LaboRem: open lab for remote work

Franck Luthon & Anca Petre
Department of Industrial Engineering
University of Pau and Adour Province
IUT, Anglet, France
franck.luthon@univ-pau.fr

Dan Steriu & Andrei Besleaga
Faculty of Electronics
Polytechnical University of Bucharest
UPB, Bucharest, Romania
dansteriu@yahoo.com

Abstract—**The advances in communication networks together with the improvement of device interconnectivity and web technology make it possible nowadays to develop distant e-learning applications that encompass the entire learning process : not only lectures, tutorials and simulations, but also practical labwork on real or virtual instruments. Pros and cons of e-labs are listed in the literature, based on comparative studies between the three types of laboratories: hands-on, virtual and remote.**

In this paper, we present an open client-server architecture for remote instrumentation within a virtual lab. It is based on a plug-in technique for sharing technical resources (i.e. connection of new available experiments). We describe a prototype implementation for labwork at undergraduate level in signal processing, circuit measurement and system testing. Our contribution is twofold: a) a template plug-in technique, making it easy for tutors to design and insert new experiments within the lab framework (flexibility and extensibility); b) the use of a game-like pedagogical approach: the distant labwork is implemented as a game sequence with various steps.

Finally, this project is intended for being used by foreign universities that do not have at disposal all the expensive equipment required for measurement and testing, but only have access to the network. We hope this will contribute to the "democracy of knowledge".

Index Terms—**Remote laboratory, virtual laboratory, distant learning, labwork, e-learning, simulation laboratory, LabVIEW, virtual instrument, VI.**

I. INTRODUCTION

Pros and cons of e-labs are extensively discussed in the literature. There are many comparative studies on virtual versus remote labs, real versus simulation work, regular versus rich media interface, distant versus local work, hands-on versus 3D virtual reality etc. [1]. The three main advantages of virtual/remote labs are listed below: they impact altogether money, space and time.

- Money: the first outcome is cutting the costs of investments in instrumentation hardware. Indeed, a single piece of device may be used by many clients or students at an international scale. Maintenance costs can also be shared by mutualization, i.e. mutual use of a single hardware by different universities (technological sharing).
- Space: the labwork can be completed from any place where the world wide web is available. The distant client communicates with the server through TCP Internet Protocol. Students can work at home, or in any room equipped with PCs at the university. Non-affiliated students may also run the application, after authentication from their own PC.

- Time: clients can work anytime since the lab-server is active 24h/24 and 7d/7. International partners (as other universities) might also book some temporal slices for their students to complete the labwork that requires the use of instrumentation hardware located at our Institute. More time may be allocated for people that need it, *e.g.,* disabled people or low-level students. This is not possible with hands-on activities in a real lab: the job must be done within the 3 to 4 hours devoted for the lab session in the dedicated room.

These are three key-points for releasing the heavy constraints induced by classical hands-on labwork activities [2]. Starting from those statements, our objectives are the following:

First, international cooperation with foreign universities is one important assignment of our Technological University Institute (IUT) in Bayonne-Pays Basque, Anglet, France. Our remote learning platform is especially suited for universities in developing countries that only have access to the network (web-browser) with very few (or very old) instruments available locally. This is a step further towards the "democracy of technical knowledge".

Secondly, the two-year course in electronics, chosen as a first case study for developing our platform prototype, is part of a curriculum for technicians at undergraduate level in the Department of Industrial Engineering at our university. There are 25 other Industrial Engineering Departments in French universities and we want to share our experience and resources with them (this corresponds to more than 1000 students that are potential clients of our platform). Indeed, moving towards this kind of collaborative pratice is of vital importance to cope with the shortage of university financial means.

Third, we want to make the **labwork as a game**: based on the fact that young people like e-games, that they are slow at work but good on web, we adopt a pedagogical approach that is game-based learning (GBL). The remote labwork session is built as a game sequence with various steps, including multiple-choice questionnaire, simulation, remote control, measurement, virtual lab. Each client, i.e. student, is allowed an amount of time, can repeat some steps if desired, or proceed further to reach the objective of the work. We will see in section IV that the lab sequence is implemented as a test sequence, where the student is the device under test (D.U.T).

Finally, another objective is to offer a way to help tutors to develop themselves easily new experiments thanks to a

template plug-in technique for experiments. Previous work in [3] dealt only with the proposal of architecture and plug-ins, without presenting labwork sequence implementation.

The paper is organised as follows. In section II, we focus on recent results available in literature, and point out some important features of e-labs. In section III-A, we present the architecture of our remote lab called LaboRem; a prototype implementation is described in III-B. Section IV-A presents the game-based learning approach and lab sequence design; section IV-B illustrates some labwork sequences dedicated to courses at undergraduate level in signal, circuits and systems measurement and testing. Results with statistical evaluation are given in section V. Finally, discussion in last section opens the field to remaining questions that still have to be worked-out.

II. RELATED WORKS

Design criteria for distant virtual labs are discussed in [4]. An implementation with LabVIEW and GPIB is given. Both authors in [4], [5] focus on complementarities of the three types of labwork : **hands-on, virtual, and remote labs** (this last one meaning taking distant control on real-world instrumentation). The interest of timeless labwork is especially important for disabled people [5], [6], who need more time for doing a job, or need to repeat once or twice before proceeding further in the labwork sequence.

Even if virtual labs often use Matlab [7], many remote labs (RL) are based on LabVIEW and GPIB: [2] presents a client-server application with two separate servers: a web server and an experiment server. They use ActiveX control embedded in html pages. Simultaneous connection of 8 clients is possible. In [8], a LabVIEW web server is used for a remote lab in optical circuits. The strategy includes both simulation and remote experiments. The NetLab described in [9] is another implementation based on LabVIEW and GPIB. A webcam and a chat window for communication between users contribute to the richness of media. Addition of audio communication is also discussed, especially for collaborative work. Authors put emphasis on the advantage given by remote labs for repeating experiments: students that need to repeat a task can do it. But a key-issue is the accessibility of the RL : the booking of sessions should be easy. Another important point is that students like to have some degree of freedom : they like to wire themselves the circuits to test.

Some works try to evaluate the effectiveness of remote labs. Authors in [10] present a pedagogical experiment in mechanics (study of a mechanical vibration system). They address the rich media issue and conclude that richness of media does not matter so much, due to adaptation behaviour of humans. [11] gives some metrics (Goal Question Metric) for the measure of effectiveness for a remote lab prototype ERRL in radio electronics. The implementation is based on a Learning Management System (LMS) and the use of Java applets. Authors in [12] present a Remote Lab Generator (RLGen) to help lab instructors to prepare remote lab.

The main technologies that can be used for RL are listed in [13]: download of software (6%); applets (58%); LMS (36%).

They propose an evaluation of RL based on seven criteria and compare 5 different laboratories. Worth mentioning is that they are working on game-like usage of the remote lab.

Major limitations of RL are discussed hereafter. One drawback is that the activity is not done with direct communication, each client is working alone in a virtual community (other students, tutors, administrator, technicians, webmaster). This raises the questions of rich media and of synchronous versus asynchronous communication for collaboration. The universal virtual lab for introduction to electrical circuits [6] is based on the exchange of text files since it is found to be a reliable method to function as closely as possible to real-time.

Another concern is the prerequisites and a bad preparation of remote session. Since there is no tutor physically present to deliver some complementary information or debug the system, one must be sure that : a) there is nothing important missing in the data package delivered to the student (CDRom or zip file downloaded) ; and b) the application might never end to a locked state because of the hardware devices in improper configuration. One important remark correlated to this issue is that there is also a strong need for maintaining network and lab server available 24h/24.

Of course, the pedagogical efficiency remains the most important criterion, and must be evaluated properly for assessing a labwork quality. Evaluation of student's activity in virtual environment is therefore a key issue [14].

To conclude this overview, we want to put emphasis on two key-points that we try to address in this paper: there is a strong need both for helping tutors to develop other experiments by themselves and for entertaining youngsters involved in technical curricula. Therefore, we will focus on two main features of our work: the template plug-in technique, and the game-like pedagogical approach.

III. REMOTE LAB PLATFORM

A. Open Architecture

The software/hardware architecture is described in Fig. 1. It is a client-server architecture with two logical servers: one for the controllable experiments, the other one for web users. Depending on local network configuration, the two servers could either be physically separate, or shared on the same hardware server. Clients are remote students that can connect via intranet or internet (TCP/IP protocol).

The hardware that can be remotely controlled includes instruments for measurement and testing, electronic protoboard, DAQ boards, switching matrix for power on/off and for configuration and connection of device under test. Various software programs can be used as: LabVIEW, Multisim, PSpice, Matlab, TestStand, etc. The client communication channel uses both an asynchronous mode (upload-download of data files) plus a synchronous mode for webcam.

The main feature of this architecture is the plug-in concept implemented in LabVIEW as simple or complex virtual instruments (VI programs) either for simulation or real-world experiment control. As regards plug-ins, templates are available that help any authorized tutor to develop new experiments or
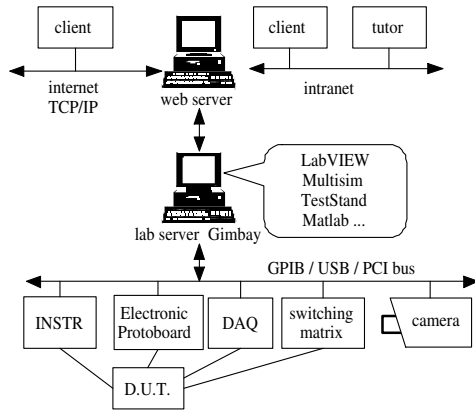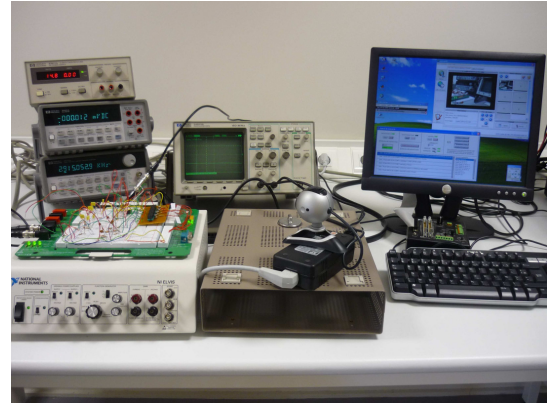
Fig. 1.   LaboRem Architecture

Fig. 3.   Instrumentation hardware on server side

media to be integrated into the platform. The server-side plug-ins are complemented by client-side plug-ins, easily distributed to client application and automatically integrated inside the application menu (Fig. 2).
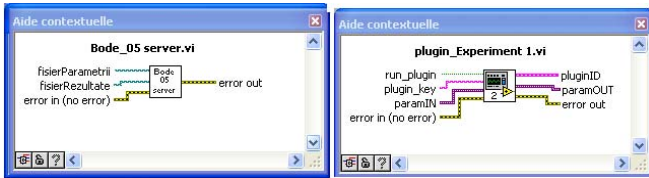

Fig. 2.   Plug-in for experiment connection: a) Example of server-side plug-in for Bode plot; b) Template plug-in on client-side

B. Prototype Implementation

1) Lab Server Gimbay:

a) Hardware: The lab server Gimbay is a PC located in our department that can take control of various hardware devices: NI-DAQ acquisition boards (PCI or USB), GPIB devices (oscilloscope, waveform generator, power supply, multimeter), protoboards (like NI-Elvis) for wiring circuits, switching matrix with relays for powering and configuring the system (Fig. 3).

b) Software: The lab server is developed with LabVIEW and Internet Toolkit library. The web communication is handled by the G Web server through CGI scripts. At the server side, the control center (Fig. 4) is a human machine interface (HMI) for management of available experiments list, of user access list, of user sessions and requested experiments in queue. It permits the modification of the lists (of users, of experiments) and configuration changes by a server administrator either locally or remotely via Remote Panels control. The server also implements a watchdog routine which checks for proper use and running of server programs (web server and experiment server applications) and restarts them in case of failure. Log files are available for checking detailed usage.

c) Network Configuration: Since the LabVIEW server belongs to a private VLAN not visible from outside, the technical solution currently implemented for internet connections
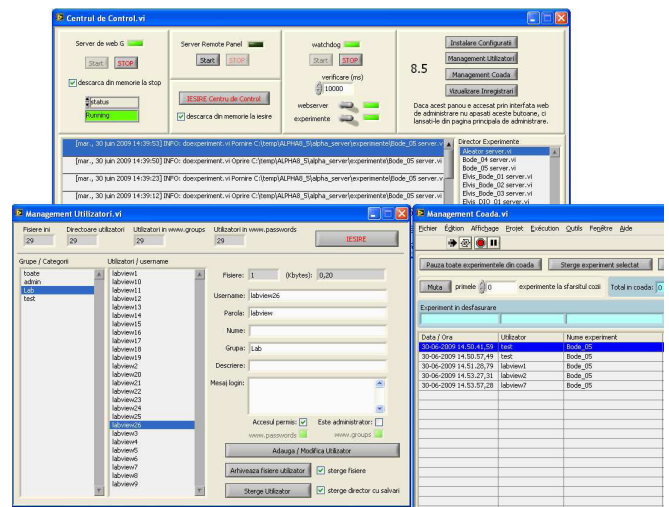

Fig. 4.   Control center on server side: top) web server and watchdog start; bottom) configuration and management of users and experiments in queue.

to the lab, is a reverse-proxy cache service based on Apache HTTP server. Two virtual hosts are declared on two special TCP ports: one is to access the LabVIEW server, the other one is dedicated to the webcam connection. Of course, web technology will be changed for future deployment of the lab over internet: we intend to make use of a VPN.

2) Client Application LaboRem:

a) Deployment: Deployment is done either by downloading an installer from our website[1], or by sending a CDRom to the client, with all the binaries and docs needed.

b) Human Machine Interface: After authentication, the client gets access to the interface shown in Fig. 5. From this interface, he can conduct a lab sequence and have remote control on instruments. After downloading results, a graphical user interface exhibits the requested plots for interpretation. Finally, an evaluation form is electronically completed for scoring the session.

c) Asynchronous Communication: Client communicates by upload-download of parameter/data files. This is a robust

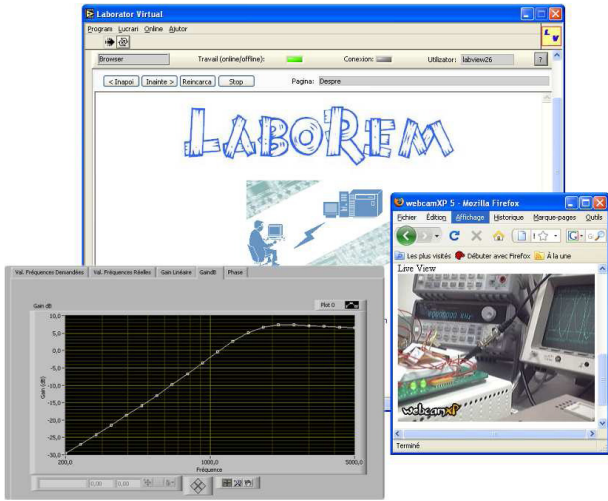[1] http://www.iutbayonne.univ-pau.fr/ luthon/

Fig. 5. Remote laboratory client interface with camera-feedback and example of Bode plot result

communication mode that allows many users to work simultaneously: user requests with uploaded parameter files are stored in a FIFO queue and served one after the other. Since a typical manipulation (*e.g.* Bode plots of active filters) lasts only a few minutes, this proves to be efficient for our purpose of multi-user connections.

Note that a synchronous communication system could be used as well very simply, in case one needs real-time control of experiments, via plug-ins developed as clients and communicating directly with the corresponding server-side VI plug-in experiments through TCP/IP/datasocket/remote panel/web or any other available internet technologies, separate from the application upload/download request/result system, but part of the application user/queue system.

In addition, a webcam service (synchronous communication) is available for visual control of remote instrumentation steps, in order for the distant student to know what happens in the hardware environment.

IV. LAB SEQUENCE

A. Sequence Design

The lab sequence is designed as a game including several levels that are reachable depending on students score (Fig.6). This application takes from e-games the following concepts: increasing difficulty level, limited number of lives, limited amount of time, and score. The lab sequence is implemented as a test sequence with TestStand software from NI.

- Levels: the application is built like a game, containing several levels that have to be passed over during the labwork. Level 0 is a quiz which tests student's theoretical background (prerequisites). If the score of this quiz is not satisfying, the next level can not be reached. Levels 1,2,3 all have to be done for passing to the next one. Throughout these levels, clients can execute several experiments like remote measurements, simulations with MultiSim, or hands-on labwork. It is considered that these

steps are important for acquiring the essential ideas of the subject under study. Next levels 4,5,6 are also based on manipulation and simulation activities, but with an increasing difficulty of the subject. Every time a level is not passed over, students have to fill in a questionnaire about the application (section V-B) before exiting.

- Lives: several attempts (or lives) are allowed in order to succeed in the current level. In the case of our LaboRem application, the number of lives is limited to three.
- Time: for each level an amount of time is allowed. The client has to finish on time, otherwise the game is over (Fig. 8b). In that case, the client can try again if he still has some attempts left, or he has to exit the application.
- Score: depending on the level reached and on the accuracy of student's results and answers, a score (or mark) is given as shown in Fig. 6.
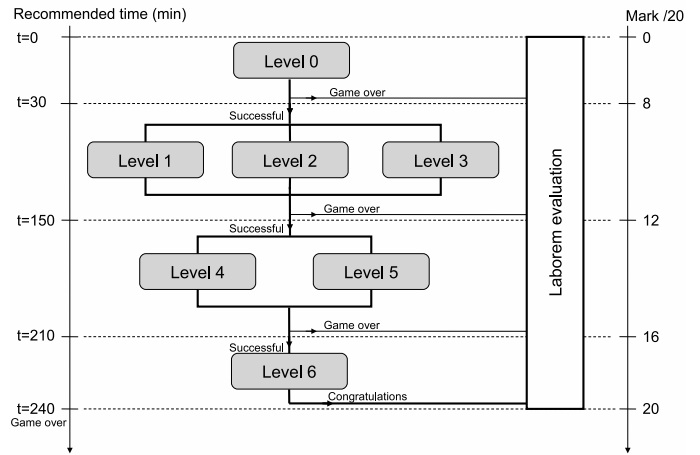

Fig. 6. Game like diagram

Each level is based on a typical sequence, as shown in Fig.7. The sequence begins by the user's authentication, then the
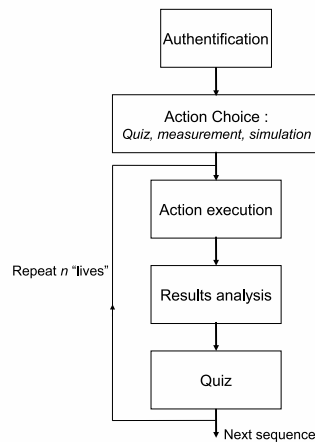

Fig. 7. Sequence steps

choice of the next action is proposed. This action depends on the user's level and can be simulation, measurements or hands-on work. The client has to choose the right parameters for the selected action and then to upload these data onto the server. Once the experiment is finished, the user can download the results and then proceed to a detailed analysis of the results. A quiz is proposed at the end of each sequence in order to be sure that the student has understood the main ideas of that work. In case of failure, the client has the opportunity to repeat the sequence, but no more than 3 times.

B. Examples of LaboRem Sequences

1) LaboRem for Signal Processing: LaboRem application was first developed for a case study on active filter teaching. Within the national pedagogical program of IUTs, the active filters module consists of 40 teaching hours, including lectures, tutorials and laboratory work sessions. This topic was chosen because of its typical needs in the remote control environment (oscilloscope and waveform generator must be controlled for Bode plots and various OpAmp circuits must be selected for choosing the filters) and due to our wish to end up with a complete remote solution. From now on, only the labwork part is remotely available (but not yet all the lectures and tutorials). This distant labwork contains several sequences, as described previously and which we detail below: hands-on, quiz, remote control, Multisim simulation and final evaluation form.

- Hands-on step means manipulating real-world hardware equipment. It should cover two situations: a) in universities with equipment (even if old), the students conduct manually their measurements and then enter the data results into LaboREM for further interpretation; b) in universities without any equipment (apart from PCs with network connection) or in the case of an isolated student, the hands-on step can be switched to simulation step, *i.e.* processing of synthetic data generated by virtual instruments (LabVIEW VIs) since the student has no local access to any measurement device.
- Quiz and questionnaire steps are used for testing student's abilities in filtering and in OpAmp circuitery (Fig. 8a).
- Remote control step: eight different active filters (like Sallen-Key etc.) are installed on an Elvis platform (Educational Laboratory Virtual Instrumentation Suite) from National Instruments. The client, via the remote control, can choose a filter and characterize it (Bode plots). At the beginning, both frequency range, input voltage and number of measurement points have to be chosen. These parameters are written into a file which is uploaded to the server. As explained in section IV-A, the obtained results are then downloaded onto the local computer of the client for analysis and interpretation.
- Simulation step : during this step, using MultiSim software, the client should design the electrical circuit corresponding to the same filter as the one he characterized during the remote step. After determining the main characteristics of the filter, the user must compare simulation results with experimental ones.
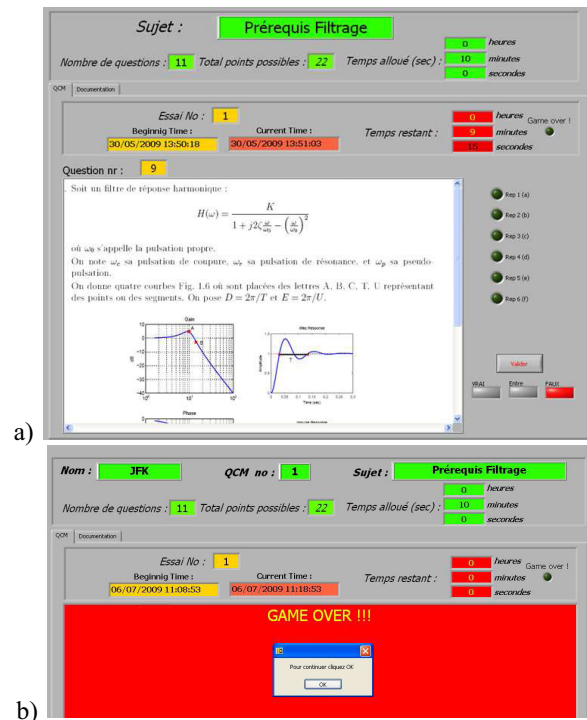

Fig. 8. a) HMI with a quiz on filtering for student scoring ; b) Game over

2) LaboRem for Circuits, Systems and Data Acquisition: LaboRem may also be used for circuits and components characterization or for system testing (see [3] for details). Some available experiments are listed below:

- For characterization of MOS transistors, three multimeters and two power supplies connected to GPIB are required.
- For characterization of an audio LF power amplifier, two multimeters and a waveform generator are used.
- We also developed a plug-in to control signal acquisition via a DAQ board, and do spectrum analysis and digital filtering of the acquired signals.

V. EVALUATION OF RESULTS

A. Network availibility

In intranet mode, 30 simultaneous connections of clients were tested successfully. As regards internet connection, a test was conducted last winter with Bucharest University, thus validating the prototype for our objective of international cooperation.

B. Quality of LaboREM

Evaluation was done in June 2009 on a bench of 30 first-year students. After completing the lab session, they had to fill-in a questionnaire about the application. It uses a 5-point Likert-type scale (1=strongly agree, 2=agree, 3=neutral, 4=disagree, 5=strongly disagree). A sample of 6 of the overall 69 questions is given in Tab. I. (with the average scores written in bold). The statistical results are shown in Tab. II, where all the answers to the questionnaire are synthetized in 10 main issues and where mean-value and standard deviation of

TABLE I
QUESTIONNAIRE: EXAMPLE OF QUESTIONS

I liked to do labwork from anywhere and anytime	**1**	**2**	3	4	5
I liked to control remotely real instruments	1	**2**	3	4	5
I was motivated by the game-like lab sequence	1	**2**	**3**	4	5
I need more preparation to LaboREM	1	**2**	**3**	4	5
I liked to repeat some lab steps	**1**	**2**	3	4	5
I need a remote control of the camera	**1**	**2**	3	4	5

students answers are given. From those first results, we see that

TABLE II
STUDENTS ANSWERS (WITH MARKS RANGING FROM 1 TO 5)

Evaluated issue	Mean	Std.dev	assessment
Quality of interaction with HMI	2.3	1.96	+
Technical quality of platform	2.9	2.67	±
Interest of game concept	2.3	2.16	+
Pedagogical efficiency	2.1	2.03	+
Preparation, Docs & Prerequisites	3	2.67	±
Interest for remote control of devices	2.2	3.57	+
Release from space-time constraints	1.9	2.9	++
Collaborative work	3.0	2.93	±
Autonomy demand	2.2	1.62	+
Overall satisfaction	2.46	1.78	+

we still have to improve the technical quality of the application, and to deliver more documentation and preparation. Moreover, students need more collaborative work. As regards good points, they like the game concept, the HMI is O.K. (especially with the video feedback that students enjoy) and they like to take remote control on hardware devices. But above all, they do appreciate to be able to do labwork outside the classical spatio-temporal constraints. Of course, more evaluation is needed to assess this work, both as regards technical and pedagogical efficiency. Next year, the 25 Industrial Engineering departments in France will test our prototype (about 1000 students), so that we should have statistical meaningful data to process for quantitative evaluation.

VI. DISCUSSION AND FUTURE WORK

In addition to the pratical training for technicians in electronics or automation, the LaboRem prototype can be extended to the training of engineers in the field of circuit design thanks to Multisim, or signal processing design thanks to Matlab/Simulink. We also intend to develop plug-ins for mechanical engineering (vibration analysis in a damping system). LaboRem platform could also be used for applied research activities, in the case of dispersed distant research laboratories.

Although NI TestStand engine is well suited for building lab sequences with various steps calling different applications for either simulation, remote control or HMI with quizzes (written in various programming languages like C/C++, LabVIEW, Matlab, Multisim, html, CVI, ActiveX or Java), we intend to make use of an LMS (learning management system like Moodle or Dokeos) for easier handling of user sessions (authentication, lab activity management, evaluation). This will allow collaborative work through a richer media interface (chat, audiovisual communication, etc.) As regards the HMI and the rich media issue, the asynchronous communication (upload-download of files) will be complemented with the use of a user-controlled motorized camera (synchronous communication channel).

As regards network technology, we will investigate further the best choice to make. Our prototype based on a reverse proxy technique should evolve towards a more secure solution : a VPN will be configured for deploying the application on the net. Another key-point to address is the mutli-server architecture for sharing LabVIEW virtual experiments from all over the net.

REFERENCES

[1] H. Benmohamed, A. Leleve, and P. Prevot, "Remote laboratories: new technology and standard based architecture," in *Int. Conf. on Information and Communication Technologies: From theory to applications*, Damas, Syrian Arabic Republic, 2004.

[2] I. Gustavsson, T. Olsson, H. Aakesson, J. Zackrisson, and L. Haakansson, "A remote electronics laboratory for physical experiments using virtual breadboards," in *Proc. of the 2005 ASEE Annual Conf. & Exposition*. American Society for Engineering Education, 2005.

[3] M. Steriu and F. Luthon, "Open architecture for signal processing lab distance learning," in *4th IEEE Digital Signal Processing Education Workshop (DSP'06)*, Jackson Lake Lodge, Wyoming,USA, Sep. 24-27 2006, pp. 305–310.

[4] D. Grimaldi and S. Rapuano, "Hardware and software to design virtual laboratory for education in instrumentation and measurement," *Measurement*, 2008.

[5] Z. Dvir, "Web-based remote digital signal processing (DSP) laboratory using the Integrated Learning Methodology (ILM)." IEEE, 2003.

[6] M. Duarte, B. Butz, S. Miller, and A. Mahalingam, "An intelligent universal virtual laboratory (UVL)," *IEEE Trans. on Education*, vol. 51, no. 1, pp. 2–9, Feb. 2008.

[7] R. Dormido, H. Vargas, N. Duro, J. Sanchez, S. Dormido-Canto, G. Farias, F. Esquembre, and S. Dormido, "Development of a web-based control laboratory for automation technicians: the three-tank system," *IEEE Trans. on Education*, vol. 51, no. 1, pp. 35–44, Feb. 2008.

[8] D. Gurkan, A. Mickelson, and D. Benhaddou, "Remote laboratories for optical circuits," *IEEE Trans. on Education*, vol. 51, no. 1, pp. 53–60, Feb. 2008.

[9] J. Machotka and Z. Nedic, "The remote laboratory NetLab for teaching engineering courses," *Global J. of Engng. Educ*, vol. 10, no. 2, pp. 205–212, 2006.

[10] J. V. Nickerson, J. E. Corter, S. K. Esche, and C. Chassapis, "A model for evaluating the effectiveness of remote engineering laboratories and simulations in education," *Computers & Education*, vol. 49, pp. 708–725, 2007.

[11] G. Tokdemir and S. Bilgen, "Remote lab effectiveness assessment model," in *Optimization of Electrical and Electronic Equipment (OPTIM'08)*. IEEE, 2008.

[12] H. Hasnin and M. Z. Abdullah, "Remote Lab Generator (RLGen): a Software Tool Using Auto-Generating Technique to Develop a Remote Lab," *iJOE*, vol. 3, no. 4, 2007.

[13] H.-D. Wuttke, K. Henke, and N. Ludwig, "Remote labs versus virtual labs for teaching digital system design," in *Int. Conf. on Computer Systems and Technologies (CompSysTech'05)*, 2005, pp. IV.2–1;IV.2–6.

[14] J. Heh, J. Chang, S. Li, and M. Chang, "Providing students hints and detecting mistakes made by students in a virtual experiment environment," *IEEE Trans. on Education*, vol. 51, no. 1, pp. 61–68, Feb. 2008.