LABCENTER. A remote laboratory system platform


**Dpto. Ingeniería. Pontificia Universidad Católica del Perú. Perú

Abstract: A web system server especially suited for remote laboratories has been developed. Typical e-learning systems do not offer the possibility to perform a remote laboratory where real experiments can be done online, accessing real hardware located at the University facilities. Allowing students to connect to hardware systems remotely provides them with additional knowledge about real devices; very often, real laboratory devices are time or space restricted. The proposed LABCENTER platform is a general frame designed for remote laboratories connection. The platform is designed to allow an authorized student to connect to hardware systems. As direct hardware systems allow only a single user to be connected, a time-slot scheduling is included in the access system. The proposed platform includes a teacher administration system where new courses can be created and, inside courses, laboratory sessions can be defined. Every course and laboratory can be easily customized by teachers, without the need to know specific web programming languages. The web system is designed to connect to hardware systems through plug-ins specifically programmed to exchange information with the hardware. Using this approach, almost any experiment (hardware system) could be connected to the server. Once designed, the teacher can include the hardware connection in the layout editor, where the teacher can arrange all available plug-ins that will later be shown to the student. Using an individual plug-in philosophy directly connected to individual hardware allows more flexibility to design different laboratory proposal to students. As an example, two laboratory proposals are described. First, an Industrial Robotics laboratory to control an ABB industrial robot is defined, the teacher defines different proposals to students based on specific robot plug-in controlling the hardware. Secondly, an FPGA laboratory system is described, in which the student can download his own developed program and test if it works properly.

Keywords: Learning platform, remote laboratories, remote experiments, remote control platform for education

1. INTRODUCTION

The use of remote laboratories is becoming of special importance for students due to the possibility of downloading programs and controlling hardware systems from any location. This approach allows the students to have more contact with special hardware systems and some facilities that are not easy to use due to the risk of direct usage or the time availability limitation existing in case of those hardware systems where reduced units exists at laboratories.

Due to the new approaches to teaching education systems, it becomes very important to offer flexible-learning procedures adapted to student needs, providing ubiquitous access to information, but also to learning tools and hardware. This is of special importance in those studies related to engineering, where the use of hardware equipment is a must. Today, it is common for a student to be a part-time worker and thus, it is not possible for him to attend lectures and labs regularly. Moreover, homework activities are necessary for the students to complete the topics covered in the classroom, making homework activities an essential part of the learning process at the university.

On the other hand, students must acquire independently certain capabilities not directly related to technical content. For instance, they must be able to organize their study time, read and understand technical documentation so they can succeed with the proposed task, and acquire a good familiarity with lab tools. Lecture hours at the university are high, thus, the only way to increase the practicing time and acquire more experience with hardware systems is by means of the completion of remote lab sessions that will provide the necessary skills to work as in real applications.

Nowadays, it is common to find different web servers allowing students to handle hardware equipment remotely. However, these web systems are oriented to specific applications. The development of Internet systems for laboratory experiments using remote access is an important field of work; they are called web-labs, e-labs, remote labs, etc. (Jing, 2006) wrote a good review of the current state of e-lab deployments and its learning implications.
Several possibilities to carry out electronic experiments by Internet are described in (Fjeldly, 2003). In 2002, the work proposed in (Gustavsson, 2002) was awarded the best work prize for remote access control of electricity experiments due to its novelty and interest, (Hercog, 2005) and (Herrera, 2006) carried out several proposals for remote laboratories in the engineering field. Other authors propose a remote robot programming using the Internet. In some cases, the web grants access to a simulated environment, not to a real robot (Candelas Herias, 2006; Candelas Herias, 2003). Other experiences offer access to a real robot (Šafaric, 2005), but an educational robot type, not an industrial robot. The same situation arises in (Guimaraes, 2003), which presents the remote control of a mobile robot through a remote service. The robot can be moved remotely across a room while it is observed with two panoramic cameras from the user's Web browser. Finally, (Tzafestas, 2006) shows a comparison between virtual and remote implementation of robotic laboratories. Other applications such as remote FPGA programming, control and debugging have been proposed (Iskra, 2009).

Concerning e-learning platforms, it is common that most of the universities offer an e-learning platform as a support tool to students. There exist several software packages conceived for supporting the tasks commonly used in online courses that ease the work of educators in creating courses and defining the information given to students through the web. Typical examples of open-source software packages are Moodle (Moodle, 2010), dotLRN (Dotlrn, 2010), ILIAS (Ilias, 2010), Dokeos (Dokeos, 2010) and ATutor (Atutor, 2010). However, none of these software environments provides tools for remote laboratory management. Existing e-learning platforms provide an excellent framework for distance communication between student and teacher, information repository, agenda, evaluation and grade management.

However, remote access to hardware systems is not a general task as common as other actions that e-learning systems typically propose. Each hardware device needs a different communication procedure for sending and receiving information as well as a different interface with the user. Depending on the kind of device, the user may need to send data or values, and receive more data to be further shown as a graph, table, or saved in a file. Web communication with hardware systems is not novel. Most manufacturers, especially in the electronics, automation and industrial systems fields, offer tools to control and send/receive data via web. In these cases, the problem arises when trying to integrate different hardware communication systems in a common framework, especially when it is oriented to learning purposes.

In this paper, we describe a remote laboratory web server oriented to control and monitoring hardware systems with typical e-learning software capabilities. The system is an open platform designed to provide an easy management for teachers and ease of use for students.

2. SYSTEM DESCRIPTION

The proposed LABCENTER web system offers several advantages to teachers and students. For teachers, the system is conceived to provide the tools required to manage laboratory sessions for both lab session configuration (further browsed by the student) and information collection from student's activity. For students, the system offers a general web environment where specific laboratory activities must be previously defined by the teacher, thus being as operative as the teacher defines for every laboratory session. Additionally, one of the most important features that LABCENTER includes is the ability to define the so-called “plug-ins”, consisting of individual functional modules able to exchange information between the web system and any external hardware system. These plug-ins must be programmed by a web system programmer according to the desired external function and hardware device properties, but once defined, the teacher can use it for the lab proposal definition in conjunction with the related hardware. LABCENTER is thought as an additional tool for students to use hardware systems not easily accessible; teachers must design carefully a course using this tool to support traditional learning processes.

The design of the LABCENTER web system involves several programming tools in order to adapt to all different parts of the system. The first part consists on a pure web programming system, using Apache, PHP and MySQL as the main software. These programs are used to program all administrative tasks. The second part is related to the hardware connection, i.e. the exchange of information between hardware systems and the web system. In this case, it may depend greatly on the target hardware, but the most used approach consists on developing custom libraries (in Java or C/C++) that are to be called from the web server. The libraries include high-level functions to communicate with the web server, allowing controlling specific hardware.

The system is divided in three main blocks. First the administration site, where the web administrator can create new courses. Second the teacher website, where teachers can access the courses and customize the lab sessions offered to students. Finally, the student website, where students access to authorized courses and get access to contents previously defined by the teacher.

2.1 Administration and teacher web environments

The access to the administration site is different that those for students, and is only permitted to the administrator and teachers. The administrator has all the tools and functions available. The teacher’s website is the same as the administrator one with the only difference that a teacher cannot create courses. The administrator is in charge of creating courses and grant access to teachers.

The functionality provided by LABSERVER for the administrator is the following:
- **User:** This option allows enrolling users in a certain course. The administrator is using this option to enroll teachers in a course, and later, those teachers can use this option to enroll students in the course. When enrolling, certain options such as username, password, email contact, etc. must be filled.

- **Course:** This option is exclusive for the administrator to create or delete courses (Fig. 1). When creating a course, several teachers can be added.

- **Practice:** This option allows a teacher to create new laboratory sessions for a certain course (Fig. 2). This is one of the most important options in the system as this will create the student interface for remote laboratory practice. The flexibility in the definition of the laboratory sessions allows the teacher to define different proposals according to the complexity level or the skills that should be developed by the student. This task must be done carefully and the teacher must think the learning outcomes of every lab session in order to appropriately configure the web layout to be accessed to the student. Here, the teacher must fill all the information considered as adequate for the student to carry out the proposed task. Also, any type of files can be uploaded, and the most important characteristic making LABCENTER a novel proposal, is that the teacher can configure a control panel with different hardware links (plug-ins) so that the student can connect to the hardware system, interacting with it by means of these plug-ins. Fig. 3 shows the configuration screen, where the teacher has a list of available plug-ins (left side) and, on the right side, a rectangle imitating the student screen, where the teacher can drag and drop the plug-ins where desired. By doing this, the student control panel can be completely customized for each lab session. This possibility allows a teacher to precisely define the actions a student can make, allowing him to focus on different aspects, even accessing the same hardware.

- **Plug-ins:** This option allows defining how the web server will access a hardware system. Though teachers can define new plug-ins, typically, this is a task to be done by experienced programmers, requiring good knowledge of the hardware system and the web server. A plug-in can send data or order actions to the hardware system according to the data introduced in a web page. On the other hand, a plug-in can collect data from the hardware system and show it in a web page for monitoring purposes. The plug-ins might have the appearance of a button, a color LED, a text box, an image, a scope, etc. Fig. 4 shows a list with some of the defined plug-ins. This is one of the most important parts of the system since plug-ins are responsible for laboratory equipment communication. Nowadays, plug-ins must be defined as Java applets, Perl, C++, or any language able to communicate the web server system with external equipment.

- **Settings:** This option defines the general options for the entire web system and is accessible only to the administrator. The main banner image can be defined, as well as some text and information shown in the main access page for the students.

- **Statistics:** This part is a very useful tool for a teacher to follow the student’s achievements. Here, a teacher can apply different filters to know the time when different laboratory sessions have been used, which laboratory sessions has done a student, and how long he was connected to each session. In general, this option might help to know more about students and the proposed sessions to achieve the course goals.

![Fig. 1. Course management configuration screen.](image1)

![Fig. 2. Configuration screen for lab sessions management.](image2)

![Fig. 3. Configuration screen for the layout editor in the hardware access.](image3)

Depending on the hardware system used, the laboratory activity might include an auto evaluation system. This allows the student to obtain immediate response from the web system about the successful completion of the proposed task. This is very important in order to provide immediate feedback to the student so that they do not feel lost when something is not working properly. Auto evaluation is a feature that is under development at present to increase the hardware systems including auto evaluation. This is done by knowing exactly the goals for every lab session, thus, the system is revising...
automatically how the student is performing the tasks and can inform him automatically. Moreover, this can be of use to the teacher to know more about the student’s performance.

![Configuration screen for plug-in management](image1)

**Fig. 4.** Configuration screen for plug-in management.

### 2.2 Student’s web environment

After validating the username and password in the main page of the student website, the system offers a link table with all the courses where the student is enrolled (Fig. 5). Because the remote laboratory requires the exclusive usage of hardware resources, there exists a first come / first serve queuing system, although a booking system is also available. The main screen in the system also shows if a remote laboratory is currently in use (red status); in that case, the student can use the “Reserv” option to allocate a time and date for the laboratory usage, if the laboratory is available (green status), the student can enter the remote laboratory immediately.

Once the student has entered a specific laboratory, all the lab sessions are offered and a time limit is set. Time limitation is created so that a single user cannot use the system during unlimited time, forcing a time sharing among users. At this stage (Fig. 6), the student can browse all the information the teacher made available for him and go ahead with the remote laboratory execution by clicking in the lab session number he intends to perform.

![Main screen for student’s website](image2)

**Fig. 5.** Main screen for student’s website.

Once the student enters one of the remote laboratories, the screen will show all the monitoring and controls defined by the teacher. Depending on the lab session and course, the student can send and receive data from the hardware system, or even send a program or a movement sequence in order to learn how certain hardware systems work.

![Main screen for the lab sessions in a course](image3)

**Fig. 6.** Main screen for the lab sessions in a course, information files and a link to the hardware are available.

### 3. EXAMPLES

The proposed LABCENTER web system is conceived as a generic web environment, being the framework for any remote laboratory activity that could be defined through the plug-ins. This section shows two examples of remote laboratories built in LABCENTER. General controls such as an IP camera image view plug-in can be used in a hardware system in order to have real time monitoring of the devices.

#### 3.1 ABB Industrial robot programming using RAPID

Specific libraries have been developed to communicate with an ABB® industrial robot and a Siemens PLC, through which several plug-ins have been created to control the robot. In this case, developed plug-ins are buttons to control the robot activity and monitoring text boxes showing different information according to the robot state; additionally, a plug-in allows to upload a RAPID program to the robot and execute it, viewing the results in real time. In this case, plug-ins have been developed using Java applets and specific commands for communicating to the PLC and the robot controller web server.

This proposal allows the student to perform up to 10 different laboratory sessions remotely, learning to use the robot and their RAPID programming language. Fig. 7 shows the screen for robot control and monitoring in one of the proposed lab sessions. In this figure, Start and Stop buttons are two Java Applets sending a bit value to the PLC where it is considered as an input. The dialog for sending the program file is a plug-in to communicate via FTP with the ABB IRC5 robot system.

#### 3.2 FPGA programming, control and monitoring.

A communication system with an FPGA has been developed, allowing downloading of precompiled programs and running different tests on a real board, activating different input signals and receiving the output information through the web interface. Doing this, the student can simulate and generate the FPGA program (according to the teacher requirements) using standard ISE
Webpack tool installed locally, after generating the .bit file, the verification is done by downloading the .bit file in the remote FPGA board and check that requirements are met, thus learning how a real hardware system works.

In this case, a Xilinx Spartan3 FPGA is used through a specific developed plug-in for FPGA programming from the web server without using Xilinx proprietary tools. One of the novel proposals of the system is the Asynchronous JavaScript and XML (AJAX) information exchange system and the ability to program the device with standard Joint Test Action Group – JTAG – (IEEE, 1990) tools, allowing to use the system as a base for programming and debugging other JTAG devices, with no need for commercial tools. Web pages created with AJAX allow high degree of interaction, while requiring no complex plug-ins like Adobe Flash or Java.

![Fig. 7. Example of a remote lab operation for an ABB industrial robot programming.](image)

4. CONCLUSIONS

This paper presents a new web platform that allows teachers to create custom remote laboratories for real hardware control and monitoring. The system helps the students to increase their knowledge by using real hardware systems, which are sometimes difficult to access in laboratories due to time restrictions.

The system intends to be a common framework for any remote laboratory activity. The essential tools for the management of users, courses, access control to hardware, etc. have been developed. Programming new plug-ins, LABCENTER can be used in any hardware system. Once the plug-ins are created, its usage is simple for both teachers and students, offering powerful control and monitoring options within an easy to use web environment. Addition of new plug-ins is open to new controls and easy to integrate in LABCENTER in order to accommodate any experimental setup to the web system.

ACKNOWLEDGMENTS

This work has been financially supported by DocenTIC research project number 94/DT/29, granted by UDIE (Unitat D’Innovacio Educativa) at the Univ. of Valencia.

REFERENCES

Autor, (2010), [http://www.autor.ca](http://www.autor.ca)
Dokeos, (2010), [http://www.dokeos.com](http://www.dokeos.com)
Ilias, (2010), [http://www.ilias.de](http://www.ilias.de)
Moodle, (2010), [http://moodle.org](http://moodle.org)