Development of a Mobile Scheduling System for Interactive iLabs

Cosmas Mwikirize
e-mail: bmwikirize@tech.mak.ac.ug
Davis Agaba
e-mail: agabadave@tech.mak.ac.ug
Sharlotte Andinda
e-mail: sandinda@tech.mak.ac.ug
Paul Isaac Musasizi
e-mail: pim@tech.mak.ac.ug
Department of Electrical and Computer Engineering
Makerere University
Kampala, Uganda

Phil Bailey
Massachusetts Institute of Technology
Boston, Massachusetts, USA
e-mail: pbailey@mit.edu

Abstract- **Makerere University has for the past seven years used internet-based laboratories or iLabs to support practical courses in the curricula of the undergraduate electrical, telecommunications and computer engineering programmes. The iLabs have provided a meaningful laboratory experience to students in the face of increasing numbers, insufficient lab equipment, space and lab personnel. One version of the iLabs, the interactive type, requires pre-scheduling so that a user is granted exclusive access to the experiment hardware, a procedure previously conducted online. The online scheduling is inconvenient since it requires a lot of time investment for the users before conducting the lab. The paper presents the work undertaken in development of a Short Message Service application to facilitate the scheduling process. Starting with an overview of iLabs and the requisite scheduling procedures, the rationale for development of the Short Message Service application for the interactive version of the iLabs is explained. A technical overview of the application is given, including the requirements specification, the system design and the implementation. Finally, a demonstration of the application, error handling mechanisms and benefits of utilizing the service are presented. The application has provided a convenient scheduling system for interactive iLabs, thereby improving the efficiency of access to the lab.**

Keywords- SMS, interactive iLabs, iLab Scheduling

I. INTRODUCTION

In Africa, Universities and other Institutions of Higher learning are challenged when it comes to provision of relevant practical and technical training because of inadequacies in equipment, space and personnel. The equipment, if available, is in most cases obsolete. To get around the impasse, Makerere University has adopted online teaching and learning methods. Thus, although hands-on laboratories have traditionally been argued to provide the best avenue for experimentation, Makerere University has greatly utilized internet laboratories, also known as iLabs. With iLabs, users are able to access, interact with and control equipment remotely over the internet (or intranet), with the support of the relevant hardware, suitable graphical user interfaces, protocols and software. All a user requires is a computer that can connect to host, which is connected to the laboratory hardware.

The idea of iLabs was developed at Massachusetts Institute of Technology (MIT) by Prof. Jesus del Alamo in 1998, and introduced to Makerere University in 2004. Since then, the

iLabs Project at Makerere University has developed several laboratories to support different fields in the curricula of the Department of Electrical and Computer Engineering [1],[2].

MIT initially developed a generic web service infrastructure, known as the iLabs Shared Architecture (ISA) to support access to a wide variety of online laboratories. The package offers generic services like user registration and authentication and laboratory administration that are independent of any particular experiment domain. The lab provider customizes the ISA to host particular laboratories through establishing connections to suitable hardware, development of user interfaces and other intermediate services requisite to deliver the laboratory experience remotely through web services.

The iLabs developed at Makerere University fall into two categories, the Batched and the Interactive iLabs. Batched experiments are those where the entire course of the experiment is specified way before the experiment begins whereas Interactive experiments are those where the user monitors and controls one or more aspects of the experiment during execution [3]. Although they provide a more enriching experience for users, interactive labs are more complex to administer.

Unlike Batched iLabs where user requests are queued and de-queued on a First In First Out (FIFO) basis, interactive labs require pre-scheduling. This therefore calls for a scheduling policy whereby a student schedules for a lab prior to conducting it. The available scheduling system hitherto utilized was an online one, under the control of the User Scheduling Service (USS). With the USS application, students are required to a specific time for executing the due experiment. Figure 1 shows the USS interface.

II. RESEARCH CONTEXT

Although e-learning has become popular today, its effectiveness highly depends on the ease of access to resources deployed, especially in instances where time-sharing is inevitable. Mobile phones applications have become a popular tool worldwide in facilitating e-learning. Access to curricula material in form of video lectures, e-notes and lab resources can all be controlled with mobile phones [4]. In fact, some online laboratory technologies have a Short Message Service (SMS) service for user registration, scheduling of time slots and notification [5].

Figure 1. USS interface

The implementation strategies for the online laboratories differ and no single unified architecture exists, although various models have been suggested [6], [7]. Thus, integration of services like SMS support for scheduling or notification is also specific to the platform and technology used. Makerere University adopted the MIT ISA, which had no inherent support for SMS [8],[9].

The recent surge in student numbers at Makerere University has necessitated great efficiency in scheduling interactive laboratories. For this to be effected, students need to make timely reservations during the time when a particular laboratory is deployed. As mentioned before, this was only possible by way of online computers. It is known that the access mode to the lab has a huge bearing on the learning outcomes for students [10]. We conducted a survey, which revealed that 60% of students in the College of Engineering, Design, Art and Technology (CEDAT), who were targeted by iLabs do not own personal computers. Of the 40% who did, less than half had more than two hour access to the internet on a day to day basis. On the other hand, 99% of the students own mobile phones. Leveraging some of the popular applications on mobile phones for the purpose of scheduling laboratories would thus be a silver bullet, with SMS at the forefront. The idea for the application was henceforth born out of the need to provide a flexible platform for scheduling interactive laboratories, using Global System for Mobile Communications (GSM) enabled phones. This would in turn alleviate the inconveniences and inefficiencies that used to arise while utilizing the online scheduling system.

III. SYTEM OVERVIEW

Generally, the development of mobile scheduling system involves an integration of services whose major objective is to ensure automated timed reservations and administration for the service provider to the clients. Services provided by the mobile scheduling system include; a mobile scheduling service for the clients, an administration panel for the service provider to set time blocks and view reservations made and ability for the

client to cancel an already made reservation. The design of the system followed the traditional cycle of requirements specification, system design and implementation.

The Requirements Specification handled three major concerns; the functional requirements, non-functional requirements and integration requirements with the ISA. Figure 2 shows the use case diagram for the application, detailing how a typical user interacts with the system.

Various architectures were designed to depict how the application was to be developed, deployed and function. The functional architecture design portrays system behaviour in the event of a user request. The information architecture developed shows the information distribution among the various entities of the application, right from the user to the core of the application.
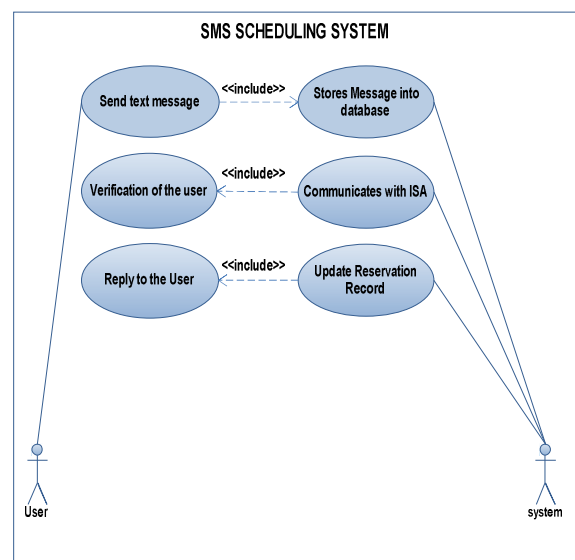

Figure 2. System use case analysis

The entities include; user (who is the actor), the SMS Gateway, the core application and the ISA, which contains the interactive iLab. Figure 3 shows a high level context diagram designed for the application, describing how it is meant to work with the ISA. The standard protocol for the application and the error handling procedures were also set up. These described the standard message to be sent by a typical user and the errors involved if the standard message was not followed. The standard message expected by the system is in the format:

"iLab Username Lab_Acronym Date_of_Schedule Start_Time_Range End_Time_Range". Table 1 shows the functions of each of the elements required in the scheduling process.
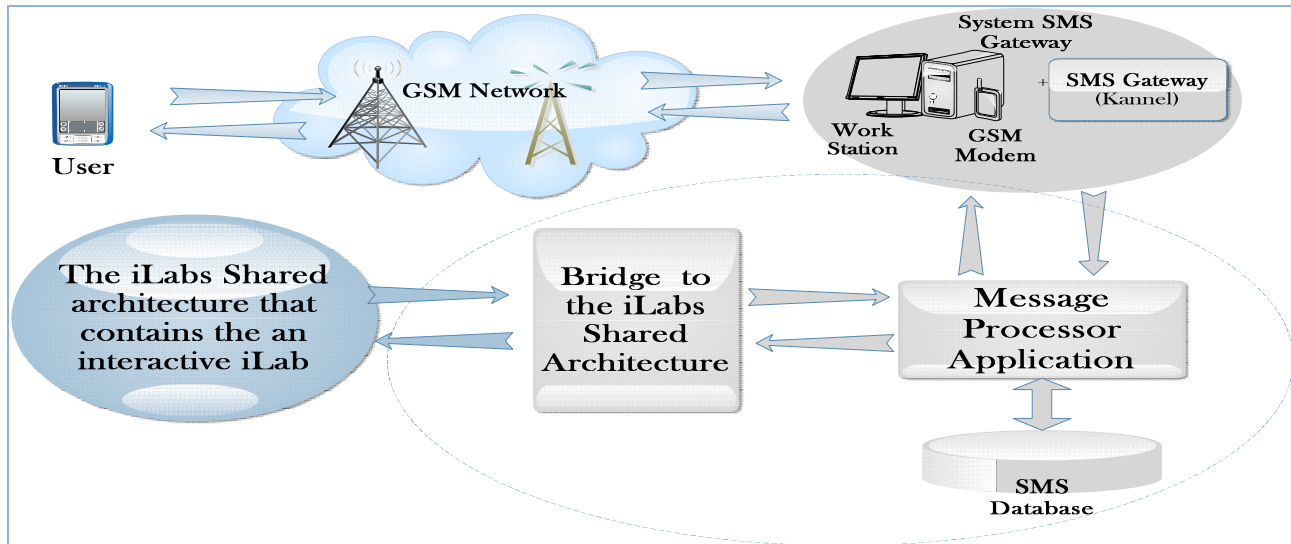

Figure 3. System context diagram

TABLE 1. MESSAGE PARAMETERS

Element value	Description and Function	Format
iLab	The default keyword that users make to identify the message for iLabs server application	Must be 'ilab' or 'ilabs'
Username	The username of the student as used by the student while registering with the iLabs Service Broker. Used to attach the scheduling session made with the username	Should be similar to the username used for registering the user, should have no spaces in between
Labname	The name of the lab the user is requesting to schedule and experiment for.	Should have no spaces in between, should be similar to the name specified by the administrator
Date	The date that the user wishes to do the lab. Specifies the date when the user wants to do the lab	The date should not have passed, should be of the format "DD/MM/YYYY where DD is date MM is Month and YYYY is year, the day, month and year should be separated by either '/', or '-', or '.', and should not contain any spaces
StartTimeRange	(Start Time Range) the time for the range within which the student would like the lab to be scheduled.	The Start Time given should not have passed, it should not be after the End time Range provided, it should be of the format "HH:MM" where the HH is the hour and MM is the minute, the hour and the minutes should be separated by either ':' or ';', and it should not have any spaces in between.
EndTimeRange	(End Time Range) this is an optional element that defines the user sets to give the end of the time range within which the lab can be scheduled.	The End Time given should not have passed, it should not be before the End time Range provided, it should be of the format "HH:MM" where the HH is the hour and MM is the minute, the hour and the minutes should be separated by either ':' or ';'it should not have any spaces in between.

The development Architecture describes the environment used for developing the SMS Scheduling application. Since the ISA was developed with the .NET framework and with SQL Enterprise Manager 2005, the SMS application had to be built with the same technologies to ensure compatibility. Kannel, an open source Linux based gateway was used for the system [11].

The deployment architecture describes how the application should be setup for use on a given system. The deployment of this system is implemented with two work Stations, which communicate via http Get and Post methods: a Linux based server that holds the SMS gateway, and the Windows Server, which hosts the ISA and the SMS Scheduling System. The deployment also involved the installation of the database files of the application and the configuration of the application before running the application. The iLab administrator is required to setup a virtual directory through the Internet Information Service (IIS) default web site. This deploys various web sites for setting up the Lab and the Web Service, which holds the SMS reservation application. The administrator then registers the SMS Reservation Application with the iLabs Service Broker and address the Lab to be scheduled, and the other credentials that users apply while scheduling into the application.

With the different architectures in place, the requisite code modules for the system were developed, and debugged individually and wholesomely to eliminate syntax and runtime errors. System tests were carried out to ensure proper functionality and stability.

IV. DEMONSTRATION OF THE SYSTEM

Consider a student, Dave, who would like to carry out an experiment on the 26th of May 2011 sometime between 12:00 to 14:00. He initiates the session by sending an SMS message to the iLabs System. The iLabs System has been setup with a modem, which contains a SIM card and SMS gateway (Kannel) to receive and send SMS messages although a Short Code could be used. The user has to follow the protocol or a particular format that has been described in the previous section. The test results for the message that Dave sends are illustrated in Figure 4. According to the message, the student Dave has requested to do a lab called "tod" between 12:00pm and 2:00pm on the 26th of May 2010.

Almost instantaneously, the system sends feedback to Dave, indicating the reservation status. If the reservation is made, the reply contains the user name, the detailed date and time scheduled for the user. The system reply is shown in Figure 5. In this case, his desire was fulfilled and a request was made between 12:00 and 12:30 on the 26th of May 2011.

To prove that the scheduling was made successfully, the online scheduling system should depict that the time slot for the schedule made via SMS, is taken up. The online Scheduling system lists the available time slots for the user and also identifies available time slots via a popup window as illustrated

in Figure 6. The green sections show the free time slots while the red sections show the time slot which has already been booked by the user, through the SMS system.
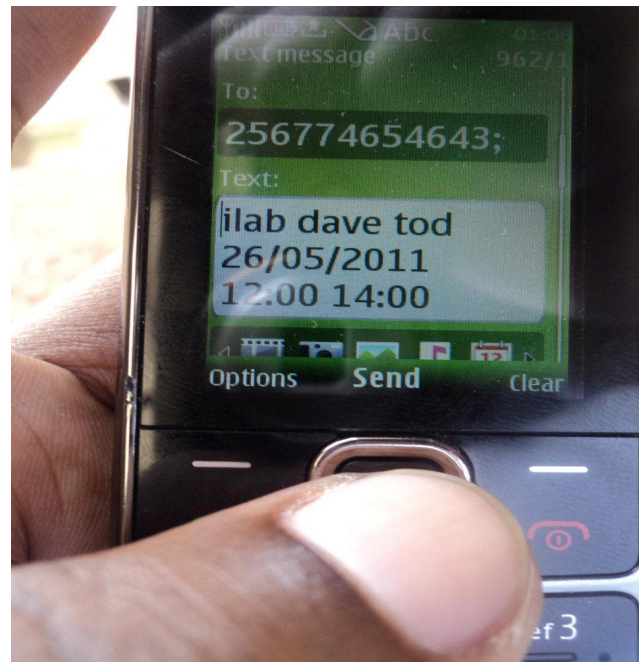

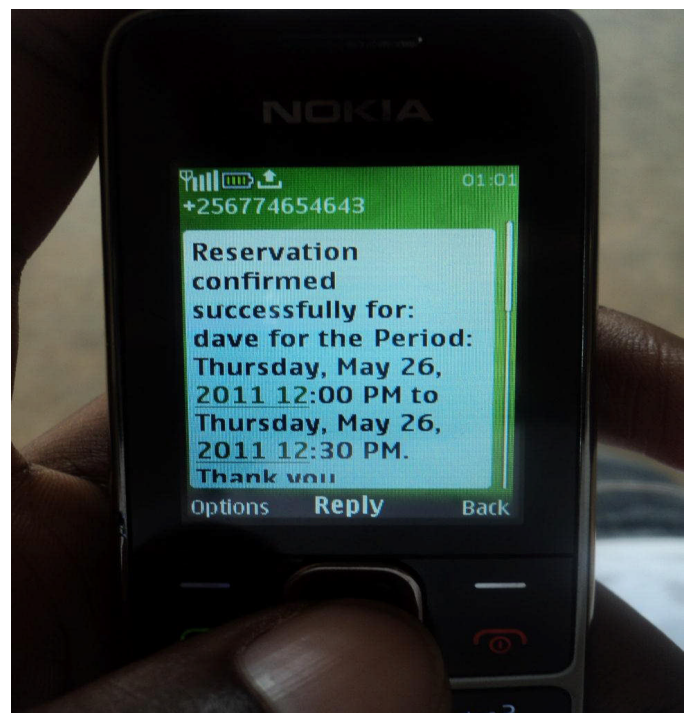Figure 4. User sends a scheduling request
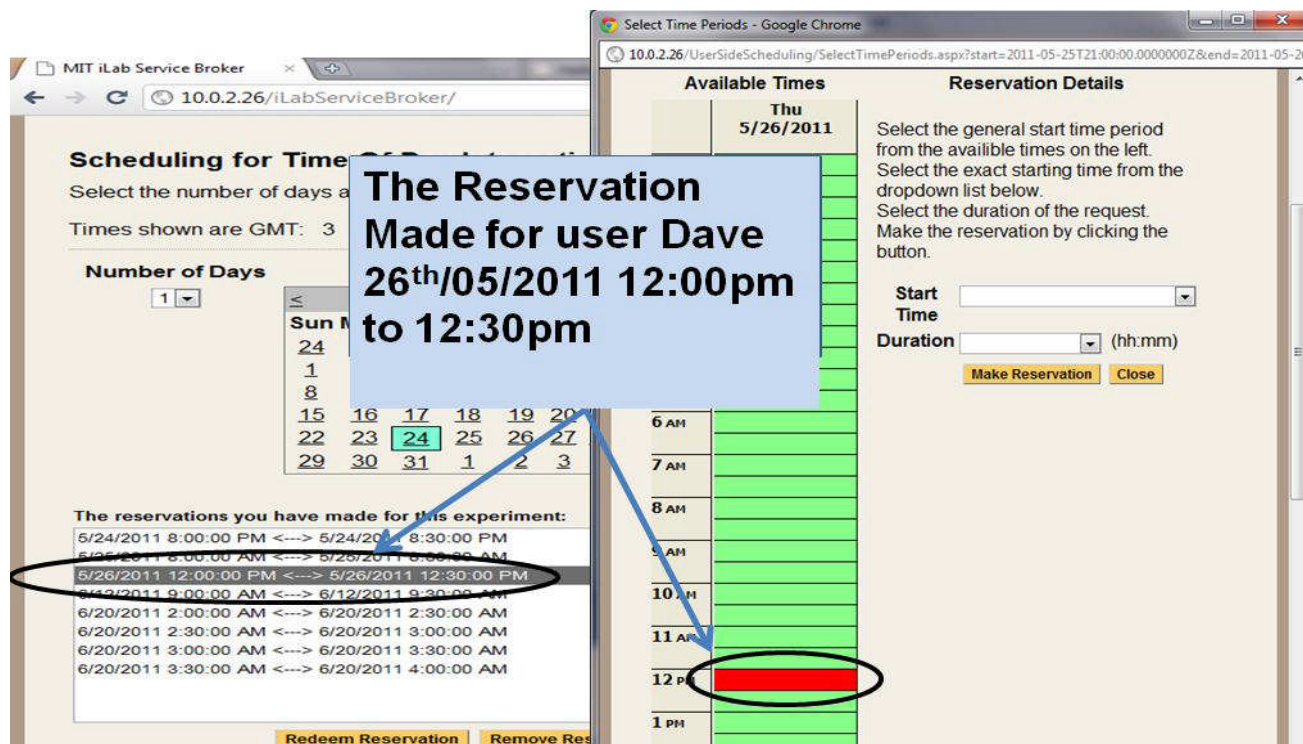

Figure 5. Reservation confirmation

Figure 6. Online system reserves the time booked

V. ERROR HANDLING

Systems are made close to perfection. But this is not the case with the people using them. At times, users tend not to follow instructions while using the system and definitely our case is not any different. In order to deal with errors and wrongly formatted messages, the system was tailored to identify these errors and reply accordingly.   Consider a scenario where Dave, at one point attempts to schedule for an interactive iLab by use of SMS, but unfortunately, he types his username twice. This brings a reply that points out why the schedule could not be made and gives advice on how to overcome the error as illustrated in Figures 7 and 8.
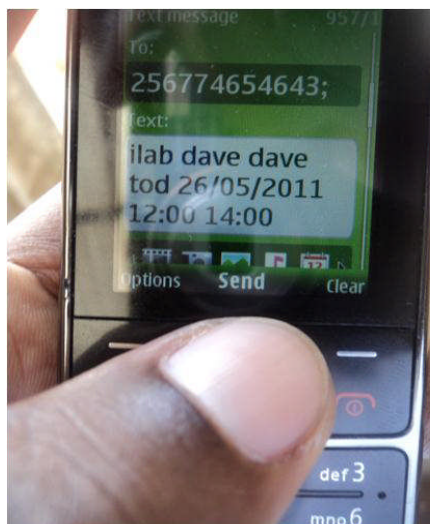

Figure 7. Wrongly formatted message

Figure 8. System response to a wrongly formatted message

VI. RESEARCH CONTRIBUTION AND CONCLUSION

The development of the SMS Scheduling project is coupled with various benefits, not only to the students who use the ISA for carrying out interactive labs, but also lecturers and administrators overseeing iLabs, the iLabs Society and the entire academia fraternity.

The students are the main beneficially of this research as they have convenience in carrying out interactive labs. Students who find it difficult to access computers will have their problem half- solved as they won't have to move up and down while trying to make reservations for the lab. Also, the time taken in making reservation by SMS is way too short compared to that of the online system. The time wasted in log-in and authentication is saved. This application brings in more flexibility in not only scheduling for the iLabs, but also in using the iLabs Shared Architecture to carry out interactive online laboratories as a whole. The student will find it much easier to make a request to do an iLab by sending a message, which is more familiar to them, and much more accessible than internet hotspot services.

Over the years, administrators at iLabs@MAK have encountered endless problems of students failing to meet deadlines for laboratory work. The application will help students to meet their deadlines more effectively since it cuts short the time spent in scheduling for a laboratory. iLabs@MAK and the iLabs society at large will for the first time be enriched with the availability of mobile applications. Therefore, this application has open doors for the optimization of internet laboratories using mobile applications. Needless to say, the SMS Scheduling System for interactive laboratories has added functionality to the iLabs Shared Architecture, and has the potential to support access to interactive iLabs worldwide to tens of thousands of users.

The concepts presented in this paper, along with the system requirements and expectations are also currently being used to develop similar and more comprehensive applications for other service providers, such as hospitals, clinics, restaurants and travel agencies in Uganda.

REFERENCES

[1] Mwikirize, C., Tumusiime, A. A., Musasizi, L., Namuswa, V., Nakasozi, M. D., Mugga, C. , Musasizi. P.I., Tickodri-Togboa, S.S., Butime, J., (2010), "Development of Online Laboratories for Modulation and Combinational Logic Circuit Analysis Using NI ELVIS II^{TM} Platform". 2010 Seventh International Conference on Information Technology (pp. 1069 - 1074).

[2] Mwikirize, C., Tumusiime, A.A., Musasizi, P.I., Tickodri Togboa, S.S., Katumba, A., Butime, J., "New Dimensions in Teaching Digital Electronics: A Multimode Laboratory Utilizing NI ELVIS II^{TM}, LabVIEW and NI Multisim". International Journal of Online Engineering (iJOE). Vol 6, No. 4 (2010) pp. 49-54

[3] Harward, V.J., Del Alamo, J.A., Lerman, S.R., Bailey, P.H.,Carpenter, J., DeLong, K., Felknor, C., Hardison, J., Harrison, B., Jabbour, I., Long, P.D., Tingting Mao, Naamani, L., Northridge, J., Schulz, M., Talavera, D., Varadharajan, C., Shaomin Wang, Yehia, K., Zbib, R., Zych, D (2008), "A Web Services Infrastructure to Build Communities of Internet Accessible Laboratories". IEEE Proceedings, Vol. 96, Issue 6, pp. 931 - 950

[4] Trifonova, A. and Ronchetti, M. (2003), "Where is Mobile Learning Going?" . In A. Rossett (Ed.), Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education 2003 (pp. 1794-1801). Chesapeake, VA: AACE

[5] Maiti, A. and Tripathy, B.K, "An improved scheduling scheme for the management of online laboratories". IEEE Conference on Recent Advances in Intelligent Computational Systems (RAICS), 2011. Proceedings, pp. 667-670.

[6] Kondabathini, V., Boutamina, S., and Vinjarapu, S.K.D., " A Theme to Unite The Resources of Different Remote Laboratories". IEEE International Conference on Technology for Education (T4E), 2011, pp. 51-55

[7] Gustavsson, I., Nilsson, K., Zackrisson, J., Garcia-Zubia, J., Hernandez-Jayo, U., Nafalski, A., Nedic, Z., Gol, O., Machotka, J., Pettersson, M.I., Lago, T., Hkansson, L., "On Objectives of Instructional Laboratories, Individual Assessment, and Use of Collaborative Remote Laboratories", Learning Technologies, IEEE transactions on, pp.: 263 - 274, Volume: 2 Issue: 4, Oct.-Dec. 2009

[8] Jiwaji, A., Hardison, J., Ayodele, K., Tickodri-Togboa, S., Mwambela, A., Harward, J., del Alamo, J., Harrison, B., Gikandi, S., "Collaborative Development of Remote Electronics Laboratories: The ELVIS iLab." ASEE. June 14, 2009. Austin, TX.

[9] J. Harward , J. A. del Alamo , V. S. Choudary , K. DeLong , J. L. Hardison , S. R. Lerman , J. Northridge , C. Varadharajan , S. Wang , K. Yehia and D. Zych "iLabs: A scalable architecture for sharing online laboratories", Int. Conf. Eng. Educ. 2004, 2004

[10] Lindsay, E.D, and Good, M.C, "Effects of laboratory access modes upon learning outcomes," IEEE Transactions on Education, vol. 48, No.4, November 2005

[11] Kannel: Overview. <http://www.kannel.org/overview.shtml>, 13.12.2012

89