THE DESIGN AND IMPLEMENTATION OF A LEARNING MANAGEMENT SYSTEM

Awodele Oludele, Onuiri E. Ernest, Agboola A. Ifetayo, Bot M. David and Chukwuemeka K. Chinazom

Department of Computer Science, Babcock University, Ilishan-Remo, Ogun State; Nigeria
dealalways@yahoo.com, ernestonui@gmail.com, oyat2010@yahoo.com, mafeng.bot@gmail.com & chukwuemekachinazom@gmail.com

KEYWORDS

Real Time-Access, Learning Management System (LMS), Communication, Content Management, Students, Learning Environment, Education

ABSTRACT

Learning is the act of acquiring knowledge or skill through study. In order to survive in this ever evolving universe, one is expected to acquire various survival skills and thus improve himself to become a better individual. There is the need for every child to be educated. It has been realized that learning can be more fun and of greater value outside the traditional four walls of a classroom. With software such as the Learning Management System (LMS), students can now acquire knowledge in the comfort of their rooms, places of residence and other locations of convenience. Major problems encountered by students involve inaccessibility to course materials and the lecturer’s. Our main objective is to provide an easier platform for student-lecturer relationship and easy dissemination of information by the creation of LMS software. The Agile software methodology was employed in the development process. This is because it provides visible results for quick iteration and advancement of the system. We discovered that creating an LMS can be quite challenging because it requires time to transform our ideas into reality with effective use of proper tools for the software development. It is indeed an excellent method of impacting knowledge and should be implemented in all Nigerian institutions of higher learning.
Introduction

The twenty-first century is characterized with people living on the fast lane. The daily struggle is to get much work done with little hassles. This can be seen in the manufacturing industry where on a yearly basis (or shorter time frame) new and improved automobiles and mobile phones are created. What is termed “new” becomes obsolete. The education sector hasn’t been left out. There is the constant need to keep up with varied forms of learning best suited to different categories of students that pass through the education system. LMS has its roots in the term, Integrated Learning System (ILS) which offers functionality beyond instructional content such as management and tracking, personalized instruction and integration across the system [1]. The term ILS was thought up by Jostens Learning, and LMS was originally used to describe the management component system of the PLATO K-12 learning system. It was content-free and separate from courseware [2]. LMS is a general term that is used for a wide range of systems that organize and provide access to online learning services for students, teachers, and administrators. These services usually include access control, provision of learning content, communication tools and organizations of user groups [3]. In addition, observation has shown that students do not have access to course materials. They prefer to search online than spend countless hours at the library searching for related books. Students also experience missing grades and assignments due to absence of records for those who submitted.

Lastly, students lack accessibility to lecturers and also have limited time for completion of coursework. This is due to the bulky course load and limited time to consult lecturers. Consequently, this research provides real-time access to lecturers/students and course materials through the use of online chats and forums. It also provides easy dissemination of information to students via announcements and posts online as well as ensure prompt submission of projects and assignments for effective grading and assessment. Our scope places emphasis on three end-users which are the students, lecturers and the administrator of the system. The students will have access to course materials which can be downloaded, viewed or read online. The lecturers are responsible for course management. This involves uploading course materials for students and grading for assignments or projects. The administrator manages the content of the whole system. He can add or remove courses, lecturers or students. He has access to the database.

Review of Closely Related Works

Over the years, the rapid growth of the internet empowered institutions to pioneer the development of web-based e-learning content libraries. These included training in project management, team development and leadership. Once companies became aware of the potential for technology-based learning delivered via the internet, the market place welcomed a new class of systems: LMSs. LMSs enable institutions to plan and follow up the learning needs and accomplishment of users. Examples of well-known LMSs are Moodle, Sakai and Blackboard.

Blackboard is a proprietary software developed in 1997 by the co-founders of Blackboard Inc., Michael Chasen and Matthew Pittinsky. This web-based server software features customizable open architecture, course management and scalable design that allow integration with student information systems and
authentication protocols. It has quite a number of features such as announcements, discussions, chats, mail, course content, calendar, learning modules, assessment, assignment, grade books and media library [4].

Its major benefits are increased availability, quick feedback, tracking, improved communication and skilled building. Some of its weaknesses are the fact that it is costly, hard to learn and it limits creativity because it is a licensed software [5]. The login page is shown below:

![Blackboard login page](image)

Figure 1: Blackboard login page [5].

Another LMS taken into consideration is Sakai. It was funded by a Mellon Foundation grant, Sakai was built by a consortium of five large U.S universities, including Michigan, Stanford, MIT, UC Berkeley, and Indiana institution. Sakai was released to the public in 2005 and is managed today by the Sakai foundation, which oversees its development and project roadmap. The application is programmed in Java and designed to be a service-oriented application suite. Some features include document distribution, a grade book, discussion, live chat, assignment uploads and online testing. In addition to the course management features, Sakai is intended as a collaborative tool for research and group projects [6].

Conversely, Sakai’s critics point out that it lacks comprehensive competency profiling and management, which makes it unsuitable for large enterprise environments. It can also be challenging to integrate Sakai with other enterprise software systems, such as talent management, HR software suites and ERP solutions. Sakai’s greater complexity also makes it more challenging to install and set up than other LMSs like Moodle. This makes it less suitable for simple, rapid deployment projects that require an LMS [7].

Moodle, on the other hand, is an open source software that stands for Modular Object Oriented Dynamic Learning Environment. It is one of the most widely used LMSs with over 43 million registered users. It lets teachers, trainers and administrators manage online learning and training. Some of its features include assignment submission, discussion forums, files download, grading, instant messaging, online calendar, online news, announcements (College and course level) and online quizzes [6]. The figure below shows what a Moodle log in page look like:
There are five basic modules that support assessment and grade evaluation in Moodle. These are the Assignment, Quiz, Survey, Workshop and Choice Modules. Other Modules that indirectly support assessment are the Forum and the Journal Modules.

In the assignment module, lecturers can set various assignments and due dates. The students are then permitted to upload their assignments in specified formats (.pdf, .txt, .ppt, etc.) Late submissions are not
allowed. Here’s what the assignment editing page looks like:

Figure 3: Moodle’s assignment editing page [8]

In the quiz module, lecturers could choose to set time limits and offer variety such as multichoice questions, short-answers, true/false questions and many more. The quizzes could have functionalities such as immediate grading and multiple attempts on questions. Below illustrates the quiz editing screen in Moodle:

Figure 4: Moodle’s quiz editing page [8]
Major benefits of Moodle are that it is easy to run, administer and maintain due to the fact that it is free. Despite these amazing benefits, some users are that Moodle relies on third-party add-ons to create functionality rather than including it as part of the core product. Though Moodle is free to acquire and use, administration and maintenance costs are often more significant than purchase [9].

Generally, learning management systems incorporate the use of content management and virtual learning environments. Hence, information can be disseminated to multiple users at the same time via the internet. A major limitation is that most LMSs do not consider the individual differences of learners. Recently, increasing attention is paid to characteristics such as learning styles, their impact on learning and how these individual characteristics can be supported by learning systems [10]. In relation to our own software, we would incorporate the core features of Sakai, Blackboard and Moodle and optimize them for use in Babcock University which is our case study.

**Methodology**

The agile software development methodology was employed in the development process. This is because it allows for visible results quickly and for quick iteration and advancement of the system [11] [12].

Features of the proposed system are the home page, the administrator login page, lecturer login page, contact page, help page, register student page, register instructor page, update instructor/student page, view instructor page and view student page; also security, delivery, interaction, reporting and record keeping will be put into place. The system will require the user to login to have access to the website to do the following tasks: view and download course materials, upload files, projects and other materials for people to see and also interact through live chats and discussion forums. Also the LMS will allow instructors to upload videos that will be viewed and downloaded. All users will not be allowed to login simultaneously on multiple systems.

The hardware and software requirements are that there must be good internet connection, minimum RAM of 512MB and 1.5GHz processor speed for any user to have access to the LMS. Also, it will work with any operating system with an installed web browser but the web browser must meet the following specifications, Java runtime environment 1.4, Java script pop-ups and cookies set to enable.

The system will be designed and structured to cut down the level of disjointedness and ease the learning process. Most of the time, there is break in the flow of study and information gathering because the student has to leave his/her current study location to look for the lecturer or get to the library to retrieve extra needed information.

A mailing and real-time support will make it possible for a student to contact a lecturer or other students instantaneously from his/her very location, thus keeping the study session active and continuous. This will make the assimilation level higher since there are no breaks in communication. Course materials will also be made available for easy accessibility.
The system design will be structured with a mass communication section. This is to ensure easy dissemination of information to the entire student body. Also, upcoming events can be posted on the LMS to effectively notify the university community of scheduled events. Online forums would also be made available for academic discussions.

This LMS is indeed very feasible as it provides a larger avenue for learning in addition to rapid delivery of learning content on a scalable web-based platform.

Once tested and implemented, the LMS would be updated and maintained at regular intervals.

**Design Tools**

1. Use Case Diagram: A use case is described in terms of a sequence of interactions between some actors and the system by which the system provide a service to the actors. Each use case then captures a piece of functional requirements for some users. All the use cases together describe the overall functional requirements of the system. The first step in requirements capture is to capture requirements as use cases. We have three use case diagrams for the lecturer, administrator and the student.

![Lecturer USE Case diagram](image)

**Figure 5 Lecturer USE Case diagram**
Figure 6: Student USE Case Diagram
Figure 7: Admin USE Case Diagram
Figure 8: Entity relationship diagram for the LMS
Figure 8 displays the relationships between each entity in the LMS database. For example, the lecturers and the students have a many to many relationship as many students have access to many lecturers.

**Hardware and Software Requirements**

This entails the hardware and software required to run on this application (LMS). The LMS uses server resources in an efficient and intelligent manner it can support thousands of users. The LMS requires:

**Hardware Requirements**

i. An intranet connection  
ii. Processor 2.5 GHz  
iii. 1014 MB RAM

**Software Requirements:**

i. Database Server  
iii. SQL Server 2008, SQL Server 2005 or SQL Server 2000 - you can also use SQL Server Express  
iv. Client and Browser  
v. Operating system of any kind (Microsoft Windows, Mac OS, Linux)  
vi. Web browser (Safari, Firefox, Google Chrome, Internet Explorer, and Opera) must include Java Runtime Environment 1.4 or higher and also JavaScript Pop-ups are enabled Cookies are enabled, Java is installed, up-to-date and AJAX enabled; lastly Pop-Up blocking disabled.

**System Implementation and Testing**

In the course of building this system information concerning the database creation and entity relationships involved was gathered. In addition, cues were taken from the creators of Moodle and Blackboard learning management systems. Programming tools in use were HTML (Hypertext Markup Language), PHP (Hypertext Preprocessor) and JavaScript.

In order to identify potential defects and malfunctions, the software was subjected to various forms of testing:

1. **Database testing:** Our LMS database is made up of 16 tables and each table contains the name of the fields, data types, sizes and other constraints that define the table. Below are a few screenshots of each table along with a brief explanation:
Figure 9: Admin Table

Figure 9 shows the admin table and this contains the login details of all those authorized to access the admin section of the website. The table keeps track of the admin_id, username and password of each system administrator.

Figure 10: Assignment table

Figure 10 is the assignment table and this records the assignmentId, assignmentCode and lecturerId. The table keeps each assignment unique and provides a link to the lecturer table, identifying the lecturer that posts an assignment.
Figure 11: AssignmentTable Table

Figure 11 is the assignmentTable table and it keeps track of the solution files that students upload for assignments given based on the assignment code.

2. System Testing: System testing involves integrating one or more components system functions or features and then testing the consequent integrated system. It is in two major phases:
   
a. Integration testing: This involves building a system from its components and testing the resultant system for problems that arise from components interaction. Integration testing is mostly concerned with finding defects in the entire system. We shall consider the administrator page for this test. The administrator is responsible for:
   
i. Adding course code and course title.
   
ii. Adding levels
   
iii. Editing student and lecturer pages
Figure 12: The Administrator Option for Adding Course Codes and Titles.

Figure 12 above reveals the interface for adding course codes and titles. The administrator enters new courses and assigns course codes to them as well. This newly added course is then stored in the LMS database.

- LOGOUT
- ADD SUBJECT
- ADD LEVEL
- ADD ADMIN
- ADD COURSE OF STUDY
- VIEW STUDENTS
- VIEW LECTURER

Learning management system Techniques, Tools and Strategies

Course added Successfully

Figure 13: A course has been added successfully

Figure 13 shows that once the course addition process has been completed without any errors, the system sends a notification that the new course has been added successfully.
Learning management system Techniques, Tools and Strategies

Figure 14: An Error Message The Course Already Exists.

As seen in Figure 14, the system sends an error message because the stated course already exists in the database. Thus, duplication has been checked.

Once the test has been carried out on this page after integration and it passes all tests, then we would proceed to the next testing phase. Testing on these modules would verify that only registered students and staffs would be able to sign in and out later on. Another test is to ensure the functional and non-functional requirements for these pages have been met successfully.

b. Release Testing: This involves testing a version of the system that would be released to users.

Risk Identification and Management
A risk is the possibility of an undesired event occurring. As illustrated in Table 1, risk mitigation, monitoring and management (RMMM) refers to the solution put to correct the risk at hand. A good RMMM should try to avoid the risk, monitor and also manage it.
Table 1: Risk Mitigation, Monitoring & Management.

<table>
<thead>
<tr>
<th>Risk Component</th>
<th>Impact Description</th>
<th>Risk Category</th>
<th>Probability</th>
<th>RMMM</th>
</tr>
</thead>
<tbody>
<tr>
<td>The technology to be used for the project is new to most members of the group.</td>
<td>Schedule risk marginal impact</td>
<td>Project risk</td>
<td>60%</td>
<td>Project members were put under tight schedule to learn and implement the technology after class hours.</td>
</tr>
<tr>
<td>Deadline for project submission will be tightened.</td>
<td>Performance risk and Schedule risk critical impact</td>
<td>Project risk and Technical risk</td>
<td>40%</td>
<td>More attention would be put towards building the application.</td>
</tr>
</tbody>
</table>

Discussion of Results

During the course of this research it was realized that building an LMS is a rigorous task. Consequently, a system that will provide real-time access to lecturers/students and course materials, announcement of information and dates relating to the learning process across to a larger percentage of the student population was developed.

Furthermore, this research project advocates the importance of simple and yet attractive design. The design allows users to easily navigate the site and carry out tasks with minimal hindrances. An attractive layout also ensures that the user does not get bored while using the system.

In addition, the quiz component of the LMS provides a faster way of handling large classes. It leaves no room for mistakes with respect to students not writing down the appropriate question or missing something the lecturer said. The system also eradicates the commotion that arises when it is time for submission of quizzes taken.

Recommendations

The LMS meets a lot of expectations but would perform better if the following recommendations and suggestions are considered:

i. System testing and maintenance should be performed regularly to avoid sudden system failures.
ii. Updates and other modifications should be introduced with prior notice to users.
iii. A course administrator should be employed that can manage the interactions between lecturers, students and administrators on one side and technicians on the other. This will aid ensure that proper support is provided for the system.
References


