Software Architecture and New Functions in Learning Management System "NOBASU"

Nobuo Funabiki, Toru Nakanishi, Noriki Amano, Hiroki Kawano, Kanako Uemura *

Abstract — **Recently, the *learning management systems (LMS)* using the Web technology has been developed and used at a lot of educational institutions. We have continuously developed an original LMS called *NOBASU (NetwOrk-Based Assistant System for University education)* to support various educational activities in our department to improve the quality while reducing costs. In this paper, we describe the software architecture and the outline of distinguishing functions in NOBASU. In addition to conventional functions in LMS, NOBASU has incorporated several original ones such as the *keyword submission function* for preparing and reviewing classes, the *lecture-scene playback function* using the digital picture-card show, and the *writing style check function* for report submissions. We have implemented them in NOBASU and evaluated the effectiveness through applications to real classes in our department.**

Keywords: learning management system, NOBASU, keyword submission, lecture-scene playback, writing style check

1 Introduction

Recently, due to the advancement of the information and communication technology (ICT), a variety of network applications and services using the Web technology have been developed and utilized at many groups, societies, and organizations. In educational institutions such as universities, the *learning management system (LMS)* has become extensively used to improve educational services while reducing their costs [1]-[5]. The LMS supports various learning activities of teachers and students in classes such as deliveries of course materials, submissions of reports, transmissions of mails, and managements of grades through the Web technology. By using electric mails and bulletin boards, the LMS can help two-way communications between teachers and students in classes, whereas one-way communications become common in many classes.

Independently, we have continuously developed an original LMS called *NOBASU (NetwOrk-Based Assistant System for University education)* to support various educational activities of teachers and students in our department to improve services while reducing costs and loads. As the independent LMS, NOBASU has implemented several original functions in addition to conventional ones in LMS. The *keyword submission function* requests students to submit three important words in each class to encourage them to prepare and review it. The *lecture-scene playback function* allows students to retrieve the necessary content of each class on demand, where a fine digital photo and the lecturer voice corresponding to one slide page explanation can be downloaded at the Web browser as the digital picture-card show [6]. The *writing style check function* checks the writing style of reports from students, and provides the results within a few seconds, which encourages students to keep revising their reports until they have no errors in the writing style.

As an LMS under continuous development at our laboratory, NOBASU adopts the four-layer software architecture, so that the code modifications including bug fixings and function additions can be easily realized among a number of inexperienced students. Each layer is related with the selection level at the service request. The first layer is related with the user role selection, the second layer is with the service object selection, the third layer is with the service function selection, and the fourth layer is with the service function supply. The codes for the first-third layers are written by JSP. The codes for the fourth layer are by Java where the design pattern is used to improve the reusability and maintenance.

The rest of this paper is organized as follows: Section 2 describes the four-layer software architecture of NOBASU. Sections 3-5 present the keyword submission function, the lecture-scene play back function, and the writing style check function as the original functions in NOBASU respectively. 6 concludes this paper with some future works.

2 Four-layer Software Architecture in NOBASU

As shown in Figure 1, NOBASU adopts *Linux* for the operating system, *Apache* for the Web server, *Tomcat* for the application server, and *MySQL* for the database system. The program codes for NOBASU are written by *JSP/Servlet*. In this section, we describe the software

*Dept. of Communication Network Engineering, Okayama University, 3-1-1 Tsushimanaka, Okayama, 700-8530, Japan Email: funabiki@cne.okayama-u.ac.jp

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

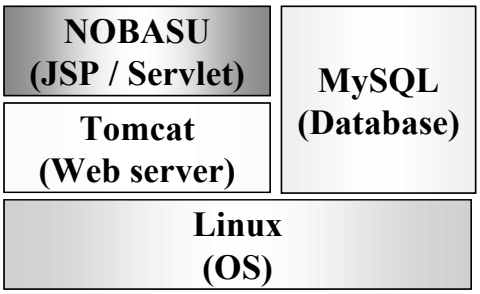

Figure 1: Software platform in NOBASU.

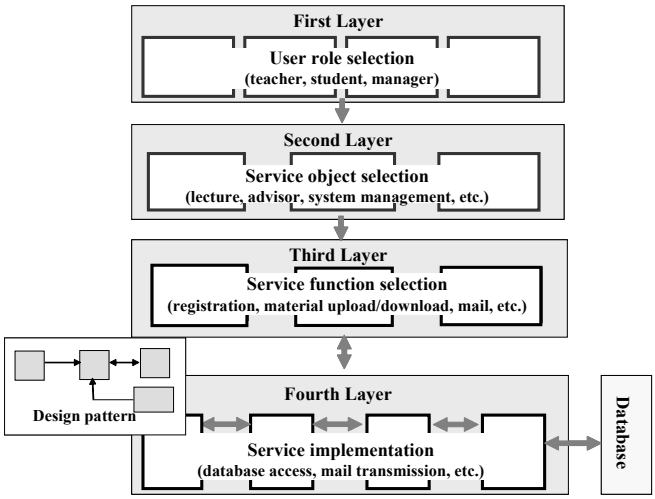

Figure 2: Overview of software modules in four-layer architecture.

architecture in NOBASU.

2.1 Four-layer Approach

As an LMS that has been continuously developed by many students at our laboratory, the modified modules and their source codes in NOBASU should be limited into as small areas as possible when new functions are introduced. For this purpose, we adopt the four-layer software architecture by considering the selection process for a service request. Actually, when a user accesses to NOBASU, it needs to select the user role, the service object, and the service function for the service request. Thus, we assign the user role section to the first layer, the service object selection to the second layer, and the service function selection to the third layer, where the codes are written by JSP. Besides, the service function implementation is assigned to the fourth layer, whose codes are by Java. Figure 2 illustrates the overview of software modules in the four-layer architecture.

2.2 First Layer

The first layer defines the selection of the user role with the user interfaces composed of the top page and the authentication form. By authenticating the user using the standard function in JSP, it determines the role of the user, and controls the contents that the user can access to. NOBASU has the following three different roles for users:

- **system manager**: can access to any content in addition to the system management functions in NOBASU.
- **teacher**: can access to any content in the service functions in NOBASU.
- **student**: can access to limited contents in the service function permitted to students.

2.3 Second Layer

The second layer defines the selection of the service object with the corresponding user interface. The service objects in the current NOBASU include the lecture subject, the programming exercise, the academic advisor assistance, and the system management. By the user role selected in the first layer, the candidates of service objects to be selected may be different. Thus, the code for this layer generates the menu page of showing the list of service objects and accepting the selection.

2.4 Third Layer

The third layer defines the selection of the service function. The service functions include the student registration for each lecture, the lecture material upload/download, the mail transmission, the report registration/submission/access, and the grade upload/access. Here, we note that even for the same service function, the contents and behaviors allowed to the user may be different by the role.

2.5 Fourth Layer

The fourth layer defines the implementations of the service functions. By receiving the request from the third layer, the module in this layer implements the procedure of the service function. The modules here include the database access module. Thus, there is no user interface in this layer. Unlike the previous three layers written by JSP, the codes in this layer are by Java programs for JavaBeans. For the better reusability and the easier maintenance, *Facade pattern* is adopted as the Java code design pattern, where all the methods calling the related classes are implemented in one class.

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

3 Keyword Submission Function

The keyword submission function requests the students to submit three keywords on the content of each class before and after it, by referencing the lecture material. This function intends to encourage students to prepare and review the class.

3.1 Purpose

To improve the interest and the understanding of a class for students, they should know the points by reading the material and/or the textbook before the class, and to describe the definitions of keywords with sentences. In this function, students are asked to submit three keywords before the class for the preparation, and to submit three keywords with their definitions in sentences after the class for the review. They are disclosed to all the students after the deadline so that they can refer them for self-study.

3.2 Service Functions to Teacher

The following four functions are provided to teachers:

- **class schedule registration**: a teacher can input the class schedule with times, month, day, hour, and comment. This schedule is used to select the submission deadline for each class.
- **submission result view by class**: a teacher can view all the keywords and the definitions submitted for each class from all the students with their names and student IDs.
- **submission result view by student**: a teacher can view all the keywords and the definitions submitted for all the classes from each student with the name and student ID.
- **submission result view by keyword**: a teacher can view the names of all the students and their definitions for each submitted keyword.

3.3 Service Functions to Student

The following three functions are provided to students:

- **keyword submission**: a student can submit three keywords discussed in the next class before the class for the preparation, and three keywords with their definitions in the last class before the next class. Beside, a student can write a comment about the class.
- **submission result display by class**: the same function for a teacher is provided. We note that the view of student names and IDs will help students to know who understands the class well.
- **submission result display by keyword**: the same function for a teacher is provided.

3.4 Evaluation

We applied the keyword submission function to one lecture in our department, and took the questionnaire from 42 students. To the question“ Did you increase the studying time because of this function ? ”, 18 students increased the time by 50% or more for preparations and 20 students did it for reviews. On the other hand, 14 students replied no change for preparations and 4 for reviews, where we suspect that these students did not use the keyword submission function. To the question “ Are keywords of other students useful ? ”, 18 students replied positively, whereas 7 students did negatively. To the question “ Is the function effective to understand the lecture ? ”, 25 students replied positively, whereas 11 students did negatively. Thus, we confirm the effectiveness of the keyword submission function in encouraging students to prepare and review classes, although additional schemes are necessary for further improvements.

4 Lecture-scene Playback Function

The lecture-scene playback function allows students to access to contents in classes that are edited as the digital picture-card show as the distance learning. One page of the picture-card show consists of a still image of the lecture scene and the lecturer voice corresponding to the teaching interval for one slide.

4.1 Overview

Figure 3 illustrates the overview of the lecture-scene playback function. Before starting the lecture, the lecturer needs to set up a high-resolution digital photo camera using a tripod, a mobile PC, and a projector. This PC is used both for projecting the slides in the presentation tool as class notes and for recording the data required for the digital picture-card show, including digital still images from the camera, the voice of the lecturer from the microphone, and the time stamps of changing slide pages. During the class, the camera periodically takes scenes that cover the screen, the blackboard, and the lecturer. By setting up the camera at the backside of a classroom, it can take every necessary object into one image without disturbing students. The PC automatically stores the images using the camera control software. The time stamps are used to edit the images and the voice file on a lecture scene basis. The picture-card show is edited soon after the class is over, and is uploaded to the Web server. Then, students can play back the lecture scenes on demand.

4.2 Three Modules

The lecture-scene playback function consists of the following three modules:

- **Recording module:**

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

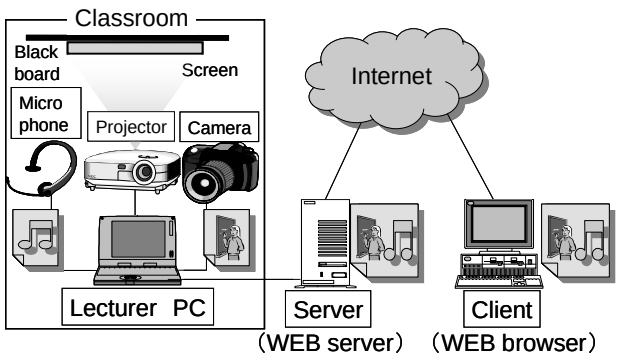

Figure 3: Overview of lecture scene playback system.

The recording module collects the data set required to edit the digital picture-card show in the editing module. This data set includes still images of lecture scenes with the resolution such that the handwritten characters and figures on the blackboard in the image can be recognized clearly, a lecturer voice file, and a time stamp file for recording slide change timings. The images are taken in the 30 second interval, and saved in the PC automatically. The voice is also saved in the PC using the encoding software. Time stamps of changing slide pages by the lecturer are recorded into the PC by using "COM object" in *MicrosoftPowerPointObjectLibrary* of *Visual C#.NET*.

- **Editing module:**
  Using the time stamps, this module selects one image of representing one lecture scene for one slide page lecture, and divides the voice file into a collection of individual files so that each one only contains the voice of explaining one slide page. Then, it generates an HTML file to compose a digital picture-card show by pairing the image file and the voice file for the same lecture scene.
- **Delivering module:**
  This module delivers the lecture scene on demand as a picture-card show to students using the Web system. A student can access to any page of the digital picture-card show, and start the voice playback from any timing or hold it by manipulating the voice operation bar in the Web browser.

4.3 Evaluation

We applied the lecture-scene playback function to one lecture in our department, and took the questionnaire from 54 students. To the question " Do you think the system has improved your learning motivation ? ", 32 students replied positively whereas 7 students did negatively. To the question " Do you think the system has improved your course understanding ? ", 35 students replied positively whereas 6 students did negatively. To the question " Do you want to use the system in the future ? ", 30 students replied positively whereas 7 students did negatively. On the editing time, the edition of 46 image/voice files for one 90-minute class required only 22 minutes for the whole procedure of copying, editing, and uploading files. Thus, we confirm the effectiveness of the lecture-scene playback function in encouraging students to study and understand classes.

5 Writing Style Check Function

The writing style check function checks the writing style of a report at the server when it is submitted from a student, and returns the result within a few seconds. This prompt feedback encourages a student to repeat the *learning cycle* of checking the writing style in the report and revising the errors until it has no error in the writing style, as shown in Figure 4.

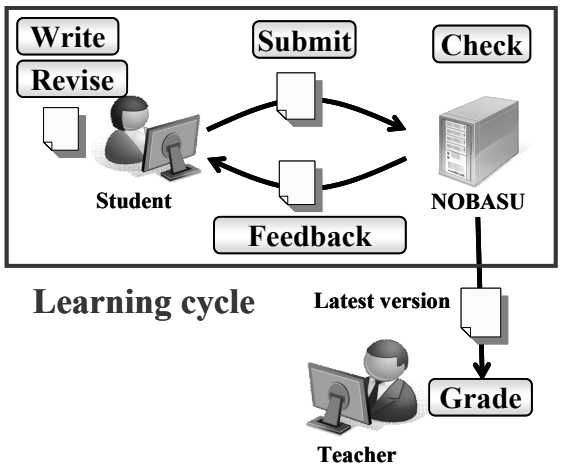

Figure 4: Learning cycle for writing style.

5.1 Three Writing Styles for Check

5.1.1 Rhetoric

Our writing style check function actually checks three items in the writing style. The first item is *rhetoric* that represents principles and rules to write a clear and readable report [7]. Actually, the following ten terms are checked in our implementation:

- **length of one sentence**: a short sentence is usually better in readability whereas a long sentence is often hard to read and understand. In our implementation, the average length (the number of characters) of one sentence is checked, in addition to the maximum length, because every sentence in a report may have a different length. If the average length is smaller than 40 or larger than 60, and the maximum length is larger than 150, this term is checked as *NG*.

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

- **literary style**: there are two styles of the sentence end in Japanese, namely the *colloquial style* and the *literary style*. Every sentence in a report should follow the literary style. Thus, if the colloquial style is found, this term is checked as *NG*.

- **unity of punctuation**: four combinations of symbols for punctuations may exist in Japanese, such as "、。", 。", ", 。", and ", . ". In a report, every sentence should use the same combination. Thus, if two or more combinations are used, this term is checked as *NG*.

- **ratio of Kanji characters**: the ratio of *Kanji* characters used in the sentences in a report should be around 30% of all the characters. If the ratio is larger than it, the tone becomes hard, and if it is smaller, the report becomes hard to read. Thus, if the ratio is smaller than 20% or larger than 40%, this term is checked as *NG*.

- **ratio of passive sentences**: passive Japanese sentences are often ambiguous because there are no subjects. At the same time, passive sentences are sometimes used for objective expressions. Then, the average ratio of passive sentences among submitted reports is calculated for the threshold. If the ratio of passive sentences in a report is 10% smaller or larger than the threshold, this term is checked as NG.

- **diversity of vocabulary**: a document using a variety of words and expressions is good in understanding and grace. On the other hand, a document using the same words in many times becomes wordy and hard to read. Instead of using the same words, similar and different words should be used. Then, the following K *characteristic value* in [8][9] is calculated to represent the diversity of vocabulary:

$$K = \frac{T - S}{S^2} \times 10,000 \tag{1}$$

where

$$S = \sum_{n=1}^{max\_n} (n \times f[n]) \tag{2}$$

$$T = \sum_{n=1}^{max\_n} (n^2 \times f[n]). \tag{3}$$

$f[n]$ represents the number of words appearing n times in the document, and max_n represents the maximum value of n. S represents the first-order moment of word appearing times, and T dose its second-order moment. This K *characteristic value* becomes large when only limited words are used frequently in the document. In a Japanese newspaper, the average value of the K *characteristic value* in editorials and columns is around 100. Thus, if the value is larger than 150, this is checked as *NG*.

- **paragraph length**: a paragraph plays an important role by focusing on one subject, so as to evolve a topic, an opinion, or a fact in a document using several paragraphs. A long paragraph can discourage readers from understanding the subject, whereas a short one may not be sufficient to express the subject. Actually, it has been known that the proper number of sentences in a paragraph exists between three and seven. Thus, if the number of sentences in a paragraph is small than three or larger than seven, this term is checked as *NG*.

- **ratio of conjunctions**: a conjunction is necessary to connect words, sentences, and paragraphs. At the same time, a conjunction should be avoided as best as possible. It has been known that conjunctions are used in 10%-20% of the sentences in a good document. Thus, if the ratio is smaller than 10% or larger than 20%, this term is regarded as *NG*.

- **sequence of particles**: a particle is necessary to connect words properly. However, if the same particle is used in a row in one sentence, it becomes hard to read. Thus, if the same particle is used in three times or more for " no " and twice or more for " ga " in a row in a sentence, this term is checked as *NG*.

- **ratio of noun-verbs**: a noun and a verb play important roles to convey messages and information in a document. The number of nous and that of verbs should be balanced in a good document. However, this ratio can be changed depending on documents. Thus, the average ratio among submitted reports on the same subject is calculated, and if the ratio of a report is 10% smaller or larger than this average, this term is checked as *NG*.

5.1.2 Report Length

The second item is *report length* that represents the number of characters in the report. The length of a document including a student report, an essay, a recommendation letter, and a technical paper, is sometimes limited into a certain range. In a report, the teacher often specifies the required length. Thus, if the report length is shorter than 80% of the specified one or larger than 120%, this item is checked as *NG*.

5.1.3 Keyword

The third item is *keyword* that represents the inclusion of the keywords in the report. The keywords are specified by the teacher as important ones in the lecture. The keywords can describe the lecture content briefly, and should be included in the report as best as possible to show the proper understanding. Thus, if the report length does not

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

include any keyword, this item is checked as *NG*, where the lacking keywords are shown to students.

5.2 Implementation

The modules for the report registration and the keyword registration are implemented for teachers to use this function. The modules of the report submission, the writing style check, and the result view are for students. The morphological analysis using the morpheme analyzer *Sen* [10] is applied to every sentence in the report to analyze the word class of every word. Besides, the frequently used keyword view is introduced for students to improve the contents by referencing the keywords used by other students. Figure 5 shows the user interface to show the check result.

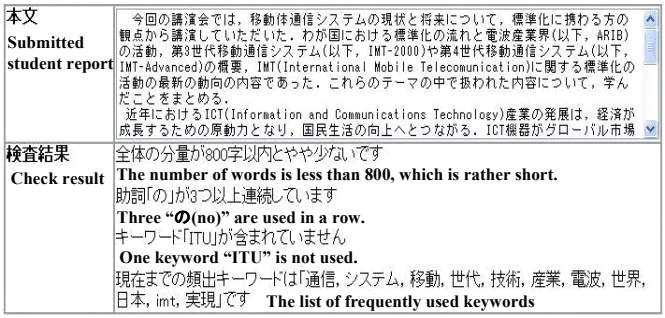

Figure 5: User interface for writing style check result.

5.3 Evaluation

We applied the writing style check function to one lecture in our department, and took the questionnaire from 42 students. To the question " Was the check result appropriate ? ", 18 students replied positively, whereas 10 students did negatively. This reason may come from the fact that the check result interface does not show the places of style errors in reports. To the question " Did you become attentive to the writing style ? ", 29 students replied positively, whereas 6 students did negatively. To the question " How many times did you resubmit the report ? ", 14 students did it three or more times, whereas 2 students did it only once. Thus, we confirm the effectiveness of the writing style check function in encouraging students to repeat the learning cycle of revising and resubmitting reports.

6 Conclusion

This paper presented the software architecture and three distinguishing functions of *keyword submission*, *lecture-scene playback*, and *writing style check* in *NOBASU* as the original LMS to support educational activities in our department. The evaluations using questionnaires in their applications to real classes confirm the effectiveness of these functions. Our future works include the usability improvement of these functions and the development of new original functions to contribute learning and educational activities.

References

[1] Grob, H.L., Bensberg, F., Dewanto, B.L., "Developing, deploying, using and evaluating an open source learning management system," J. Comput. Inform. Technol., pp. 127-134, 2004

[2] Harrington, C.F., Scott A. Gordon, S.A., Schibik, T.J., "Course management system utilization and implications for practice: a national survey of department chairpersons," http://www.westga.edu/ distance/ojdla/winter74/harrington74.htm.

[3] Coates, H., James, R., Baldwin, G., "A critical examination of the effects of learning management systems on university teaching and learning," Tertiary Edu. Manage., vol. 11, pp. 19-36, 2005.

[4] Ma, Z., *Web-based intelligent e-learning systems: technologies and applications*, Idea Group Inc., 2006.

[5] Cole, J., Foster, H., *Using Moodle: teaching with the popular open source course management system*, 2nd edition, O'Reilly, 2007.

[6] Funabiki, N., Mouri, A., Hassan W., Nakanishi, T., "A lecture scene playback system using digital picture-card show for distance learning," Proc. INVITE, pp. 1071-1076, 2008.

[7] Kinoshita, K., *Composition technique for science course*, Chuokoron-sha.

[8] Ishioka, T., Kameda, M., " Automated Japanese essay scoring system, " Japanese Society of Computational Statistics, vol. 16, no. 1, pp. 3-18, 2003.

[9] Yule, G.U., *The statistical study of literary vocabulary*, The University Press, 1944.

[10] Sen (Morphological analyzer written in Java), https://sen.dev.java.net/.

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)