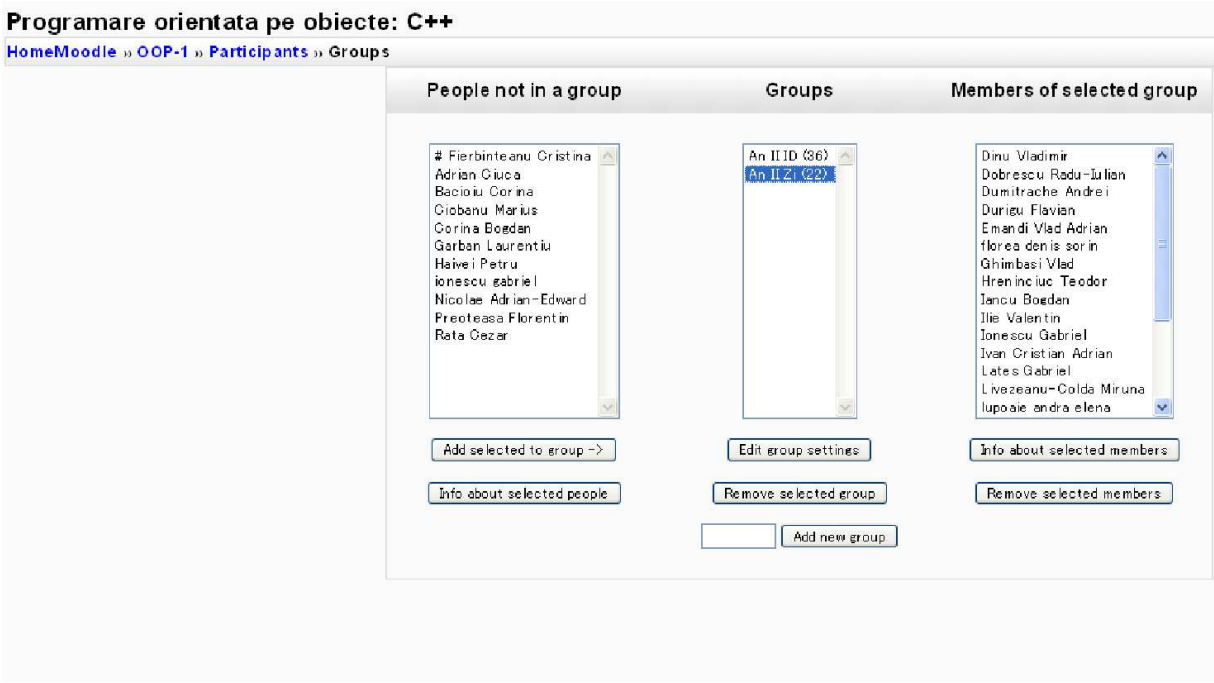

Fig. 1. Group definition in Moodle

Two years ago we began using Moodle, in an attempt to offer support to distance learners (Fierbinteanu (2005)). Soon we realized that the concept of "group" implemented in the system did not support the definition of student groups, as they are registered by the university secretariate. Moodle only supports the definition of student teams within a course. Moreover, no function is provided for the re-usability of one such team, that is if one tutor wishes to define in one of her courses a team identical to another team from another course, she would have to add the students in the new group one by one. It would have been desirable to be able to import a group of students into one's course, once that group had been already defined by someone in another course. We started the design of a tool for system administration named Maud (standing for "Moodle Administrator's Utilitarian Device") implementing the missing functionality in Moodle, and at the same time the conception of an ontology for a virtual university to accomodate the new concepts.

3. DEFINITION OF OBJECTS REPRESENTING SETS OF STUDENTS

3.1 Groups

In our view a group of students corresponds to an administrative unit of the university, as it is registered by the university secretariate. The same concept is needed in a virtual learning environment as well, for example in order to manage student registration with courses, generate specific reports related to teaching activities, examination scheduling, etc.

3.2 Teams

Teams are defined within a course. We define a team as a set of students who cooperate on a project or assignment in a given course. Generally teams are subsets of a group, as defined in the previous section. Moodle provides functionality for team definition through a friendly user interface, shown in figure 1. Note that our concept of team overlaps that of "group" in Moodle.

3.3 Relationships

Figure 2 shows the UML class diagram of our student group object conceptualization. Groups and teams are sets of students. Teams are defined within a course and are used to group students who are working together on a project within that course. Typically teams are small (5 students on an average) groups. Groups are sets of students belonging to one year of study, defined according to administrative rules by the secretariate. A course has methods for creating teams (by adding students one at a time), importing teams as they are defined in another course and exporting teams to other courses.

4. IMPLEMENTATION ISSUES

In this section we present several issues related to the implementation of student sets (teams and

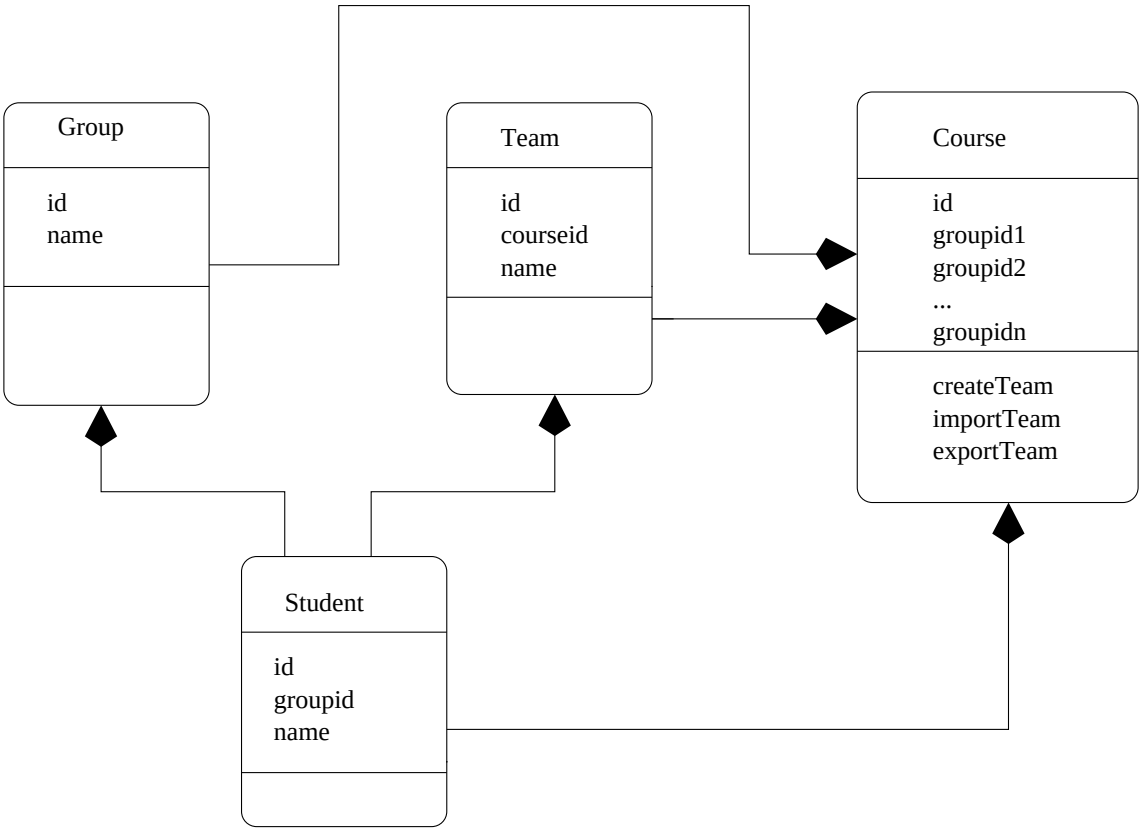

Fig. 2. Class diagram

groups) in Moodle, using MySQL and Java. The implementation of teams is straightforward, using the web user-interface included in Moodle. We approached group implementation in two different ways:

- Importing/exporting teams of students (Moodle groups) between Moodle courses
- Designing and implementing Maud, a Moodle database dedicated management system. Maud (Moodle Administrator's Utilitarian Device) is intended for Moodle administrators and includes functions for the definition and management of university student groups.

Example: importing team with name "An3ID" from course with id=2 (named "Java") to course with id=3 (named "C++"):

- add record (id=nrGrupuri+1, courseid=3, name=An3ID) into table mdl_group
- append to table mdl_groups_members the members of group 1 (named An3ID in course with name Java) , changing 1 with 4 (id of newly defined group) in the field groupid
- update table mdl_user_students

5. CURRENT RESULTS

Development is now under progress. An experimental server is running at the address https://194.102.42.1/moodle/ (Fig. 3).
Some courses, including "Moodle Features Demo", allow guest access. Other courses grant access only to registered users. Registration can be done by filling in a form available on the site. The server would then send an email requiring confirmation of the data submitted, by accessing a link provided in the mail. Some courses only allow the participation of registered users who possess a key given by the tutor. You can obtain the key by sending a message to the tutor (a "Send message" button exists on the page that opens when following the link associated with a tutor's name).

6. CONCLUSIONS AND FUTURE RESEARCH

Tracking and sequencing of student group's actions, as well as individual students' actions in the context of group work is not supported in SCORM, the most popular standardization initiative in the field of instructional technologies. However the newest pedagogical approaches promote learning in cooperation. We aim to define the necessary concepts and tools and to include them into Maud (a utilitarian program of our design for Moodle system administration). We will also investigate the possibility of using Moodle enhanced with Maud as a CSCW (Computer Supported Cooperative Work) environment and implement

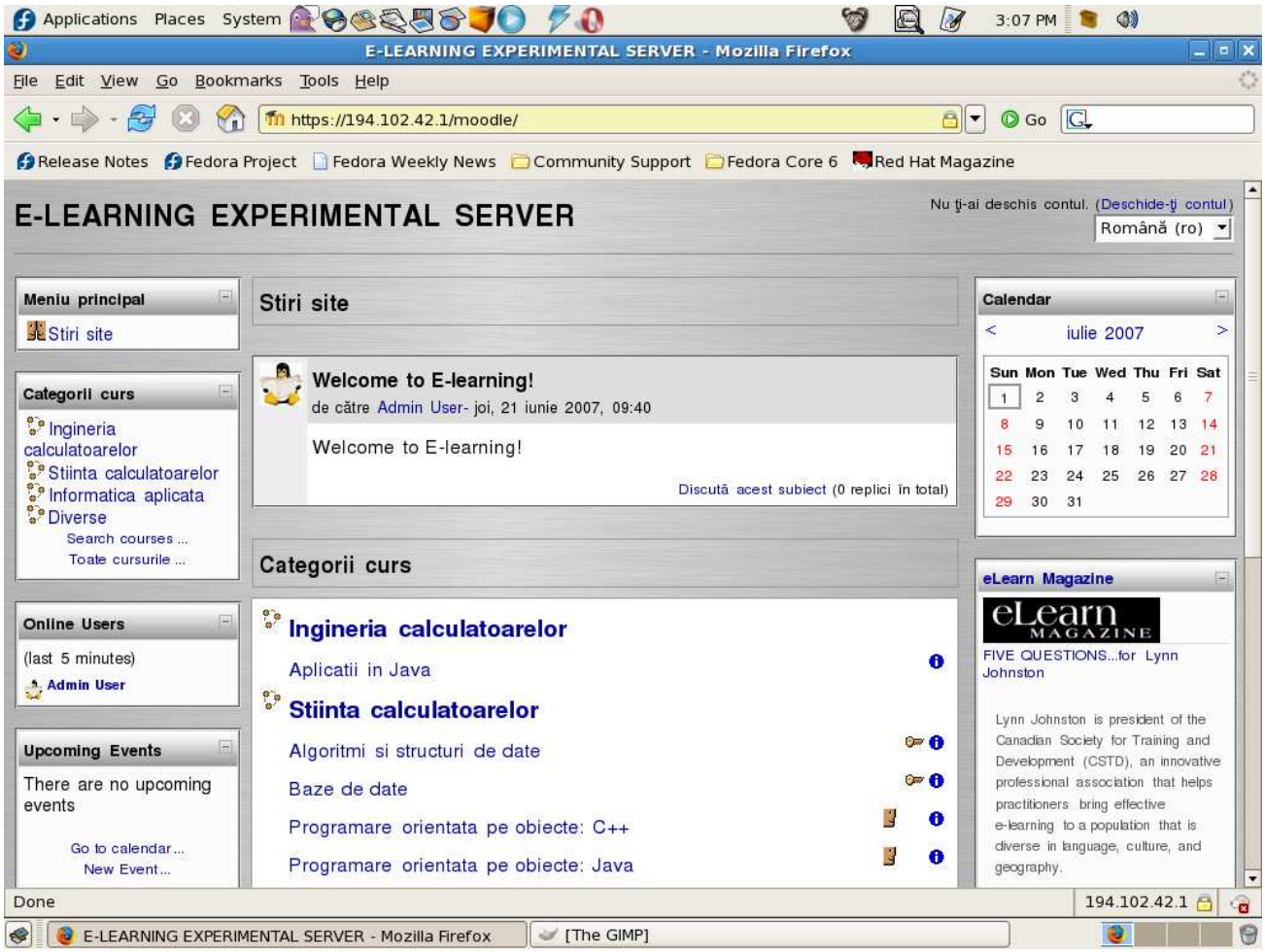

Fig. 3. Welcome to e-learning!

the results of our research into an information system for distance learning integrating teaching and administrative tasks (student group management, lecture and examination scheduling, examination result publishing, Web bulletin board, etc.). We are planning to create e-learning intelligent tutoring systems adapted for specific courses. We shall begin with e-learning platforms for C, C++ and Java programming and continue with virtual classes for data base (knowledge base) systems, computer network architecture and administration, operating systems. One issue that has to be dealt with according to specific characteristics of each subject is the testing method. Moodle offers support for the definition of on-line tests, assignments and quizzes, which we have been testing. We will also develop specific on-line testing tools adapted to the requirements of each subject. We are also planning the implementation of a messaging system for mobile phones, developed with NetBeans Mobility Pack (see NetBeans.org).

REFERENCES

Mihaela Brut. *Instrumente pentru e-learning. Ghidul informatic al profesorului modern.* Polirom, Iasi, 2006.

Cristina Fierbinteanu. Implementarea la Universitatea Titu Maiorescu a sistemului Moodle de gestiune a cursurilor bazate pe Internet. *Analele Universitatii Titu Maiorescu, Seria Stiinta si Tehnologia Informatiei*, 1:198–202, 2005.

moodle.org. Moodle-A Free, Open Source Course Management System for Online Learning. http://moodle.org/.

NetBeans.org. Introduction to the NetBeans Mobility Pack. http://www.netbeans.org/products/mobility/.

Victor-Valeriu Patriciu and Vasile Podaru. E-learning trends in european universities. *Analele Universitatii Titu Maiorescu, Seria Stiinta si Tehnologia Informatiei*, 1:56–68, 2005.

Aurel Serb. Architectures and standards for advanced distributed learning systems. *Analele Universitatii Titu Maiorescu, Seria Stiinta si Tehnologia Informatiei*, 1:32–49, 2005.

www.telscenter.org. Technology Enhanced Learning in Science (TELS) center. http://www.telscenter.org.