Chapter 19

The Development of Educational Environment Suited to the Japan-Specific Educational Service Using Requirements Engineering Techniques: Case Study of Running Sakai with PostgreSQL

Yuki Terawaki
Hosei University, Japan

Yuichi Takahashi
Foreach Ltd., Japan

Yasushi Kodama
Hosei University, Japan

Kazuo Yana
Hosei University, Japan

ABSTRACT

This paper describes an integration of different Relational Database Management System (RDBMS) of two Course Management Systems (CMS) called Sakai and the Common Factory for Inspiration and Value in Education (CFIVE). First, when the service of CMS is provided campus-wide, the problems of user support, CMS operation and customization of CMS are clarified from past experience. Additionally, various Japanese specific educational practices are reviewed. The candidate CMSs are selected by using proposed methods at Requirements Engineering (RE) research community. Finally, in order to consider possible application of these CMSs, the running environment of different two CMSs are discussed, and the integration of different RDBMS on two CMSs is described.

DOI: 10.4018/978-1-4666-2032-2.ch019
INTRODUCTION

Needless to say, CMS and Learning Management System (LMS) are used extensively by higher education in Japan. In Hosei University, CMS has been used since 2007 by users campus-wide. We have been operating and supporting the CMS for users campus-wide. Currently twelve thousand courses are registered with the CMS. The CMS are used by almost 90 percent of all students and teachers in HOSEI University. Various issues need to be considered from past experience (Terawaki, Takahashi, Tokiwa, Kodama, & Yana, 2010).

Large-Scale Application

In HOSEI University, there are many classes that register hundreds of students. In such classes, utilizing CMS is especially effective. Thus, CMS needs a lot of hardware capability and efficiency-enhancing architecture. For example, in order to avoid problem running the CMS using several Application Servers, we have to consider problems such as avoid storing a data in a program or decide whether or not to use session replication.

Sudden/ Short-Term High-User Demand on CMS

As deadline approaches, if a lecturer put a time limit on the report, many users simultaneously log in to CMS to submit reports. During the period of examinations, deadlines for submission of reports tend to be concentrated on specific dates. Another problem is, at the beginning of a new term, the function of self-registration to register to a course of CMS is used by hundreds of students simultaneously. Therefore, it is essential to consider the problem of sudden short-term high-user demand to CMS not only users of a particular class but also several users from different classes accessing simultaneously.

Interoperability

The authentication check of CMS needs to work together with the authentication system constructed by HOSEI University.

Also when a course is registered in the CMS, course-IDs are imported from the system of academic affairs. Moreover, the data of the attendance management system relate to the CMS. For example, the data of the attendance management system is not only displayed on the CMS but also attendance in the initial class is used to register learners to a course on CMS before determinate registration.

Customization

The function of CMS should be redesigned according to a variety of practice in the university. A Functionality requirements for the CMS varies from class to class. For example, required function for Bulletin Board System (BBS) might be the difference to learn either individually or in groups.

User Support

To support all users (Lecturers, Learners and Staff), it is preferable that the CMS has a better minimal function than an improperly matured function. To support thousands of users, our CMS should be kept as simple as possible.

However, even if the functions of CMSs are appropriate, it is difficult to fulfill the needs of faculty staff. Therefore we have to develop tools to satisfy this need. This means that the Application Programming Interface (API) of the CMS should be open. Moreover, it is preferable that the source code of the CMS is also opened.

The current CMS is difficult to redesign due to license and technical restriction because commercial software package CMS is used. It is well known that vendors of commercial-off-the-shelf (COTS) try to meet the needs of the marketplace instead of satisfying the requirements of