Abstract

Learning management systems (LMSs) are software applications that comprise a suite of tools for learning and online teaching. There are many commercial and open source LMSs that can be found on the web. Because there are many LMS systems in the market place, one of the problems facing a user is how to choose a system that can meet the requirements. This paper is about a computer program developed at the Near East University for the evaluation of learning management systems. The developed system, named EW-LMS, is web-based and can easily be used over the internet anywhere and at any time. The system provides a web-based decision support system that may help administrators and instructors to choose the most suitable LMS system for their needs and requirements. This evaluation system was designed using the MS-Visual Studio .NET together with a MS-SQL Server based database.

Keywords: Evaluation of LMS; LMS; computer aided LMS; choosing an LMS.

1. Introduction

In this day and age, we can see clearly rapid changes and developments in technology. Recent advances in the internet provide a common virtual space for students and instructors who are physically separated; internet is widely acknowledged as an essential component of teaching and learning environments in online settings (Liu, 2005). Recently, due the rapid increase in the popularity of the internet the delivery of learning programs have gradually shifted from local desktop to online-based applications. As more and more technological tools become available for online education, there is an increasing interest among educators and other professionals in the application of these tools in online courses (Hanna, 2003; Moore, 2003). Virtual learning environments represent an entirely new form of educational technology. A virtual learning environment (VLE) is a set of teaching and learning tools designed to enhance students’ learning experiences by including computers and internet in the learning process. This environment is online, and is dependent on the technology of the internet in order to be able to exist.

In large-scale operations, online learning management systems (or LMSs as they are commonly known) can save costs and time. LMSs are software applications that comprise a suite of tools for learning and teaching online (Naidu, 2006). Some of the widely known LMSs are: WebCT, Blackboard, Moodle, ATutor and Claroline. In conventional educational settings, online-learning management systems can help to improve the speed and effectiveness of the educational processes, communication among learners, and also staff and students. One of the biggest problems facing an administrator or instructor is how to choose an LMS system that can meet the needs rose. This paper describes a software system developed by the authors to aid in the selection of a suitable LMS system. The software is named Easy Way to Evaluate LMSs (EW-LMS). This is a web-based software that can easily be used over the internet anywhere and at any time. The software provides a web-based decision support system that may help users to choose the most suitable LMS system for their needs and requirements.

2. The EW-LMS System Configuration

The potential objects for the developed system are: Administrating the system, product information, evaluating sub-system, comparison between products, opinions and discussion. The Context DFD diagram of the system is shown in Figure 1. The system has two types of users: Administrator, and General User. Each user has different privileges and can do different tasks depending on their privilege levels.
2.1. The Administrator

The administration team is responsible to administrate and manage the whole system tasks and processes. Here, one can see how an administrator can affect the system in the cases stated below:

- Administering all the system’s tasks,
- Adding new administration team members,
- Updating any administration team members’ information (their real names or accounts’ passwords),
- Deleting an administration team member,
- Inserting a new LMS to the system,
- Updating LMSs’ information,
- Deleting a LMS from the database and all its related information,
- Describing list of features of each new LMS,
- Updating LMS’s information and/or their features’ description,
- Ranking each feature of each LMS,
- Editing the list of all features and their definitions,
- Managing the opinions given by the users,
- Giving opinions about an LMS,
- Managing the discussions given by the users,
- Discussing a feature for an LMS.

![Diagram](image-url)

**Figure 1:** The Context DFD diagram of the system

2.2. The General Users

General users of the system can get benefits from the system through evaluating and making comparisons among the LMSs offered in the program. A user is offered the following choices:

- Viewing all products (or LMSs).
- Making the evaluation process:
  a. Choosing the group of features depending on needs,
  b. Seeing all LMSs that match your group of features,
  c. Viewing the administrator ranks of each feature relating to each LMS,
  d. Viewing a full description of each feature relating to product,
  e. Weighing each feature depending on your needs,
  f. Ranking each feature of each LMS,
  g. Getting the evaluation result,
  h. Viewing the previous evaluation results.
- Comparing between products:
  a. Full comparison of all features of all products,
  b. Comparing a group of products with same features.
• Giving an opinion about an LMS and reviewing the previous opinions given by the previous users.
• Discussing a feature of an LMS and reviewing the discussions of previous users.
• Viewing the definition of each feature.
• Viewing the feature’s description of each feature relating to each product.

3. Design of The System

4. The Programming Language Used

The system was based on Microsoft Visual Studio .NET 2005 (Visual Basic .NET, ADO .NET, ASP .NET), Hyper Text Markup Language (HTML), Dynamic HTML, XML, and VB Script.

4.1. Application programs

The following application programs were used: Rational Rose Enterprise Edition, Adobe PhotoShop, Microsoft FrontPage 2003, Microsoft PowerPoint 2003, and Microsoft Visio 2003.

4.2. Database applications

The system contains a database to store all LMSs’ information and the data related to the evaluation and comparison operations. Microsoft SQL Server 2005 is used to build this database.

4.3. The evaluation process

The system demands using artificial intelligence methods and decision making procedures to provide a smart process to help users in making their decision. This kind of artificial intelligence algorithm depends on fuzzy logic comparisons and structural conditions to find a system from a group of choices as a suitable choice for the user.

The features of LMSs are criteria that may enable users to evaluate the systems. Fifty-two comprehensive features were used in the system for the following popular LMSs: Moodle, ATutor, Blackboard, WebCT, and Claroline. The user can select the group of features from the list. The system can then find the best LMS to match the selection criteria. The database contains a full description of the most popular LMSs widely-used over the world. The steps in a typical application are (see Figure2):

Step 1: Selecting user needs:
This step will present the 52 features of the LMSs stored in the system and ask user to choose the group required. User clicks on a feature’s name to view the feature’s definition. Up to 52 features can be selected at any time by the user.

Step 2: Choosing the group of products user want to include in the evaluation process:
This system checks the list of all products and can recognise the one that matches user’s needs and user group of features selected in Step 1. User can choose all the products appeared or some for them, but not less that one.

Step 3: Weighing user group of features:
Here, the user is required to weigh each feature from the group of features that has already been chosen in Step 1. User’s selection depends largely on the required amount of each selected feature. The weights are between 0 and 1.

Step 4: Ranking each feature for each product:
In this step, the user has to be careful about the ranks. The feature names should be clearly understood. Feature definitions can easily be seen by clicking on a selected feature. The user can see the administrator rank. The administrator can rank the features for each product depending on the level of experience. Also, the user can see the opinions given to each product.

Step 5: The results:
Here, the user can see all the results needed from the evaluation operation already made. A group of results such as the following can be seen:
1. The group of products selected their versions, and the grade of each one. Also, user can see the details of each product.
2. The product name, version, and the grade of the best product. This result depends on the weights and ranks user has given in Steps 3 and 4.
3. The product name, version, and the grade of the product that takes the best grade depending on the administrator’s given ranks. This result presents the administrator advice depending on his/her high level of experience.
4. At the end, there is a brief summary about the previous evaluation operations performed by other users with the same group of needs or features. The products shown here were chosen as the best grade after applying the evaluation depending on the ranks given from the users. The result presents the product name, its version, and the date of the last operation.
At this point, a user can
a. give new weights and ranks to the products so that the overall process can be repeated from Step 3 above.
b. start a complete new operation from Step 1.

5. Example Run of the Program

As an example, let us assume that an instructor wishes to use an LMS system for his distant teaching courses. After a brief search on the internet the instructor may find a large selection of LMS systems both open-source and commercially available. The instructor will now need to make a critical selection among the LMSs found on the internet. This is where the system developed by the authors becomes very useful as it will aid the instructor to evaluate the LMSs and then make the best possible choice based on his requirements.

An example is given here to illustrate the operation of the software. In this example a choice between four LMSs is made:

• The user chooses four features from the available 52. (F1=View course activities, F2=Statistics, F3=Online quiz editor, F4=Search within courses)
• The system will retrieve the LMSs that match the group of features chosen by the user. The user can choose all or some of them to be included in the evaluation operation. (LMS1=WebCT 4.0, LMS2=Moodle 1.9, LMS3=Blackboard 6.0)
• The user should weigh each feature existed in the group of features which were chosen in step 1. (F1 weight = 0.5, F2 weight = 1, F3 weight = 0.2, F4 weight = 0.8)
• Ranking each feature to each LMS as shown in Table 1.

Table 1: Ranking the features of selected LMSs

<table>
<thead>
<tr>
<th>Weights</th>
<th>W1 = 0.5</th>
<th>W2 = 1</th>
<th>W3 = 0.2</th>
<th>W4 = 0.8</th>
</tr>
</thead>
<tbody>
<tr>
<td>LMS1</td>
<td>0.6</td>
<td>0.4</td>
<td>1</td>
<td>0.3</td>
</tr>
<tr>
<td>LMS2</td>
<td>0.9</td>
<td>0.5</td>
<td>0.2</td>
<td>0.8</td>
</tr>
<tr>
<td>LMS3</td>
<td>0.4</td>
<td>0.5</td>
<td>1</td>
<td>0.2</td>
</tr>
</tbody>
</table>

• Computing and getting results is as follows:
  The system then multiplies the weights set by features sets, each one independently for the selected LMSs, as shown in Table 2.

Table 2: Multiplying the weight sets by features

<table>
<thead>
<tr>
<th>Weight for F1 = 0.5</th>
<th>Weight for F2 =1</th>
<th>Weight for F3 =0.2</th>
<th>Weight for F4 = 0.8</th>
</tr>
</thead>
<tbody>
<tr>
<td>LMS1 0.5 x 0.6 = 0.30</td>
<td>1 x 0.4 = 0.40</td>
<td>0.2 x 1.0 = 0.20</td>
<td>0.8 x 0.3 = 0.24</td>
</tr>
<tr>
<td>LMS2 0.5 x 0.9 = 0.45</td>
<td>1 x 0.5 = 0.50</td>
<td>0.2 x 0.2 = 0.04</td>
<td>0.8 x 0.8 = 0.64</td>
</tr>
<tr>
<td>LMS3 0.5 x 0.4 = 0.20</td>
<td>1 x 0.5 = 0.50</td>
<td>0.2 x 1.0 = 0.20</td>
<td>0.8 x 0.2 = 0.16</td>
</tr>
</tbody>
</table>
Then the system divides each summation result by the total number of features \((n)\); in this example it is equals to 4 and,

\[
\begin{align*}
\text{LMS1} & \rightarrow \frac{1.14}{4} = 0.285 \\
\text{LMS2} & \rightarrow \frac{1.63}{4} = 0.408 \\
\text{LMS3} & \rightarrow \frac{1.06}{4} = 0.265
\end{align*}
\]

As a result, WebCT 4.0 takes 29% grade, Moodle 1.9 takes 41% grade, and Blackboard 6.0 takes 27% grade. It can be concluded that for this application the best one is the second LMS, Moodle with a grade of 41%.

6. Conclusion

The system demands using artificial intelligence methods and decision making procedures to provide a smart process to help users in making their decision. This kind of artificial intelligence algorithm depends on fuzzy logic comparisons and structural conditions to find a system from a group of choices as a suitable choice for the user.

The paper focused on the LMSs evaluation and assessment and describes how the web-based system can help and support a user wanting to make a decision for choosing an LMS.

The web-based system described in the paper could be considered as a practical and useful solution to the problem of LMS selection. The software makes the complex selection process a relatively simple task that can be carried out even by students. It is also possible to compare LMSs based on their features’ descriptions, giving the opinion or the viewpoint about any LMS or even discussing any feature of any LMS offered in the system, regardless of the level of users’ experience.

References


Moodle. Modular Object Oriented Dynamic Learning Environment. www.moodle.org

