Web-Based Testing and Self-Assessment System Implemented with Open Technologies

MARKO CARIC                             MILAN TUBA                            IOANA MOISIL
Faculty of Computer Science        Faculty of Computer Science       Dept. of Computer Science
University Megatrend Belgrade    University Megatrend Belgrade   Lucian Blaga University of Sibiu
Bulevar umetnosti 29               Bulevar umetnosti 29               4 Emil Cioran st.
SERBIA                               SERBIA                               ROMANIA
caric.marko@gmail.com                    tubamilan@ptt.rs                      im25sibiu@gmail.com

Abstract: - This paper describes implementation of a web-based system for testing and self-assessment implemented with open technologies. All standard testing parameters can easily be adjusted so the system can be used for many different purposes. Importance of web-based applications is directly connected with the fact that clients’ side is free of any additional software installation. Programming technique is more complicated, but final results are more robust and useful. There are enough free software tools to implement almost any such system. The advantages of open technologies are well known and the target platform can be Linux, as well as Windows. Using different algorithms we can achieve client-server communication with a relatively few web pages and JAVA applications. The ability to update and maintain web applications without distributing and installing software on potentially thousands of client computers is a key reason for popularity of such applications. The system was tested and proved to be reliable and comfortable to use.

Key-Words: - Testing software, Course Management System, Open technologies

1 Introduction
Web-based applications have important place because of Internet continual growth. The ability to update and maintain web applications without distributing and installing software on potentially thousands of client computers is a key reason for their popularity. Clients can get answers for different requests, but also clients can easily change server’s abilities if server is properly programmed. Interesting usage of such kind of applications can be found in teaching process. Students (and teachers) represent client side, and web application represents server side.

Teaching process may be improved by assessing the students’ progress in some subjects throughout the year. Thus we can achieve an objective picture of their knowledge at certain point and take necessary measures aimed at improvement and advancement of knowledge in timely manner. It is also essential that self-assessment is available to students at every moment in every place. This is achieved by means of a web application, an application that will enable self-assessment on the Internet at every moment. Teachers on the other hand can from anywhere adjust server side of the application and change it to suit current requirements.

Free software tools are widely used and very often can completely satisfy users’ requests. They offer greater control over application because of open source concept. Intension for using such kind of software is better understanding inner application structure. We consider that is very important in situation when we naturally predict new future demands. Also, free software often presents simplicity and scalability, because of platform independency.

There are several, open-source or proprietary, testing systems that are widely used. Moodle is a Course Management System (CMS), also known as a Learning Management System (LMS) or a Virtual Learning Environment (VLE). It is a software package for producing Internet-based courses and web sites. Moodle is provided freely as Open Source software (under the GNU Public License). Moodle can be installed on any computer that can run PHP, and can support an SQL type database (for example MySQL). It can be run on Windows and Mac operating systems and many flavors of Linux. The word Moodle was originally an acronym for Modular Object-Oriented Dynamic Learning Environment and it is also a verb that describes the process of lazily meandering through something. As such it applies both to the way Moodle was developed, and to the way a student or
teacher might approach studying or teaching an online course.

Sakai is also a CMS, VLE or LMS. While Sakai is typically used for teaching and learning (similar to products like Blackboard and Moodle) it can also be called a Collaboration and Learning Environment (CLE) because it embraces uses beyond the classroom. Sakai is distributed as free and open source software under the Educational Community License. Sakai’s open approach allows institutions to deploy, host and modify the software in whatever manner best supports institutions skills, needs and goals. Sakai’s open development process allows local enhancements to be incorporated into future versions of the software. Sakai embraces both open source and open standards. It uses open source software whenever possible and base as much work as possible on open standards. Sakai enables an open approach to support models allowing choices between institutional support, community support and commercial support, each with unique benefits and costs.

Hot Potatoes suite includes six applications, enabling creation of interactive multiple-choice, short-answer, jumbled-sentence, crossword, matching/ordering and gap-fill exercises for the World Wide Web. Hot Potatoes is freeware, and it may be used for any purpose or project. It is not open-source.

Questionmark Perception assessment management system enables trainers, educators and testing professionals to author, schedule, deliver, and report on surveys, quizzes, tests and exams. It measures knowledge, skills and attitudes securely for certification, regulatory compliance and successful learning. Questionmark Perception is available for on-premise deployment or as a hosted, on-demand solution.

Random Test Generator-PRO was designed for educators at any level to develop TestBanks of questions from which randomly selected questions are then extracted to create tests. Each TestBank acts as a repository to house a wide range of questions for a specific topic. Questions may comprise of multiple choice, True/False, Fill-in and/or essay types. Multiple choice test items may also have multiple (correct) answers if desired. All test items can be heard as well as read from the screen. This Text-to-Speech (TTS) feature provides students an added benefit to hear a synthesized voice reading questions and answer choices out-loud. The Test Builder option can be set to identify exactly how many questions to select from each TestBank to create a single test. It will then randomly select questions within each TestBank requested by the user without selecting the exact same question twice. These tests may be administered on the computer with an optional timer, printed on paper or administered via the Internet. Since each test is unique, answer keys may be printed for every paper test developed. Tests administered on computer may be automatically graded. Tests administered via the Internet allow students to easily submit their answers via e-mail for grading with a single-click. Image, sound, movies, animations or any file can be added to test items.

Hosted Test is the premier online testing and assessment software widely used by educators, human resources professionals and performance improvement specialists all over the world.

2 System Description and Requirements

There have been many efforts to improve teaching process, even using artificial intelligence techniques [1], [2], [3]. In this paper we will concentrate on a practical system solving students’ and teachers’ requests. Experience shows that students find testing most useful after one area of some subject is covered. That is why we need effective mechanism to refresh their knowledge as soon as it is possible. The next requirement that the application must satisfy is the option of final testing, when the examiner needs a number of test adjustment possibilities. The output results must be in a readable and usable format. The application must allow extensions to meet any future requirements and simple inclusion of new subjects. Furthermore, client-side requires browser only [4], so that application scalability is based on the fact that any change affects server only (Figure 1).

![Figure 1. Client and Server Application](image)

It is very important that teachers can control and adjust the application or see testing results for chosen date at any moment, from anywhere. In other words, testing system should be available if Internet...
is available. Of course, recommended environment for final testing is local network.

In addition, application must have an option to randomize order of questions and offered answers (Figure 2). From a security point of view, this is useful and even necessary.

![Randomized questions](image)

**Figure 2.** Randomized questions

It is also important that learning process is comfortable enough in given environment [5], [6], [7], [8]. Application usage is rather simple and comfortable from users’ point of view. Links on the screen guide user through desired activities. In the case that any requested input is omitted, the user will be informed by JavaScript alert. As a security measure, if user spends more than 30 minutes without any action in application, the connection will be reset, and user must log in again. Application security is based on password secrecy. Forgotten password may not be recovered, because it is stored in the database in hash format. We emphasis that we specially wanted results directly in excel format without any additional conversion because of its usability for this purpose. Also we wanted database contains data in ASCII format. Because browser offers many options, for our purpose we need to disable some tasters (e.g. f5, refresh button, alt + combination).

3 Application Structure

The testing system consists of two sections:

- Users’ section providing testing and self-assessment
- Administrator section providing maintenance of the database containing the information on users, questions and details regarding sessions.

The users’ section enables logging in, registration and testing. Administrator section enables setting the testing parameters, input of testing questions and modification and testing area as well as registration of new users and subjects. The application contains 15 JSP pages and 17 servlets (Figure 3).

At the beginning, application offers Login.jsp page. This page contains link towards another page Register.jsp. Each student doing test/self-assessment must first register and his personal particulars are entered in the database (the password is saved in hash format and must have at least 6 characters). Each log-in thereupon directly opens the testing page with the parameters currently applied.

Identification check is established by means of the following security feature: name and surname and student identification number which are entered at the time of registration appear at the top of the testing page (Figure 4) and have to be compared with the number of the identity card provided by the student.

Page Register.jsp invokes Register servlet in order to store new user in the database. If such user already exists in the database, registration process will be repeated. Mentioned mechanism is valid only for users with the lowest privilege (privilege 1). Advanced users (with privilege 3 or 5) insert new user particulars using administrator section.

![JSP pages and servlets](image)

**Figure 3.** JSP pages and servlets
Depending on the given privilege, Login page invokes Login servlet to allow testing or administration. In a testing case, Login servlet invokes Service servlet, which services the given users’ answers (from testing page Test.jsp), puts them in the database and sends the output results in a form Results.jsp page. In rare cases, when the database is empty, Service servlet will be redirected to Login page. If user with high privilege (3 or 5) successfully logs in, application directly opens basic page Choice.jsp. It is possible to choose one of the following options:

- to set application (Settings.jsp)
- to see testing results (Results.jsp)
- adding new users (AddUser.jsp)
- adding new questions (AddQuestions.jsp)
- adding new subjects (AddSubjects.jsp)
- deleting and modification exciting questions (ChoiceDeleteModify.jsp)
- adding and deleting areas (AddArea.jsp)
- logout

Page Settings.jsp invokes Setting servlet which stores chosen settings in the database. Also, Setting servlet communicates with Ch!oseQuestons.jsp page in which some questions become “active” when chosen by the examiner.

Page Options.jsp allows the examiner to choose particular date in drop-down menu in order to see all testing results for the date chosen. Using ForwardDate servlet, page Display.jsp displays all testing results in table, in excel format (invoking Excel servlet) or by mail (invoking Mail servlet). Pages AddSubject.jsp, AddQuestions.jsp, AddArea.jsp invoke equivalent servlets with the same name.

Page ChoiceDeleteModify.jsp cooperates with servlet ChoiceDeleteModify and indirectly with DeleteModify.jsp page, ModifyQuestion servlet and DeleteQuestion servlet. Thus, some questions, answers or areas could be changed or deleted. Every logout passes through Logout servlet, where all session attributes are completely destroyed.

Going back to a previous page is excluded. Every attempt to go back to a previous page passes through Brake servlet; it displays the same page again (with remembered remaining time for answer). Similar situation happens if user uses refresh option.

Validation of wrong or missing input is ensured by JavaScript on client side. It is a typical situation in which there is no need to use server side.

It is required to test students’ knowledge in particular subject. Consequently, it is also required that every application be separate subject. That is why every new subject is represented by means of new application on server. It is achieved by copying complete directory and subdirectory structure using servlet AddSubject. The mentioned servlet also does necessary changes in other files on server (e.g. server.xml). This act is completely automated. When we make new subject, together with new application, new database with the application’s name automatically is created. Initial database is empty with one user with the highest privilege.

### 3.1 Privileges

Privileges granted to certain users can be classified into three categories: privilege 1, privilege 3 and privilege 5.

Privilege 1 users — students who do testing/self-assessment without any further additional options.

Privilege 3 users — users who have access to administrator section without authorization to enter new users or new subjects.

Privilege 5 users — all rights granted.

As mentioned above, each subject is represented by separate application and uses separate database. It means that a new application with blank database and only one privilege 5 user needs to be created when including a new subject. The procedure of adding new subject must be done on the server itself while all other operations can be done from any other computer. It should be noted that new subjects are not often added.

Privilege 3 users can access the parameter adjustment page as soon as they log in. They can also view the previous testing results as all previous testing sessions were recorded in the database under a specified date. It is also possible to obtain the results in excel format or to forward them to a specified e-mail address.
3.2 Technologies Used
The following technologies were used to generate application: Programming language JAVA (JSP and servlet technology), Database server MySQL 5.0 and web server Apache Tomcat 6.14. Servlets are server-side Java programs that generate web content dynamically depending on the content of a request from a client [9]. Java Server Pages present extension of Java Servlet technology for combining Java server-side programs and HTML. JSP pages have an extension jsp. Communication between JAVA servlets and database is established by means of JDBC driver mysql-connector-java-5.0.8-bin.jar. All above mentioned technologies are free of charge.

4 Database Structure
Database structure consists of seven tables and two stored procedures (Figure 5):

Table users contains information about users such as username, password, first name, last name, index and privilege. Relationship between table users and tables evidence and results is done by primary key UserID. Table evidence contains session information and table results contains date, points number and percent. Table questions contains questions, time for answers and field to check whether the questions is active. Also, table questions creates relations with table evidence, answers and area, by primary key QuestionsID. Table answers has information about answers and points. Table area contains name of area and it is connected with table settings by means of primary key AreaID. Table settings contains all information about current application settings.

Figure 5. Database tables

Stored procedures InsertUsers and SelectUsers manipulate with users in database. Stored procedures are used in cases when there is a possibility of security breach by means of SQL injection.

Connection between programming part and database is established using JNDI technology. It is done in servlet Connection.

Database contains data in ASCII format, so we can manipulate with special characters including the Cyrillic alphabet. It enables database service through application only, so it also represents a security measure. It is additional reason why we develop own application despite many similar already exists.

5 Setting Testing Parameters
The following should be defined in order to set the parameters (Figure 6):
- whether the testing time is limited
- whether the time allowed to reply to each question is the same
- the format of the results
- the order of questions
- whether there is a possibility of reply “I do not know”
- choice of testing area

If any of the options is omitted, the preset value will be applied. After the parameters are set, the program is automatically directed to the question page to activate the required questions. It is possible to choose all questions at once, or selectively one by one in the chosen testing area. Maximum of 6 options are available for each question. However, not all six options need to be offered. During testing, the question order and the order of reply choices are randomly selected by the program (if this option is selected by the examiner).
6 Conclusion
The web-based application for testing that was developed and is described in this paper proved to be reliable and comfortable for use. Students find testing and self-assessment using such kind of application, with or without limited time, very stimulating to improve general knowledge or to correct specific details. Examiners also have effective method to assess students and improve their own teaching level. Open source technologies that were used, namely, JAVA, Apache Tomcat and MySQL proved to be adequate for such project. Application can be adjusted for many different testing environments by setting parameters in a control form and additional parameters can easily be added. The further improvement, which is rather complicated, will include possibility of giving text answers rather than only multiple choice, but that will have to be based on some AI algorithms for assessment of the quality of the answer.

References: