ALTERNATIVE COMPUTER BASED TESTING SYSTEM

© Péter SZIKORA
(Óbuda University, Budapest, Hungary)

Szikora.Peter@kgk.uni-obuda.hu

Received: 11.06.2013; Accepted: 19.12.2013; Published online: 28.12.2013

Exams are the endpoint, and therefore also the point of feedback in an academic learning process. Since teachers in tertiary education are more and more overloaded, it is inevitable to simplify test as much as possible. One way to do so is the application of electronical testing programs. The aim of this paper is to present an examination software, which is easily customized, and differs from the traditional software to be found on the market due to its easily customizable nature.

Keywords: ICT education, software development

Nowadays, the number of students relative to that of teachers is growing at our university therefore teachers have less time for one students. Hence teachers started abandoning oral exams and applying more written and electronical ones. However, since one of the most important parts of the teaching-learning procedure is the exam, providing feedback to students, it is important to have appropriate exams, if necessary oral ones. My idea to solve this problem was the pretesting of students before oral exams. With the help of pretesting, students prepared for the exam can be selected, and be made to take oral exam afterwards, while those, not prepared will not take up the teachers' time and effort. Of course the pretesting of students can be solved with the help of a suitable computer based testing program as well.

The idea of testing software is not new. There are some very good programs like moodle on the market. Most of them use closed questions, so students need to choose the good answer from four or more answers. However, most of them are not customisable enough and hence does not meet the expectation of teachers. This is why I decided to develop an examination software that is compatible with the already existing moodle environment since the students are already registered in this system (imported from neptun system), and no other authentication system is needed.

Process of the software development

Some years ago I planned an alternative information system for higher educational purposes and modelled testing within it with my colleagues. My experience during that project was that the better the data model, the better the scholar information system is functioning. Accordingly I planned and fine-tuned its basic database, and developed a matching testing system. The

results of the experiment were better than I had expected (Szikora, 2009a, 2009b; Keszthelyi, 2009a, 2009b, 2009c, 2010). The registration of each students in the project were successful. What is more, thanks to our well equipped computer lab with, I was able to check 30 students' preparedness for the exam online, simultaneously; sparing time and effort of correcting traditional paper and pencil exams.

In 2013 I developed a little, clear and consistent data model for testing a SIS module. Since I already new that a better data model saves resources, my goal was to make a better information system than the existing Scholar Information System (SIS). I used the already pre-tested data model, free software and a normal pc as test environment. In the forthcoming chapters I will address the IT environment, the software, and the data model one by one.

IT environment

In order to make my program work, I needed a server with Linux operating system that runs a database management system - in my case MySQL - and a web server (Apache), with PHP as a host programming language. This server - as described above - can be a normal pc running only free software, so does not need any extra investment other than (a usually already existing) personal computer.

Software

The first and the most important advantage of my alternative testing system is that being based on free, open source software.

Speaking about free software, 'free' means freedom, the freedom of using, understanding, modifying, developing and/or (re)distributing of the 'free' software. In addition, free software are usually free of charge, too, but this is not part of the definition of free software. Free software can have higher quality than proprietary, commercial software because there are no business aspects considered (i.e. to achieve profit) while developing them. You can utilize the software in any way to have a more proper tool for your purpose.

Free software usually have much bigger testing teams than enterprises can even dream of. In case of using free software there is no dependency: if the original developers/maintainers of the software stop their work, you can and are allowed to continue.

And last but not least, by using free software one should reach a higher level of computer security because of having their source code, too, while in case of commercial software you have only binaries without any source codes and in addition you are explicitly forbidden even to try to check the software you use.

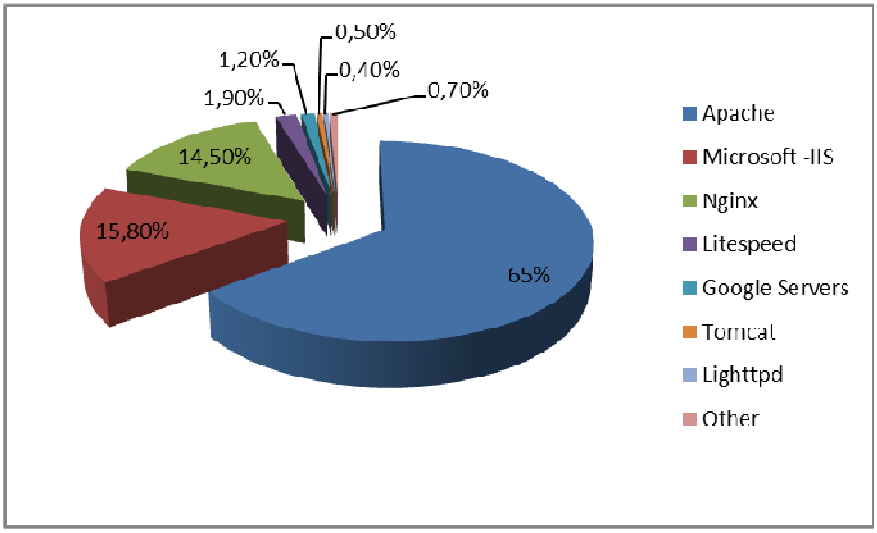
The Open Source Definition is used by the Open Source Initiative to determine whether or not a software license can be considered open source. The definition is based on the *Debian Free Software Guidelines*, written and adapted primarily by *Bruce Perens* (1997) (http://www.debian.org/social_contract). Perens in turn based his writing on the "four freedoms" of Free Software from the Free Software Foundation.

In the specific case of the alternative testing program I have chosen Linux Operating system because it is free and responsible. Linux is a free Unix-like operating system originally created by Linus Torvalds with the assistance of developers all around the world. Developed under the GNU General Public License the source code for Linux is freely available to everyone.

The comparative advantage of Linux to Windows the Linux is a free software and accordingly an open source one. This means that the granted easy accessibility enables numerous testers and developers to experiment with and analyse the system. Consequently, this operation system grants higher availability and greater reliability.

The reasons for choosing Apache as the web server are numerous. The primary reason, however was to cut down on licensing costs, and the active support and development of it. Apache, on top of Linux, is used in most cases all over the world so it can be considered the most accurately tested http server. See Figure 1.

Figure 1. Percentages of websites using various web servers June 2013

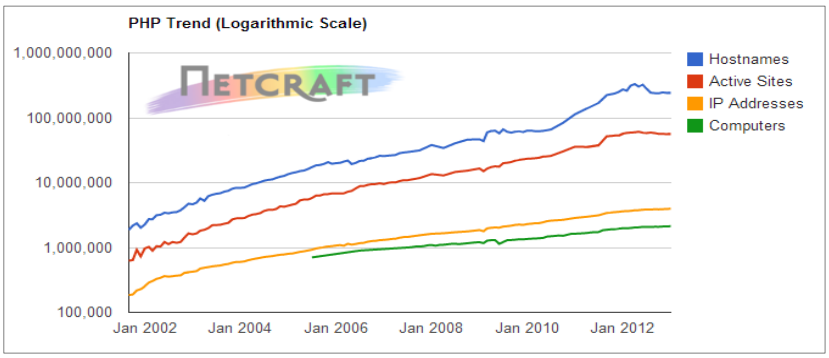


Source: http://w3techs.com/technologies/overview/web_server/all

The MySQL database management system has become the world's most popular open source DBMS because of its consistent fast performance, high reliability and the ease of use. MySQL runs on more than 20 platforms including Linux, Windows (Mysql.com…, 2013). My top reasons for using MySQL were its availability, scalability, flexibility, high performance, strong data protection, open source freedom and 24/7 support provided at the lowest total cost of ownership.

Choosing the programming language and tools there are two main fields which determine the performance of the developed program. These are the quality of the applied algorithms and the quality of program coding. When deciding on PHP I did not only consider its unquestionable popularity, but its detailed documentation, its easy to use nature, and it also being a free, software. My decision is supported by those of many from the past years, since in 2006 only 20 million sites were running PHP, but by 2012 80% of all the websites are running it. Figure 2 shows that PHP broke the 100 million domain border, which means that more than 100 million sites were running PHP by the end of 2011.

Figure 2. PHP usage statistics for Jan 2013



Source: PHP.net, PHP usage, http://www.php.net/usage.php (2013)

Database Plan

After the hardware and software has been chosen I needed to develop a good data model. There are general prerequisites of the goodness of data models. At the conceptual level a good data model needs to be understandable, unambiguous, realistic, full and minimal (Halassy, 1995). The efficiency of a database is characteried by it's capability to cope with high loads. This capability is determined by very different factors such as hardware environment, software environment (operating system, relational database management system, application programming language and tools, application programs themselves) and last, but not least, the quality of the data model (Keszthelyi, 2009b).

In my case the influence of the hardware environment was not very important, because the alternative testing program will be administered by 2-3 people only. The central elements were the software, the operating system, the relational database management system - as already described above - and the application itself.

The main steps in developing an information system are: determining what we want as exactly as possible, data modelling (i.e. determining the data structure) and finally determining the functions operating on the data structure.

Data modelling is the basis, which is necessary but of course not sufficient for success. It is only a foundation on which a good information system can be constructed. In case of data-intensive systems the data structure is the most important and even determines the functionality (Raffai, 2003). So in order to succeed, according to dr. Halassy, we need a three-level data modelling and planning (Halassy, 2002). These levels are the conceptual, the logical and the physical levels.

Specifications for the database

Every topic addressed by the alternative testing system has 2 different question-groups, public and private. Public questions are accessible from home, but private questions are only accessible within our computer lab. So students can learn at home, and test their skills of knowledge.

Every question is paired with one or more correct answers and three or more bad ones. However, the order of the answers are random, so on each and every exam the order of the possible answers is different, so that the students do not automate their choice of good answer, and concentrate on the content and not he order of the correct answer.

In each test students are facing a row of 10 different questions, that should be answered 1 minute each. The time limit is determined in order not to allow talk with the classmates, or allow cheating due to spare time. 1 minute is (only) enough to read the question, the answers, and choose the right answer from the list.

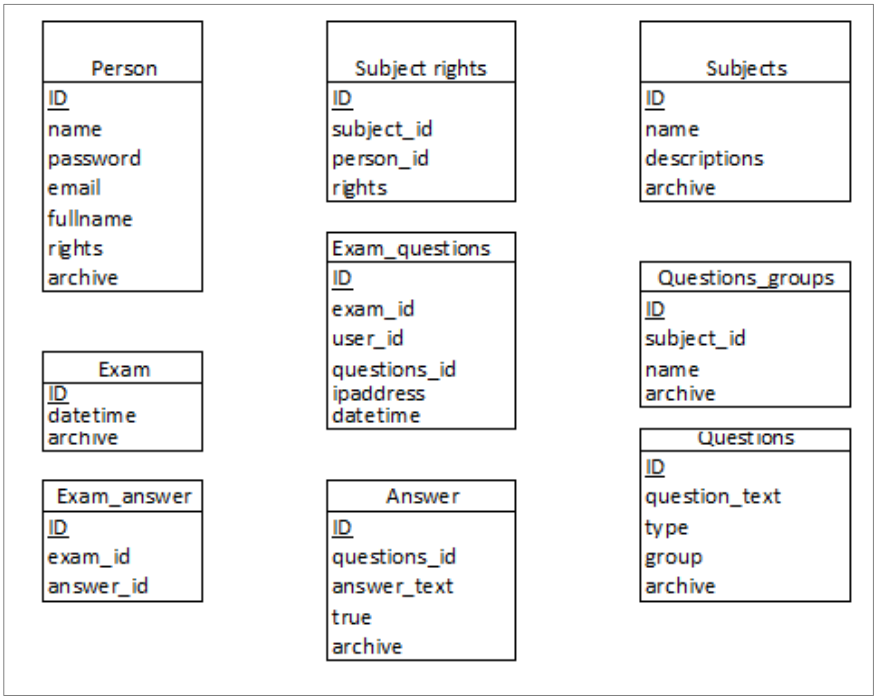
Since the aim of this alternative testing program is to make sure, only the prepared students can take the oral exam it is important that those sitting next to or in front of each other do not get the same questions.

In addition to the above mentioned features the program stores all the questions that a certain student has been exposed to in a given exam period, and is able to select the questions for retake(s) from those not asked, or not correctly answered before.

Data model

In my data model there are 9 entities. You can see the data structure in Figure 3. and 4. In the Person table the primary key is the ID. We can use NeptunId for this purpose since the NeptunId is a unique identifier, generated by Neptun Scholar Information System. The other fields are optional e.g. fullname, personal data, rights or password.

Figure 3. Database schema – tables



In Subjects table there are some data about the courses that we teach e.g. name, descriptions. The primary key in this case is subject ID. The connections table is the Subject_rights. This table connects the Person(students) and the Subjects(coures) to each other and contains the students active subjects and the rights for it. The primary key is subject_rights_ID. The foreign keys are: person_ID and subject_ID

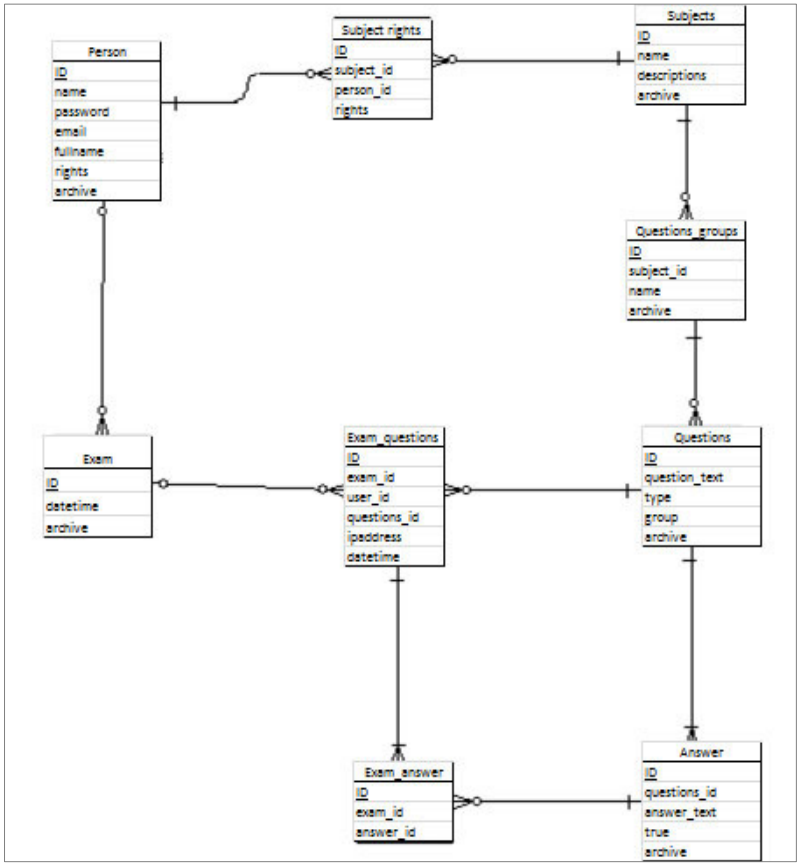
There are other four tables Exam, Questions, Answer and Questions_groups. In Exam table we can store all data of the exams, archive bit shows whether the exam is active or not.

In Questions table we can store all the questions grouped by questions_groups, and connected to subjects. The foreign key is in this table

Group_ID. In the Answer table, we can find all the answers for questions stored in questions table. The true data store whether the particular answer is right or not.

All students answer numerous questions(10 at a time) and we need to store those questions in exam_questions. We need to know which questions did the student receive earlier, and store the answers' id in exam_answer to know whether the student's answers were good or not, if we want to prevent that the student gets the previously correctly answered questions again. We also have to identify the computer and define 'nearby computer' for each and every in order to prevent nearby students receiving the same questions.

Figure 4. Database schema – data structure



Conclusion

At present at Óbuda University there are two separate information systems for students. One for storing e-learning materials and one for storing student data. Present paper endeavours to show that there can exist a more effective information system which, - besides storing student data along with test questions and answers and therewith serving as an interactive teaching system, - can also help in enlightening the burdens of loads of oral exams on teachers. The system presented in this paper, being one made with the use of free software and an ordinary pc, is easily applicable to all tertiary educational institutes, and is more customisable than those ready made application on the market. Besides cutting down on IT costs, the advantage of the described system is that through knowing the database structure that lays behind the program one can always adjust the system to the most recent needs.

References

Debian, Social Contact, , 1997
HALASSY Béla (2002). *Adatmodellezés*. Budapest: Nemzeti Tankönyvkiadó.
HALASSY Béla (1995). *Az adatbázistervezés alapjai és titkai*. Budapest: IDG Magyarországi Lapkiadó Kft.
KESZTHELYI András (2009a). How to measure an Information System's Efficiency? In *7th Conference on Management, Enterprise and Benchmarking* (pp. 213-219). Budapest: Budapesti Műszaki Főiskola.
KESZTHELYI András (2009b). Remarks on the Efficiency of Information Systems. *Acta Polytechnica Hungarica*, *7* (3), 153-161.
KESZTHELYI András (2009c). The Role of Data Modelling in Information System Efficiency. In *2nd International Conference for Theory and Practice in Education*.
KESZTHELYI András: Extended measurement of an information system's performance. In 8th International conference on Management, Enterprise and Benchmarking (pp. 209-216). Budapest: Óbudai Egyetem.
Mysql.com, Mysql Documentation (2013). Retrieved from http://dev.mysql.com/doc/ [12.12.2013]
PERENS, Bruce (1997). Debian Free Software Guidelines. Retrieved from http://www.debian.org/social_contract [12.12.2013]
PHP.net, PHP usage, http://www.php.net/usage.php (2013).
RAFFAI Mária (2003). *Információrendszerek fejlesztése és menedzselése*. Budapest: Novadat.
SZIKORA Péter (2009a). Measured Performance of an Information System. In *7th International Conference on Management, Enterprise and Benchmarking*. Budapest.
SZIKORA Péter (2009b). The Role of the Tools and Methods of Implementation in Information System Efficiency. In *2nd International Conference for Theory and Practice in Education*. Budapest.