

## Different Uses of an Open Code LMS for Educational Support

**Katerina Georgouli, Michael Grivas, and Persefoni Zahariou**

**Abstract:** The aim of this paper is to present different ways of adopting and expanding an open source learning management system to facilitate learning and educational processes in the Department of Informatics of the T.E.I. of Athens. The flexibility and ease of such a system to incorporate Internet resources within the classroom and the capability of teaching beyond the classroom by providing out-of-class access to in-class assignments make it a valuable addition to a student's and instructors tool box. Furthermore, the adopted open source technology permits the easy development of different co-operating applications, supporting important educational and administrative procedures of the Department.

**Keywords:** Learning management systems, content management systems, higher education, educational scenarios, open source software.

### 1 Introduction

As teachers experiment with technology in the classroom, it is important to recognize that the role of educator has stayed essentially the same, to educate and inspire the students. While teachers can continue to be highly effective with the traditional lecture-style instructional method, most of them start feeling that they must attempt to stay up-to-date to technological advances, adapting to a more technology oriented teaching style, although this takes commitment and time [1].

One of these technological advances that educational institutions are using is course management technology, as an open source software or commercially provided. These e-learning systems are used for distance-learning and as supplement

---

Manuscript received February 2, 2004

The author are with <sup>1</sup> (e-mail: kgeor@teiath.gr), <sup>2,3</sup> (e-mails: [mgrivas, pzahariou]@cs.teiath.gr).

to in-class lectures by posting course announcements, homework assignments, lecture notes and slides on Internet-based platforms and are called Learning Management Systems (LMS) or Content Management Systems (CMS). In this framework, different techniques and tools have been tested in an effort to facilitate teaching processes and created electronic educational material at the Department of Informatics of the T.E.I. of Athens. One of these, first applied on the spring semester of 2004, adopted an open source software asynchronous e-learning platform, named *cs e-Class* [2].

*Cs e-Class* is a customized version of *e-Class*, an asynchronous e-Learning Platform developed by the Greek University Network (GUNET), which in turn is based on an Open Source software named *Claroline* [3]. The fact that *cs e-Class* is open source software contributed to its choice over other available e-learning platforms. Open source software (OSS) refers to software programs that are distributed with the source code - hence open source. The open source license allows users the freedom to run the program for any purpose, to study and modify the program, and to freely redistribute copies of the original or modified program. In the past few years colleges and universities have begun to produce enterprise open source applications like course management systems and electronic portfolios that compete directly with their proprietary counterparts. These e-learning applications are leading a movement in higher education from proprietary software toward open source. Open source software (OSS) has been extensively used by the Department of Informatics because it gives the students the opportunity to work on and develop code, which would otherwise be difficult to build from scratch, and gives them the chance to communicate and cooperate with people all over the world working on the same code. Furthermore, OSS can be easily modified to fit the needs of the Department, offering the programmers the possibility to develop additional functionality at their own pace without begging a proprietary vendor to include a feature. The software had so far a satisfactory performance and any problems and bugs that have been identified have been quickly corrected, sometimes with the help of other developers from the open source community [4].

*Cs e-Class* allows teachers to create and administer courses through the web. Additionally, it provides a rich collection of services/features including group management, forums, document repositories, calendar, chat, assignment areas, links, user profile administration etc. Having already used most of the services of *cs e-Class* for two semesters we have been exploring the different ways this platform could be used beyond the basic functionalities it provides. There is work in progress on developing working procedures to use *cs e-Class*, as it is, to facilitate existing manual tasks. The in-class and out-of class support adopted by the course of Artificial Intelligence and the innovative registration procedure and the homework

assignments procedure for the labs of the same course are examples of successfully supported tasks by cs e-Class. The lab's examination procedure for the course Database I, is one more task that has been successfully preformed with the aid of cs e-Class, where an easy and secure way has been designed for the testing process to be conducted, using the group management feature of the platform and the facilities offered by the Department LAN.

The spirit of open source is formed around diversity of input, recombination of ideas, creativity, and collaboration. These are essential ingredients for innovation and clear advantages to the open source philosophy. According to that another scenario has been looked into. This is building smaller applications that cover special functions, beyond those which cs e-Class is designed to offer. These applications might either work separately, using the main platform to meet some specific needs, or be incorporated into it as modules. Several such applications are being built. The most interesting of them is a Graduation Theses Management System (GTMS), a system that fulfils the needs of theses proposal, publicising, assignment, control and presentation for the Graduation year thesses of the students.

## **2 Innovations Adopted by AI Course**

In AI course we have very seriously re-thought the lesson plan in order to see where technology can be effective. The transition from traditional lecture style to a more technology-based format was not difficult because we already possessed relevant course material. We have first used cs e-class as supplement to in-class lectures by posting out of class information like the agenda of the course, announcements, lecture notes, homework assignments and interesting links. The latest include everything from news, articles, relevant educational sites and interesting commercial sites. Then we tried to find applicable in-class information like Internet resources for demonstrations and we also designed our own web-based applications for the same reason. Most of these home made applications are made as homework assignments by the students who have attended the course.

At the eginning of each semester, students have to register at the course in order to have the right to upload their homework assignments and to apply to the course labs. For being prepared to take notes during lecturing, the students can visit the platform at the beginning of each class, go to the appropriate folder and access the outline and the slides of the lecture. The offered information is public and can be accessible by any visitor.

More sophisticated scenarios of cs e-Class use have been designed in order to support labs. Four are the main issues concerning laboratories running: applying

for registration, in-class training, out-of-class assignments and exams. All of them need a very good organisation in order to be accurate and not time-consuming and in all cases the support offered by the functionality of the platform has been proved very efficacious.

More precisely, for labs' administration, we used the Group facility to organise the students in different groups of a specific number of participants each and then we announced to the students the exact time after which the registration to these groups would be available. Thus, the students had no more to wait in long queues out of the lab's door for being registered. The information about the remaining free places in each of the groups was always available on-line so students could make their application decision at their pace from their home or using the free lab room of the Department or even visiting an Internet café.

For lab in-class and out-of-class- assignments everything is posted in advance by the instructor of the lab and remains invisible until the beginning of the class. Students can visit and download all needed information and upload their answer before the end of the class or of any other existing deadline. After the instructor corrects the assignments students can view their own marking and the instructor' comments posted on their assignment page.

For all other needs of the lab and especially for the exams procedure the platform has been used in the same way as in Database I course described in the next paragraph.

### **3 Laboratory Examinations In Database I Course**

Cs e-Class has been used for the laboratory session of the course and particularly for the examinations. The course introduces the basics of Databases in two parts: Database Management Systems and Relational Databases. One of the course's objectives -and the main for the laboratory session- is to establish a good knowledge of SQL and a clear understanding of the use and manipulation of Tables in Relational Databases [5]. Since laboratory sessions are based on the practice and experience by use, the laboratory examinations must be done on computers and specifically the laboratories' equipment, which includes PC systems connected to a LAN and -through a gateway- to the T.E.I. network. Databases may be locally built (independent DBMS) or centrally controlled.

For the examinations the students have to show their ability to use SQL, for a given database. Those commands are divided into three groups: database creation, data manipulation and queries. Those groups are quite independent to each other making the marking more difficult since a query may be wrong compared to the

proper schema but correct under the assumptions of the student' and for his/her created database, making automatic correction and marking practically impossible. Another difficulty is the available laboratory space, which entices students to cheat. Cs e-Class has been used to solve both problems.

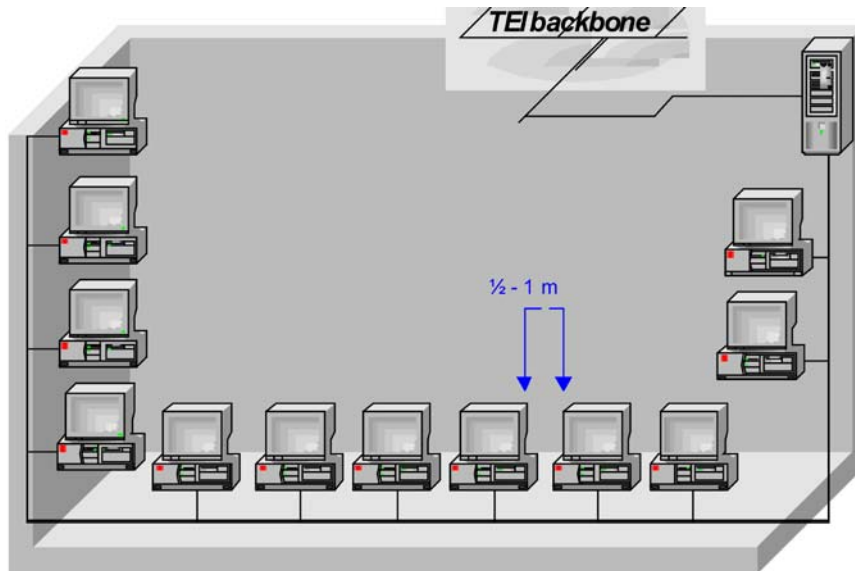


Fig. 1. Laboratory setup - lack of space

The major issue of human-based corrections and marking cannot be changed from its roots. However, it can be thoroughly assisted. The usual way for written exams is the collection of the papers and the scrutiny of each one of them, a process that is quite tiring and time-consuming, as any examiner knows. Several solutions have been proposed, including multiple-choice papers etc [6, 7]. Although, such tools have proven their value [8], they cannot usually apply to cases where the examinees' responses are not strictly limited or directed [9, 10]. Furthermore, laboratory exams cannot -or better should not- be done on paper, since students' ability to use the computer is also evaluated. For programming code, automatic execution could be plausible, a case which has not been studied extensively [11, 12, 13]. Such system would be limited to a set of proper answers, inflexibly prohibiting any relatively correct or properly approached response, which is not the best for engineering students [14]. This case is under investigation and cs e-Class' ability to use external systems and embed their functionality to the overall system will be studied.

No matter how the examiner evaluates the answers, it is crucial to have a simpli-

fied and effective way of collecting students' responses. At a first glance, a shared disk space, allowed to all students for writing-only would seem easy and solving the problem. A better solution might be private disk spaces allowing reading to the examiner. The major drawbacks of such a solution are that there must be a clear understanding of where the disk space resided and how it can be used (from both examiners' and examinees' sides), that a file server is always more vulnerable than a web-server [15, 16, 17, 18], that there must be an extra functionality for the server to automatically -or when examiner decides- cease accepting write requests from all or some students and that it does not help relate students' responses to questions, especially if there are different question sets for different students.

So is introduced the second major issue, which is the avoidance of cheating or the simplification of detection among suspicious similarities in responses. That can be easily solved with double or multiple examination papers. Additionally, examination tasks that differ slightly and in such a way that one cannot tell at from a simple glance, offer a good threat against cheating and a clear distinction between original and copied thoughts. Such examination papers require proper handling of tasks and responses.

### **3.1 Web-Based Tasks**

Cs e-Class provides combined solutions for both issues. Regarding the collection of examinees' responses, cs e-Class has a fully equipped uploading facility that can distinguish who saves the file [2, 3] and does not require any kind of specific knowledge or protocol regarding file naming etc. Cs e-Class is a web application, meaning that it is flexible and versatile for the user, robust and easy to maintain from the other side. Things are simple and straightforward and any user connects to it using a simple web browser over any Internet connection. Security also is quite ensured since it is based on well-defined and proved protocols and programs [19, 20].

The examinee sees the examination paper on the screen and he or she can copy any given material (i.e. programming code) without typing hassle or errors. He or she can accordingly upload the responses as file, while cs e-Class provides additionally the ability for on-line questions and answers, should one use such a method. The examiner can upload the examination papers any time and the same way that an examinee can upload the responses file. He or she can also download examinees' files, accordingly, knowing who uploaded each file, without having to enforce a specific way of work or naming.

Our use of cs e-Class had proved very helpful, based on our detailed study of these two issues. Cs e-Class can embody other routines or programs for additional

functionality. Each computer in our laboratories has a specific IP address, which comes out of a simple algorithm. It is ensured that each neighbouring computer has following IP addresses, meaning that neighbouring computers will always have an odd and an even IP. In such case, an add-on to cs e-Class distinguishes which IP the examinees computer has and shows him the respective examination paper, all from a single URL [21, 22, 23]. That gets even more complicated by including time -or more specifically time zones- to the election of paper. So, all examinees (of all classes) contact the same server, the same URL, but they get different papers. Having the tasks in each paper resembling very much but being clearly different to those of the neighbouring examinee, makes it practically impossible to cheat and straightforward for the examiner to detect any plagiarism.

In Databases I the examinee usually submit three text files that include the SQL commands that respond to the given questions. Sometimes extra questions may be placed that the students have to respond in a more traditional way. Apart from the previously described procedure, the examiner may have also a look to the results (data) as placed in the database for an immediate picture of the results.

This solution has proved to be very helpful and very simple to use and administer. Maintenance is also simplified, since past papers may remain in the cs e-Class server but only to the professor and/or examiners' disposal for reference or later use. A pool of tasks to choose among arbitrarily is also possible.

#### **4 Graduation Theses Management System (GTMS).**

For Graduation Theses supervision, the professors expressed the wish to be able to publish a list of (a) suggested topics and (b) information on the ongoing Graduation Theses they are supervising. This would eliminate the need of posting such information on the message boards, save the students the time of meeting with each professor individually and save the professors the time of going over the same grounds with each student who was interested in starting his/her Graduation Thesis.

The e-learning platform presently used, cs e-Class, as with other Learning Management Systems, has not been designed for this purpose. So, the lists of Graduation Theses, supervised by each professor, were uploaded as static HTML pages or as downloadable documents, rendering even the smallest changes hard to make. So, it was decided to build a Graduation Theses Management System (GTMS) solely for the management of the Graduation Theses of the Department, sharing the same users' Database with cs e-Class.

After the initial requirements' analysis the project was found to be bigger than thought initially and it should cover the needs of three categories of users, (a) pro-

fessors who would like to upload and administer their students' theses, (b) students who would like to have an easy way of browsing through and searching for theses and communicating with the supervising professors and (c) the secretariat who would like to automatically receive application forms for the commencement and the presentation of the theses, as well as being able to administer and search through the application forms.

To build our system, first, web-design was chosen over client-server technology because of its simplicity of use and the ease with which one can access the application from any computer with an Internet connection and a conventional browser instead of client applications. Also, both professors and students of the department are already accustomed to such an interface. Our experience has taught us that users are difficult and unwilling to be educated in the use of new applications. So, we decided to keep things as simple and as close to the existing applications as possible, without sacrificing anything in the way of functionality. Lastly, our application should easily coexist with or be incorporated in the already existing web based applications or those which will be designed in the future, sharing their LDAP (Lightweight Direct Access Protocol).

#### **4.1 Functionality of the GTMS**

The system we designed based on the four basic users the professor, the student, the secretariat and the administrator. The user visiting the home page can login and perform tasks specific to his user type. The main functionality of the application though, which is searching for theses, does not require the visitor to be logged in.

Searching is practically the only action an anonymous visitor can perform. There is a set of ready queries by theses status (suggested, ongoing, to be presented). Additionally, an interface is provided through which one can search through the database using a broad set of criteria that refer to almost every attribute of the thesis; from a drop down list of professors' names, to supplying a keyword to be found in the thesis summary.

Additionally, the home page lists the theses we consider to be most interesting. Those would be the most recently posted and the ones that will be presented in the close future. Some short announcements/news are also available on this page. If the user wishes to accomplish specific tasks he can login to the system. Care has been taken to design the login page to prevent automatic logins using a security code and timing out the user after continuous failed login attempts in a short period of time. Let us note that the username/password used to enter the system are not stored in our applications database, because, both and most other user data is stored in the current LMS database and we have created a wrapper to access that data. This de-





Fig. 2. The application's home page

creates our control on the application and relies on the other systems authentication methods; with all the advantages and disadvantages this may impose. However, it was considered more important to give the user a unique username/password set thus decreasing the need for administration of such data and simplifying things for the user and the Institution. At this point let us describe the different types of users and what each of them can and cannot do. Basically, there are four categories: the professors, the students, the secretary and the system administrator.

(a) The professors. These are the most important users of our application. Considering the professors accounts are well checked before they are created in the LMS so than no student or other person can get such an account and that the professors of the department are computer savvy and responsible; they have full administration rights for the thesis they create for their students or are part in. The whole idea of this Graduation Theses Management System is anyway, to put the power in the hands of well-informed and responsible users and let things run themselves. So a professor can create a new thesis and administer the ones he has already created. That includes changes made to the thesis description, setting the students who will work on it, setting start and presentation dates and uploading files some of which will be viewed by the public and some of which are private only to the professors and students that are working on the particular thesis.

(b) The students. Students, on the other hand, cannot be allowed to have such rights. With that in mind there is only one thing they are allowed to do. Upload

a private file for the thesis they are working on. Thus, the professor can delete it or set its status to public. There is no easy way, we can imagine, any harm can be done by the student-user.

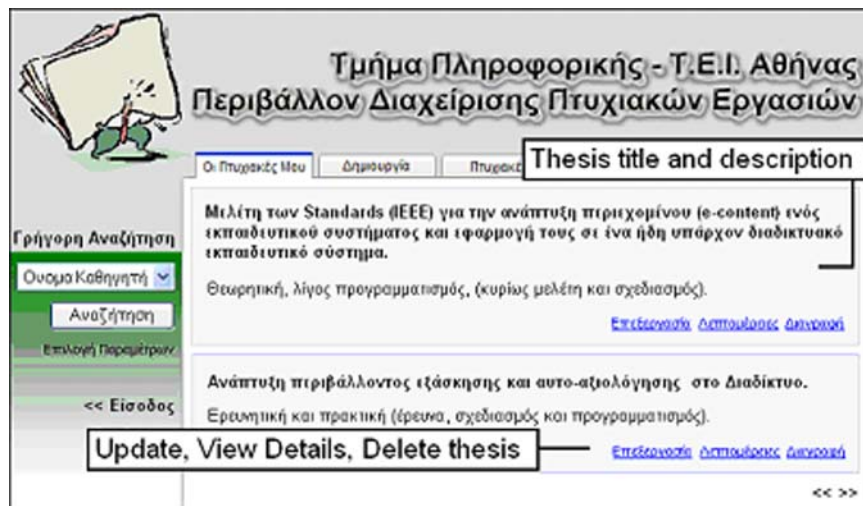


Fig. 3. The professor's interface

(c) The secretariat. A single user account is given to the secretary of the department. Whenever a professor sets a thesis start date, the thesis appears on the secretary's commencement applications form' list. Similarly, when a professor sets a thesis presentation date, it appears on the 'presentation application form' list. The secretary can view and print out these forms. The data for the forms is filled in from the data entered for the thesis so there is no need for the students to fill in a paper and submit it. New forms are marked and remain so till the secretary chooses to mark them as 'done'. It is easy to search through the forms as they are categorized according to type (commencement, presentation), year and month. In case such a search is not sufficient an extended search can be done too, using keywords, date, students' names etc.

(d) Despite the common practice we did not consider it necessary to give any special rights to the system administrator. The reasons are: a) it is not the administrator's responsibility to manage theses, b) the administrator doesn't need to manage users either because user management is performed strictly by the LMS administrator. To sum up, the system administrator can only upload news. This is also good if we want one person (a secretary or a student trainee) to simply upload news but have no rights to handle the database.

## **4.2 Implementation issues in GTMS**

The application was developed on a Pentium 4 2.4MHz, Fedora Core I Linux, Apache, PHP and mySQL.

The first thing considered when choosing the Apache/PHP/mySQL combo was the low cost of setting up our web server and application. The servers OS and all the software used were downloaded from the Internet free of charge. On the other hand, in order to use an ASP-like or other proprietary platform one needs a Windows or other specific server, (e.g. IIS and SQL Server) all of which are usually quite expensive. Also, in case of hardware unavailability Apache/PHP/mySQL can easily run on any older and cheaper hardware that can be found.

Second, PHP does not require a deep understanding of a major programming language to perform basic functions. Although it can be considered a programming language that supports most features of object-oriented programming, especially PHP5, it is easy to learn, especially for a student, since a lot of projects in our department are worked on and expanded by trainees. Hence, it could be used as an educational tool.

Third, the PHP/MySQL combo runs on most operating systems and is compatible with the three leading Web Servers (a) Apache HTTP Server for UNIX and Windows (b) Microsoft Internet Information Server and (c) Netscape Enterprise Server (iPlanet Server). It is fast in its execution especially when compiler as an Apache module on the UNIX side and MySQL server executes very complex queries with huge result sets in record-setting time [8].

Finally, PHP is becoming increasingly popular. There are strong user communities, as with other OSS, and people on the mailing list are available to answer questions and help debug code. This procedure helps students develop team skills.

## **5 Conclusions and Future Work**

The demands on higher education require a fundamental change in direction—and technology can facilitate that change. Open source e-learning platforms have paved a new road changing the existing ways of teaching and learning.

Cs e-Class open source e-learning platform has proven a valuable, extensible, versatile and powerful tool that can assist in many educational tasks in our department. Although its main use remains the educational content management, other uses can be supported with additional embedded or collaborating software.

A number of educational scenarios have been designed with success in order to use cs e-Class for out-of class information, educational content repose and re-

trieval, in-class supporting material during course lecturing, and mainly for labs registration, assignments and examinations.

Extra utilities have been added to provide missing tasks. A Graduation Theses Management System has been built specifically for the publication and control of the final-year theses. Each professor can publicize a theme for a potential thesis; he can accept messages, assign the thesis to one or more students, grant access to specific users, monitor its progress and finally have it published as a finished thesis for reference.

In the end of last winter semester, a questionnaire has been delivered to the students who attended 4 courses and who have been asked to denote their satisfaction or their discomfort of the implemented until now innovative educational procedures supported by cs e-Class. After the completion of the indexing of their answers, the existing problems will be localized and investigated in order to design more acceptable educational scenarios for the next semester.

Furthermore, now, after two semesters of use of the platform, we can invest in the teachers experience through thorough training, customization, and increase of the level of adoption. We intend also to initiate a pilot program proposing innovative teaching procedures for all courses based on this experience. Our imminent goal is to convince all those of our colleagues who are still afraid of technology to get involved, which is the real reward for all this effort.

## References

- [1] E. Weis and J. Efav, "Using blackboard, instead of a blackboard in the classroom," in *Proc. of IADIS International Conference Cognition and Exploratory Learning in Digital Age (CELDA 2004)*, Lisbon, Portugal, 2004, pp. 149–156.
- [2] [Online]. Available: <http://eclass.cs.teiath.gr>
- [3] [Online]. Available: <http://www.claroline.net>
- [4] C. Copolla and E. Neelley. (deposited on 23 August 2004, last visited 2 Febr. 2005) Open source - opens learning: Why open source makes sense in education. [Online]. Available: <http://dlist.sir.arizona.edu/archive/00000453/>
- [5] C. Skourlas, *Relational Databases*, 2000, iISBN 960-8105-14-5.
- [6] J. Mahe, "The tal project - using the web for computer aided assessment," in *Huddersfield Workshop on Designing and using Multiple Choice Assessment*, 11 July , 2000.
- [7] C. N. Mills, "Development and introduction of a computer adaptive graduate record examination general test," in *Innovations in Computerized Assessment*, F. Drasgow and J. Olson-Buchanan, Eds. London: Lawrence Erlbaum Associates, 1999, pp. 117–135.

- [8] G. Armenski and M. Gusev, "Using etesting methods in cs education," in *Proc. of Workshops on Computer Science Education, TEMPUS project CD-JEP 16160*, M. Stojcev, Ed., 2001.
- [9] D. Green, M. Harrison, and J. Ward, "Mathematics for engineers - the helm project," in *Proc. of Progress 3 Conference on Strategies for Student Achievement in Engineering*. UK: Hull, Sep. 25-26 2003, pp. 26–31, ISBN 1-904250-01-7.
- [10] A. Finkelstein, "Student problems in software engineering education," in *IEE Colloquium on The Teaching of Software Engineering*, 1991.
- [11] S. Benford, E. Burke, E. Foxley, N. Gutteridge, and A. M. Zin, "Ceilidh as a course management support system," *J. of Educational Technology Systems*, vol. 22, no. 3, Sep 1993.
- [12] E. Foxley, C. Higgins, P. Symeonidis, and A. Tsintsifas, "The coursemaster automated assessment system - a next generation ceilidh," in *Conference on Computer Assisted Assessment to support the ICS disciplines*, University of Warwick, April 5th - 6th 2001.
- [13] M. Joy and M. Luck, "Changing the delivery of computer science education," in *Proc. of the 6th annual conference on the teaching of computing and the 3rd annual conference on Integrating technology into computer science education*, Dublin City Univ., Ireland. New York, NY, USA: ACM Press, 1998, pp. 134–138, ISBN:1-58113-000-7.
- [14] D. Green, M. Harrison, A. Palipana, D. Pidcock, and J. Ward, "The helm mathematics learning and assessment regime for engineering students," in *Proc. of International Conference on Engineering Education*, Gainesville, Florida, Oct 2004, pp. 16–21.
- [15] J. A. et al., "Securing network servers," in *CMU/SEI-SIM-010*. Carnegie Mellon University, USA: Software Engineering Institute, 2000.
- [16] S. Henry-Stocker. (2000, Oct 6.) Square one, paring down your network services. [Online]. Available: <http://www.sunworld.com/sunworldon-line/swol-10-2000/swol-1006-buildingblocks.html>
- [17] J. Sellens, "System and network monitoring," *J. login*, vol. 25, no. 3, Jun 2000.
- [18] D. S. et al., "Securing desktop workstations," in *CMU/SEI-SIM-004*. Carnegie Mellon University, USA: Software Engineering Institute, 1999.
- [19] Anonymous, *Maximum Security: A Hacker's Guide to Protecting Your Internet Site and Network*. Indianapolis: Sams.net Publishing, 1997.
- [20] M. M. Chaddock. (2000, Oct. 11) A breakdown of sans top ten threats. SANS Institute. [Online]. Available: <http://www.sans.org>.
- [21] L. Masinter, H. Alvestrand, D. Zigmund, and R. Petke. (1999) Rfc 2718: Guidelines for new url schemes. IETF (Internet Engineering Task Force). [Online]. Available: <http://www.w3c.org/>.
- [22] T. Powell and F. Schneider, *JavaScript: The Complete Reference*. McGraw Hill, 2004, ISBN: 0072253576.

- [23] T. Converse, J. Park, and C. Morgan, *PHP5 and MySQL Bible*. Wiley Publishing, 2004.