OLMS: Online learning management system for e-learning

Vinay Kumar Ippakayala*, Department of Computer Science, Lakehead University, 955 Oliver Rd, Thunder Bay, ON P7B 5E1,

Hosam El-Ocla, Department of Computer Science, Lakehead University, 955 Oliver Rd, Thunder Bay, ON P7B 5E1,

Abstract

In this paper we introduce a learning management system that provides a management system for centralized control of course content. A secure system to record lectures is implemented as a key feature of this application. This feature would be accessed through web camera and mobile recording. These features are mainly designed for e-learning environment. In addition, we integrate a learning management system with a platform for social activities and student assessment analysis is developed. Users Including students, professors, and administrators are able to manage schedules, lectures, assignments, jobs, events, discussions, and research.

 Keywords: Learning, management, online, application.

*ADDRESS FOR CORRESPONDENCE: **Vinay Kumar Ippakayala,** Department of Computer Science, Lakehead University, 955 Oliver Rd, Thunder Bay, ON P7B 5E1.
E-mail address: vkumar@bsd.uchicago.edu

1. Introduction

New technologies have the potential to change the way teachers teach and learners learn (DeNeui & Dodge, 2006). There is hundreds of learning management systems (LMS) applications in the market such as eSSential, Bridge, Bloomfire, Edmego, Litmos LMS, etc. Many of these packages are designed primarily for commercial use such as training centers while some focuses on providing social media more than learning features. On the other hand, Blackboard Learn application is well reputed solution for learning purposes (Heirdsfield, Walker, Tambyah& Beutel, 2011). Both academic staff and students may benefit from using Blackboard as one of the most famous LMSs. Potential benefits include increased availability, quick feedback, improved two-way interactions, tracking, and building skills such as organisation, time management and communication (Meurant, 2010). Over the past 5 years, Blackboard has encountered challenges competing with freeware products such as Edvelop and Moodle in addition to D2Learn (Wang & Shao, 2012; Mbuva, 2015; Smart & Meyer, 2005). Although the Blackboard LMS has shown growth, its drawbacks are a great concern. These drawbacks involve: (a) the software is harder to learn than expected; (b) Certain options may be restricted to specific operating systems; (c) there are inefficiencies in bandwidth use when materials have to be downloaded every time access is sought; and (d) cost (Chawdhry, Paullet & Benjamin, 2011). Some studies were conducted for a comparison between Blackboard and D2Learn technologies and concluded that D2Learn is more used than Blackboard (Claar, Portolese & Shields, 2014; Curtis, Krasner & Iscoe, 1998). On the other hand, many universities and colleges have eyed on the Canvas LMS as it is appealing to college professors and students (Chawdhry, Paullet & Benjamin, 2011).

There is an increase in online course involvement by students and faculty and that underscores the importance of using an effective LMS (Borup, 2010). Such needs of both learning and social media features are fairly combined in some existing LMS applications but still they provide a marginally satisfactory performance.

This paper provides an ultimate service to enhance educational process in the professional institutions through integrating essential learning and social media functionalities in one mobile application. In this regard, we introduce an Online Learning Management System (OLMS) that develops a learning platform for an efficient course delivery and management. OLMS provides new course content management system while improving the service of social activities provided in some other applications such as D2Learn.

In doing this, we present main features of OLMS including live lecture recording techniques (mobile and web camera recordings). This has been integrated into the application to improve the content management scope while the Blackboard and D2Learn have no such advanced ability. As a result, we combine and develop key features of different packages to be in one stand-alone application and therefore we reduce the cost of multiple applications and as well cutting off the cost of the hardware parts needed as in YuJa. OLMS also will be easily used by non-technical users than those requiring advanced skills. Lecture recording is conducted by a user authorized by the course instructor/administrator to maintain the security constraints. Recorded lectures are published and accessed by registered students through uploading to OLMS server from local storage unit or from cloud database. In this application, content and user management features are mainly essential for the educational advantages particularly e-learning environment. Consequently, those students who learn remotely can watch lectures presented by instructors. On the other hand, social media abilities are facilitated through the notice board feature in the application where the user can view, share, and contribute to the learning and research related requests. OLMS is a duel application which is available on the web as well as in the form of Android mobile application.

2. Methodology

OLMS provides options once the user completes its installation including login, sign up, and forgot password. Once the user is login in, he will be redirected to the home page of the application. The software design phase implements a systematic development of the application. As indicated in (Deacon, 2009) a substantial design effort is spent coordinating a common understanding among the

staff of both the application domain and of how the system should perform within it. In order to develop our web and mobile application, we have used Ruby on Rails system which we have used as a backend processing along with PostgreSQL database (Wargo, 2015). Ruby on rails is an open source web application framework written in Ruby. It uses the Model–view–controller (MVC) pattern to organize application programming (Rumbaugh, Booch & Jacobson, 1999). It creates a new application with all required resources and available library's called as Ruby Gems. Ruby Gems are pre-build so that developing the mobile application with a ruby on rails framework is efficient and easier than other web application framework.

OLMS is an online management system related application to educational institutions which is compatible with the web platform and Android operating system. The front end of the application was built in Eclipse IDE (integrated development environment) open frame work which supports different languages including Java, JavaScript, Python and C++. During the process of development of OLMS, languages and technologies such as Ruby on Rails, PostgreSQL, Apache Cordova, Heroku, and GitHub are used. Heroku is used to cloud the back-end code into web to make application multiuser application. Ruby on rails is a back-end to manage the data transactions connective with PostgreSQL database.

Native apps are coded using the Android SDKs and IOS. Mobile web apps are accessed using Internet browsers such as Safari on iOS and Chrome on Android. Projects such as Apache Cordova contributed to develop apps with the aid of web technologies that are compiled into a wrapper creating a hybrid app such as OLMS.

We use a class diagram to implement a set of classes, interfaces, and their relationships in the OLMS. . Class diagram is important not only for visualizing, specifying, and documenting structural models, but also for constructing executable systems through forward and reverse engineering. We mainly use it for detailed modelling translating the models into programming code of our app. Front page of OLMS is shown in figure 1 and its important features include:

User Management: Instructors and students can sign up, login, and receive a password if it is forgotten. Administrator is assigned with the user management authority where he can add/edit/update the user details, assign roles (can make a user as a co-administrator), and delete a user.

Subject Management: Administrator, professor, or authorized user can record live lectures directly in the classroom. Once the operation is complete, authorized user can upload the recoded content to the OLMS server. Registered users such as students can retrieve recorded materials saved through the 'Lectures' tab of the OLMS.

Notice Board: This option performs operations to enable users (administrators, instructors, and students) to create and post notices and discussions on the notice board. This feature also provides the information about the upcoming events that are going to be held in/out the institution. For instance, when there is a graduate project presentation, the authorized person to this tab can tag the notice instead of sending an email to all intended students. Once the post has been made, the notification will be sent to registered users related to a specific subject as a multicast communication. Not all registered students should receive the notification but only those interested in a certain course. Intended students can view this notice and give their comments.
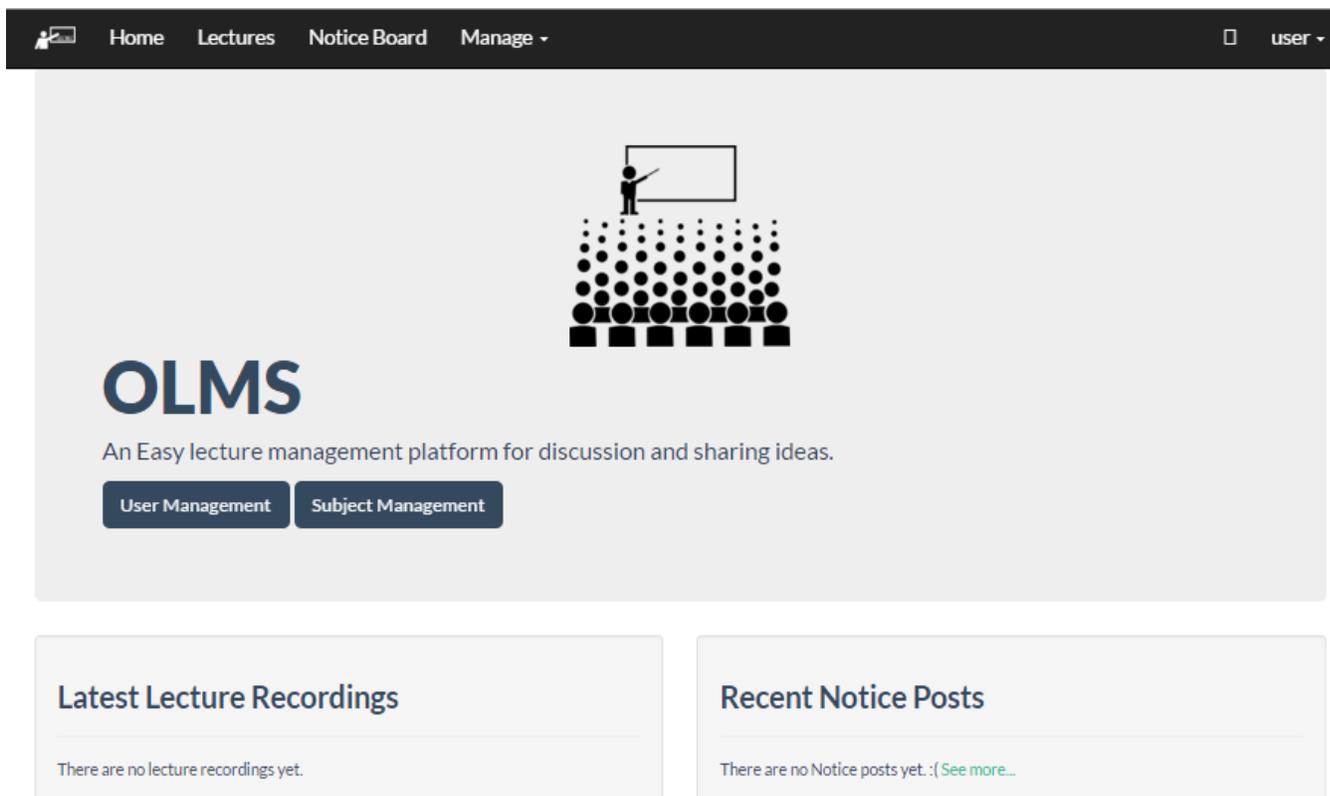
Figure 1.OLMS window

3.   Discussion and Analysis

Learning process is barely continued in some accidental situations during the semester for some students who get commitments to do such as attending a wedding, assisting parents and relatives, and going for an interview in another city. Moreover, cases may include ill students who should be home or hospitalized and can't attend regular classes. Additionally, some students won't be able to attend the regular classes because of the travel expenses. In such cases, obtaining the recorded lectures is an alternative solution for students to continue their study. Also recorded lectures can be employed in the virtual classroom and online learning to eliminate the cost needed for real classrooms enrolment. E-learning would be a compromised solution to catch up the covered material in the traditional classroom. In addition, having a platform for course content discussions and views with other students and the instructor would improve the course content understanding and enhance the course-load participation effectively. This paper typically describes what is needed by the system user as well as requested properties of inputs and outputs. Once the user is signed in with required credentials (login name and password), he will be redirected to the OLMS application home page with Lecturer/Administrator/Student authorities. The administrator and the lecturer are given Subject Management privilege while the User Management is controled by the administrator only.

One of the key features of the application is the lecture recording. There are several applications in the market for lecture capture such as VIDIZMO, YuJa, Epiphan, and Mediasite lecture capture. It should be noted that *these applications should be paired with a separate LMS application to have the recorded lectures available to students*. Moreover, some of these packages involve a hardware installation. As a result, *these packages are not stand-alone applications* and using them implies an increase of the cost to include having a freelance LMS together with the hardware components as well. So far no other LMS application has the feature to record a live lecture and publish it to the server at the same time within only one application. OLMS has two techniques implemented for lecture capture. These techniques are Webcam recording and Mobile recording. Once the lecture is recorded, there are two ways to be

published through uploading to the server. The pre-recorded lectures can be uploaded from a local storage or from cloud storage (for example Dropbox).

On the other hand, sharing ideas and views among users is a primary ingredient of some applications such as D2Learn as earlier pointed out. Based on that, we added a notice board allowing having a social media service available in our developed OLMS application. In the following sections, we focus on these functionalities.

A. Lecture recording techniques

OLMS has two features implemented to record a live lecture. These functionalities are recording a lecture using a web camera and using a mobile device such as smartphone and tablet.

1. Web camera recording

OLMS facilitates a live lecture to be recorded using a web camera as shown in figure 2. Authorized user can record a lecture through this feature in the application using a web camera. Once the lecture has been recorded, user will be given options to Save or Cancel the recording to complete the operation. If the user is satisfied with the quality of the recorded video and wants to accept, he can click save to store it. File would be saved to the local storage and hence to cloud database if necessary.
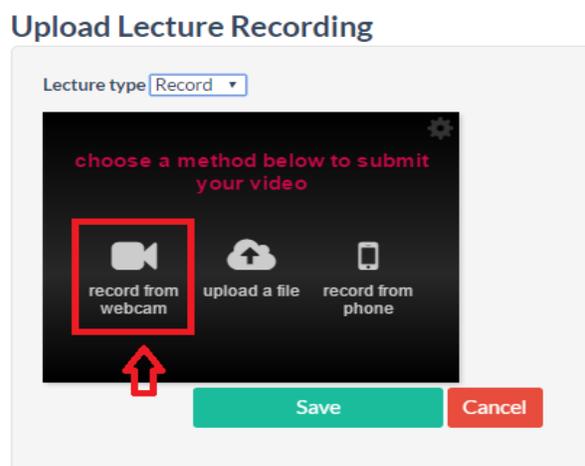
Figure 2. Record from Webcam

2. Mobile Recording

Sometimes a lecturer might be in a situation where the authorized user, likely to be a student, cannot record the live lecture using a webcam due to many reasons. These reasons may include that there is no computer with available camera or if the lecturer wants a higher resolution for his lecture presentation. In such cases, authorized user can use the camera of a mobile device (smartphone or tablet). CameraTagis an open add-on which is used to integrate this feature into our application. A user should select to record the lecture using a phone providing a valid mobile number and then press "Send Mobile Link" as shown in figure 3. Instructor/Administrator will send a domain link to the authorized user mobile device through SMS message. Once the link has been received, the user can start recording the lecture. Once the recording is finished, the video will be published directly to the OLMS server. Students registered for a particular course will receive an email notification regarding the latest uploaded information.
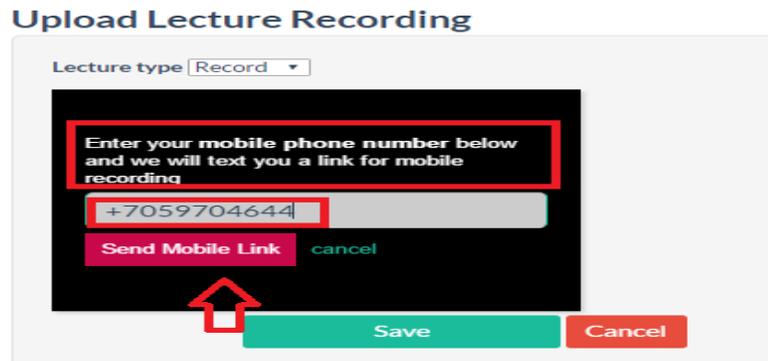
Figure 3. Mobile recording

B. Uploading recorded lecture

Once recording is complete, user will be given the option to upload the saved lecture file and this can be done by two methods as explained below. In our case, uploading a file means publishing the file to the application and allowing users to retrieve.

1. Upload from local storage

Uploading a lecture from local storage means that the lecture has already been recorded and saved in the local storage of the system. To upload a recorded lecture, authorized user has to navigate to the location where the file is saved as shown in figure 4. Once the file has been selected from the local storage, the file will be published to the application through clicking on the "Save" option. If the user does not want to upload the recording, operation can be simply canceled by tapping on the "Cancel" option.
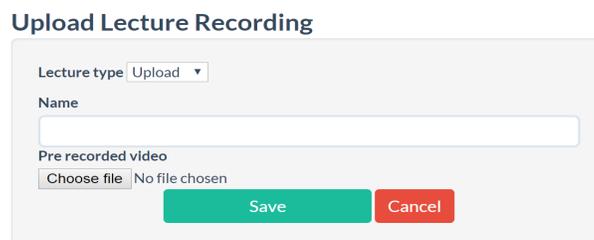

Figure 4. Upload lecture window

2. Upload from Cloud Storage

OLMS provides a feature to upload recorded lectures from cloud services (For example Amazon CloudDrive services). Once the file is selected from the cloud database, it will be published to the OLMS server by clicking save as shown in figure 2. Once the record is published, students registered for a particular subject will get email notification regarding the latest uploaded information.

C. Notice Board

The Notice board is where students, lecturers, and administrators can post notices for events in the department or university. For most categories you need administrator permission except for discussions. Categories like cultural events, sports, marketing commercial products are not allowed to post on the notice board by the users themselves. Research related notices, departmental, or university level notices can be posted with administrator permission.

In our OLMS, we create communities first in which a community is a category of similar posts. User can post a notice by creating a thread under related community. Thread in our application is an individual post under a certain category. Once the thread has been created, an automatic message will be generated to confirm the notice post and will be sent to users by email address. Administrator is the one who normally creates intended comminities. So that users including lecturers and students will be able to easily get to know where they can post their notice. In order to create a new communty, the administrator needs to enter information shown in figure 5.

Figure 5. Community Creation

Figure 6. List of Communities

Figure 6 displays a list of notices posted on the notice board page. Once the notice has been posted, a notification regarding the post is sent to all related users of the application. The administrator is able to edit and delete a post according to the standards. Under "community" section, users can create new threads to post notices of a related category. User can comment on the post through creating a thread as shown in figure 7.
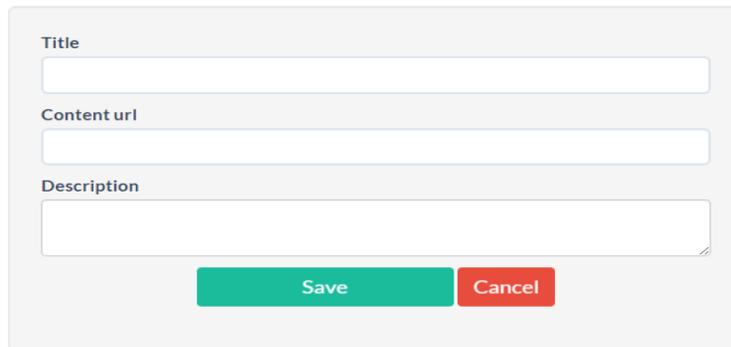
Figure 7. Thread creation for community

3. Conclusion

Education is no more restricted to traditional classrooms but it has reached far and wide. Virtual classroom that uses online learning plays an essential aspect in education and studying. This would reduce cost and provide flexibility in learning without a degradation of the learning process and its quality. Also synchronous learning and lectures accessibility are still factors that virtual classroom maintains. In addition, audio-visual representation of information makes the procedure of studying interesting and interactive for learners. OLMS is a mobile application that provides several features for E-learning and virtual classrooms. In this paper we introduce OLMS and present its features including mobile and webcam recordings made lectures recording reliable to instructors, and students. Notice board feature is a platform for social media discussions and chat. Users can publish a video hosted on public domain or cloud database. The profile management is effective and secure. Application is designed to be capable of tolerating large scale data like video recordings.

References

Borup, R. (2010). An Introduction to Ruby on Rails. In Proc. Southwest Fox Conf (pp. 50-55).

Chawdhry, A., Paullet, K., & Benjamin, D. (2011). Comparatively assessing the use of Blackboard versus Desire2Learn: student perceptions of the online tools. *Issues in Information Systems, 12*(2), 273-280.

Claar, C., Portolese, D. L., & Shields, R. (2014). STUDENT ACCEPTENCE OF LEARNING MANAGEMENT SYSTEMS: A STUDY ON DEMOGRAPHICS. *Issues in Information Systems, 15*(1).

Curtis, B., Krasner, H., & Iscoe, N. (1988). A field study of the software design process for large systems. *Communications of the ACM, 31*(11), 1268-1287.

Deacon, J. (2009). Model-view-controller (mvc) architecture. Online][Citado em: 10 de março de 2006.] http://www. jdl. co. uk/briefings/MVC. pdf .

DeNeui, D. L., & Dodge, T. L. (2006). Asynchronous learning networks and student outcomes: The utility of online learning components in hybrid courses. *Journal of Instructional Psychology, 33*(4).Blackboard Learning System, Blackboard Inc., March 2004.

Heirdsfield, A., Walker, S., Tambyah, M., & Beutel, D. (2011). Blackboard as an online learning environment: What do teacher education students and staff think?. *Australian Journal of Teacher Education (Online), 36*(7), 1.Mobiliya Edvelop, http://www.edveloplms.com/

Mbuva, J. M. (2015). Examining the Effectiveness of Online Educational Technological Tools for Teaching and Learning and the Challenges Ahead. *Journal of Higher Education Theory and Practice, 15*(2), 113.

Meurant, R. C. (2010). How computer-based internet-hosted learning management systems such as Moodle can help develop L2 digital literacy. *International journal of multimedia and ubiquitous engineering, 5*(2), 1-7.

Rumbaugh, J., Booch, G., & Jacobson, I. (1999). The unified modeling language user guide. Addison-Wesley.

Ippakayala, V.K. & El-Ocla, H. (2017). OLMS: Online learning management system for e-learning. *World Journal on Educational Technology: Current Issues. 9*(3), 130-138

Smart, K. A., & Meyer, K. A. (2005). Changing course management systems: Lessons learned. *Educause Quarterly, 28*(2), 68-70.

Wang, H., & Shao, M. (2012). Objective-Oriented Assessment in Desire2Learn for Quality Matters. *Cases on Assessment and Evaluation in Education*, 398.

Wargo, J. M. (2015). *Apache Cordova 4 Programming*. Pearson Education.

Declarations:

None of the authors have any competing interests.