A flexible modularity-based course management system

Xiaoran Wang\textsuperscript{a}, Banu Yaman\textsuperscript{b}, Coskun Bayrak\textsuperscript{a}

\textsuperscript{a}Computer Science Department, University of Arkansas at Little Rock, Little Rock, Arkansas, USA
\textsuperscript{b}Department of Educational Sciences, University of Aksaray, Aksaray, TURKEY

Received October 25, 2008; revised December 24, 2008; accepted January 5, 2009

Abstract

Course Management Systems (CMS), a kind of eLearning system, have become an important role in education. The trend towards more flexibility and intuitiveness in eLearning systems makes the design and development of course management system increasingly complex. In order to achieve a higher quality for course management system, we introduce a conceptual model -Environment of Knowledge Tracking and Extension Process (MEKTEP) - that not only promotes reusability for efficient enhancement and functionality but also serves as architecture for designing, realizing, and evaluating eLearning process.

© 2009 Elsevier Ltd. Open access under CC BY-NC-ND license.

Keywords: CMS; RBAC; Modurality.

1. Introduction

Course Management Systems (CMS), a kind of eLearning system, have become an important role in education. Not only can it provide paperless teaching, but it is also capable of promoting not-in-classroom teaching. With the emerging multimedia technology, those systems are progressively gaining more important role in schools. Nowadays, most schools have adopted a commercial version of CMS or developed one in-house. Those systems have provided great advantages in the process of teaching and learning. However, the trend towards more flexibility intuitiveness in eLearning systems makes the design and development of CMS increasingly complex and time consuming. Therefore, in order to minimize the complexity and time consumption, we have developed an environment to promote reusability for efficient enhancement and functionality.

2. Related Work

Currently, there are many CMS products in the market. With the increasing complexity and cost, they are more powerful and user-friendly. The open-source feature lets some CMS products receive feedback and technical support from a large scale of users. The following are some of those CMS products in market.

WebCT is an online proprietary virtual learning environment system that is sold to colleges and other institutions and used in many campuses for eLearning. Instructors can add tools such as discussion boards, mail systems and
live chat to their WebCT courses, along with content including documents and web pages. However, the increasing licensing fees for the organizations and learning cycle for the users introduces some difficulties.

ATutor is an Open Source Web-based Learning Content Management System (LCMS). ATutor is used in various contexts, including online course management, continuing professional development for teachers, career development, and academic research. The software is cited as being unique for its accessibility features, (useful to visually-impaired and disabled learners), and for its suitability for educational use according to software evaluation criteria established by The American Society for Training and Development (ASTD). ATutor is used internationally and has been translated into over fifteen languages with support for over forty additional language modules currently under development.

Moodle is a free, Open Source software package designed to use sound pedagogical principles to help educators create effective online learning communities. You can download and use it on any computer you have handy, yet it can scale from a single-teacher site to a university with 200,000 students. Its open source license and modular design allows users to develop additional functionality.

These systems are either expensive or too complex to be used if it is open-source. The reason for this is that the developing groups normally want to provide as many functions as possible, which makes some functions not useful for some schools. Therefore, there is a high demand of CMS which can provide flexibility based on different schools while at the same time provide development simplicity. The Environment of Knowledge Tracking and Extension Process (MEKTEP) project includes those features.

3. The Design of MEKTEP

Environment of Knowledge Tracking and Extension Process (MEKTEP) is an architecture which provides flexibility and modularity. These features extend the system to support Delegation, Ajax, and Third-party JavaScript libraries.

A. Modularity

Modularity is the most important feature of MEKTEP. While the existing CMS products are providing powerful functionalities, the requirements of different schools may not be the same. When an organization uses one of the existing CMS products, they may need to add some modules. MEKTEP has a flexible plug-in mechanism which makes another module able to be easily plugged into the system. Figure 1 shows the architecture of MEKTEP. The MEKTEP Kernel is the foundation of the whole architecture. It provides the event handling for all actions in this system. The Theme, User, Plugins, and Watchdog are the main modules in the system. Each module is an object which is formulated in the initiation stage. With the Theme, each user can change the display mode which provides accessibility when it is extended. The User module provides the functions of authentication, delegation, authorization. The Plugins module controls the working modules based on the requirement of the school. Watchdog records all actions when the system is in use.

Figure 1 Multilayered Architecture of MEKTEP
MEKTEP kernel is responsible of replying any actions triggered by each module. It forwards each action to appropriate object to process. When a new module is plugged into the system, the Plugins will add the information of the module into the database, including the version, description, and author. Then the user can use the new module. All the new modules have the same components including description, initiation, display, and trigger, so developing a new module is convenient. Figure 2 depicts the structure of each module. After the new module is plugged into the system, it will communicate with the MEKTEP kernel through Plugins Control.

Using the Plugins Control module, adding or removing a module is quite simple. The system will discover all available modules, and displays the related information for the administrator to decide which modules are required. Figure 3 shows the Plugins Control's user interface.

B. Role-based Access Control and Delegation

Role-based Access Control (RBAC) is an alternative to the traditional access control method, which is gaining increasingly more attention. It becomes the most promising solution due to its inherent richness in modeling hierarchy, separation of duty, cardinality, and dependency constraints. There are four basic components: a set of users, a set of roles, a set of permissions, and a set of sessions [8]. In an eLearning system, there are many different users, such as teaching assistant, instructor, dean, administrator, etc. This can form a role hierarchy in which high level users inherit lower level users' permissions. Delegation is an important business rule. Sometimes, an instructor may want to let his assistant to do a job representing him. When he needs to get the job done on his own, he can reacquire the permissions he grants. In this kind of situation, delegation can provide flexibility in the system and improve the system's effectiveness greatly. In order to add delegation functionality to MEKTEP, we designed a simplified delegation model. We first use an example to show the delegation process. Figure 5 shows an example of delegating a permission to a user. When Jack wants to delegate some permission to Mary, it will follow the three steps:

1. Jack creates a temporary delegation role “DR”.
2. Jack assigns the permission “confirm_program” to “DR” with permission-role assignment.
3. Jack assigns Mary to “DR” with user-role assignment.
We define which permissions are regular (unable to be delegated) and which are able to be delegated. The regular permissions cannot be delegated while delegating permissions are allowed to be delegated to other users. The modified delegation model is showed in Figure 6.

Figure 5 Process of a Sample Delegation

Figure 6 Simplified Permission-based Delegation Model
C. Ajax

Ajax (Asynchronous JavaScript and XML), or AJAX, is a group of interrelated web development techniques used for creating interactive web applications or rich Internet applications. With Ajax, web applications can retrieve data from the server asynchronously in the background without interfering with the display and behavior of the existing page. Ajax can greatly improve the user interface. It makes Web-based applications look and work like desktop software, even though they are browser-based and use few resources. Many existing modules in MEKTEP have been implemented using Ajax. In the project module, the user can drag students from Groups to a project. The database will be updated in the background. The Ajax technology greatly improves the user interface.

D. Accessibility

Web accessibility allows people with disabilities to use the Web. More specifically, Web accessibility means that people with disabilities can perceive, understand, navigate, and interact with the Web and that they can also contribute to the Web. Web accessibility also benefits others, including older people with changing abilities due to aging. Accessibility in MEKTEP is ensured by the flexible theme Plugins mechanism. Based on different requirements of different people such as older people, disabled people, etc, the system provides different displays. The display is controlled by the Theme module. Fundamentally, the MEKTEP Theme system is a way to “skin” the course management system. Yet, it is more than just a “skin”. Skinning a site implies that only the design is changed. MEKTEP Themes can provide much more control over the look and presentation of the material. A theme is a collection of files that work together to produce a graphical interface with an underlying unifying design for the system. A theme modifies the way the site is displayed, without modifying the underlying software. Themes may include customized template files, image files (*.jpg, *.gif), style sheets (*.css), custom Pages, as well as any necessary code files (*.php). With the Plugins mechanism, a new theme can be easily plugged into the system. Some accessibility issues may need to be addressed by the professional accessibility companies. In this case, after the new theme is developed, it can be applied into the system without any modifications to MEKTEP.

4. Conclusion

In this paper, we presented a design of a system architecture that provides great flexibility and modularity. To make the system more user-friendly, we introduce Ajax technology and third-party JavaScript Library. Our functionality of delegation greatly improves the system’s performance when a teacher or administrator needs to get a job done by someone else. The simplicity and flexibility makes the system also open to the currently popular technologies. In this way, a school can easily have a well-suited CMS system.

5. Acknowledgement

The authors would like to thank Mr. Jiang Bian, Marco Santos, Melanie Thornton, and Mickey etc. for his/her help. This funding for the project was provided by the PACE project (Fund #000000).

6. References

http://www.blackboard.com/us/index.bbb,
http://www.jquery.com/,
http://en.wikipedia.org/,
http://www.atutor.ca/,
http://moodle.org/,
http://www.w3.org/WAI/intro/accessibility.php,