ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/360139723

Development of an Open Source Desktop Application Client Software for Running Internet Enabled Laboratories

Conference Paper · April 2022

DOI: 10.13140/RG.2.2.16892.00640

CITATIONS
0

READS
286

2 authors, including:

Joel Tibabwetiza Muhanguzi
Sunbird AI

2 PUBLICATIONS **2** CITATIONS

SEE PROFILE

All content following this page was uploaded by Joel Tibabwetiza Muhanguzi on 23 April 2022.

The user has requested enhancement of the downloaded file.

IST-Africa 2020 Conference Proceedings
Miriam Cunningham and Paul Cunningham (Eds)
IST-Africa Institute and IIMC, 2020
ISBN: 978-1-905824-65-6

Development of an Open Source Desktop Application Client Software for Running Internet Enabled Laboratories

Joel MUHANGUZI TIBABWETIZA, Peter OKIDI LATING
iLabs@MAK project Makerere University, Kampala, 7062, Uganda
Tel: +256 701 008 715 Email: tibajoel90@gmail.com, englating@gmail.com

Abstract: While laboratory experiments have always been an essential for engineering students at the Faculty of Technology at Makerere University, they have faced major setbacks due to the expensive equipment required for the increasing number of students to interact with physical laboratory experiments. These would then be accessed remotely over the Internet. However, this architecture became unreliable with recent advancement in remote lab technology and expensive to maintain. The led to the adaptation of the Data Socket Transfer Protocol (DSTP) and C# Windows Forms to develop a new software for accessing iLabs. In turn, real time lab measurement data was acquired for experiment analysis and study. This created an open source desktop application that would then be new standard for running iLabs experiments. With this implementation, it would then facilitate adoption of hybrid e-learning labs in various academic institutions both at secondary and tertiary level.

Keywords: Data Socket Transfer Protocol, engineering, iLabs Shared Architecture, Internet enabled laboratories, real time, open source

1. Introduction

The Faculty of Technology at Makerere University was faced with a problem of catering for the increasing number of engineering students to efficiently carry out laboratory experiments. This was due to the insufficient and expensive laboratory equipment that led to the adoption of iLabs.

iLab is an abbreviation for Internet Enabled Laboratory. It is a physical laboratory experiment setup constructed on a circuit board with connectivity to a network by means of special hardware equipment and software to visualize circuit behaviour with help of graph elements.

The main hardware element onto which the physical circuit is constructed is the NI Elvis II board produced by National Instruments. This board is then connected to a networked computer i.e. the server, by means of a Universal Serial Bus (USB) cable equipped with software that sends laboratory measurement data from the physical circuit attached to it.

Students would then access readings from the experiment setup through means of special software known as a Virtual Instrument Suite (V1). It removes physical and time restrictions of the laboratory thus enabling students to have a better learning experience and understanding of fundamental engineering concepts [1].

In February 2004, iLabs were introduced to the Faculty of Technology at Makerere University. Subsequently, a memorandum of understanding was signed between Makerere University and MIT, leading to formation of the ilabs@MAK project under the umbrella of the iLabs Africa Project [2].

Authorized licensed use limited to: Makerere University Library. Downloaded on April 23,2022 at 13:36:32 UTC from IEEE Xplore. Restrictions apply.

MIT provided the initial infrastructure for setting up iLabs at the Faculty of Technology including NI Elvis II boards onto which laboratory circuits were to be constructed, and also trained student researchers on how to configure the software on computers to be able to send laboratory measurement data to remote computers and interact with lab experiments remotely. Figure 1 below shows the architecture of the lab interaction setup.

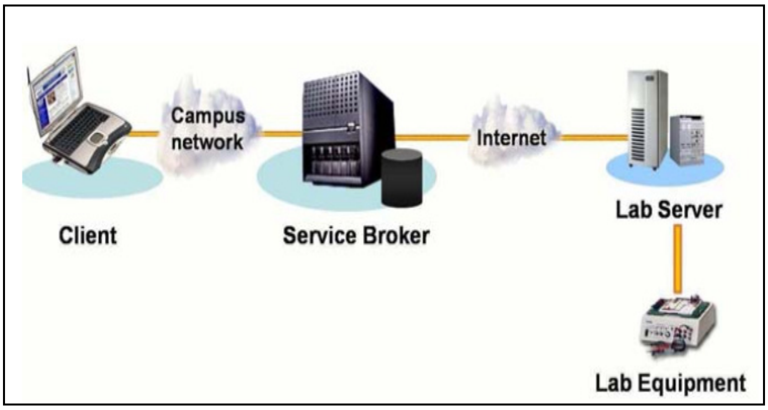

Figure 1 iLab Shared Architecture

With this setup, iLabs were used to establish a high level of interaction of students with experiment setups on a 24/7 basis, without being physically present in a laboratory environment, giving each individual a proper understanding of fundamental engineering concepts thus increasing the knowledge base of engineering students [3].

However, the architecture had shortcomings, which had created bottlenecks in setting up of online laboratories for use by students and student researchers. This was due to the incompatibility with new browsers through which the iLabs were to be accessed. It also required LabVIEW to design a new control interface for each new iLab that was developed.

A new architecture to mitigate these issues known as Makerere iLabs Shared Architecture (MISA) was developed which failed to provide real time data acquisition from laboratory experiments. It was also dependent on the LabVIEW software for acquisition of laboratory parameters from the laboratory hardware [4].

Finally, the MISA client interface was not scalable for newly developed laboratories as it required designing a new interface for each individual lab experiment. This led to development of a new stand-alone desktop software that depended on the data socket transfer protocol [5] and open source technologies for running Internet Enabled Laboratories.

2. Objectives

Main objective

To develop a fully functional open source client desktop application software that will be the standard for carrying out iLabs experiments.

Specific objectives

1. To implement a user control interface within the software that automatically adapts to various lab experiment setups for student practice without redesigning the software.
2. To establish real time lab data acquisition from remote iLabs experiment setups over a network to be visualized in the software for experiment results analysis.

Authorized licensed use limited to: Makerere University Library. Downloaded on April 23,2022 at 13:36:32 UTC from IEEE Xplore. Restrictions apply.

3. Methodology

The newly designed software was implemented using open source tools by Makerere University specifically under the iLabs@MAK research project. It was implemented in line with the curriculum for engineering students at the Faculty of Technology.

Since 2008 to date, four engineering students from the Department of Electrical and Computer Engineering have been recruited per academic year to join the project with two graduate research assistants overseen by the principal investigator to carry out this research.

The software was developed based on software development. This model was advantageous as it ensured seamless updates to the software based on feedback acquired from testing the developed software at a given time. Requirements engineering both functional and non-functional were also established to cater for challenges both to the iLabs@MAK project and students.

A test half wave circuit was setup on the NI Elvis II board and connected to a server computer linked to the Internet. The software tools used were visual studio 2019, measurement studio 2015, and the measurement studio installer builder for generating a software setup for installing the software on a client computer.

Finally, the last stage involved implementation and testing to achieve a fully working software that fulfilled the requirements.

4. Technology Description

The principle data transfer protocol used to transmit the lab experiment data from the lab hardware through the lab server was the Data Socket Transfer Protocol (DSTP). It provides real time experiment data sharing in an efficient manner in contrast to the data publishing protocol provided within the LabVIEW web VIs.

C# windows forms was used to design a user interface which would automatically adapt to both new and old ilab circuit experiments developed for students to access experiment values over a remote client computer. This interface is installed once on the client computer using a click installer software package generated by measurement studio installer builder.

The desktop application client software was implemented based on the block diagram as illustrated in Figure 2 below.

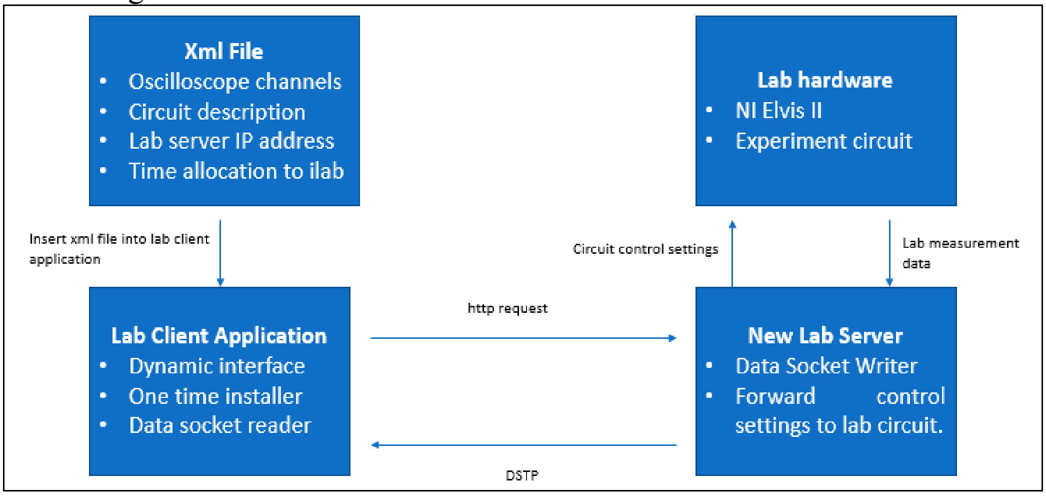

Figure 2 New lab software architecture

An XML file was used to store the configuration to be used to access the iLab on the remote client computer. It included parameters constituting the oscilloscope channels on the NI Elvis II+ onto which the waveforms were acquired from the physical circuit, the server computer IP address and other lab parameters to be loaded into the client software.

Authorized licensed use limited to: Makerere University Library. Downloaded on April 23,2022 at 13:36:32 UTC from IEEE Xplore. Restrictions apply.

The HTTP protocol on the other hand was used to transmit control settings from the lab client application to the lab hardware through the lab server. These control settings include wave signal types i.e. cosine, square waves, switch signals to trigger relays on the physical iLab circuit and corresponding oscilloscope channels from which to receive signal waveforms to be displayed in the user interface waveform graph provision.

5. Developments

A physical half wave rectification circuit was constructed on the NI Elvis II board and connected to the server computer as shown in Figure 3. The data was acquired using the data socket reader within the lab client application and displayed using a waveform graph within the new software in real time without lagging in contrast to the LabVIEW web VI setting employed in the previous architecture.

The interface was designed independent of LabVIEW tools with measurement studio controls including knobs for changing frequency values, amplitude values and switches to activate capacitors for smoothening the rectified waveform on the physical lab circuit. This established an open source application built entirely on the .NET framework.

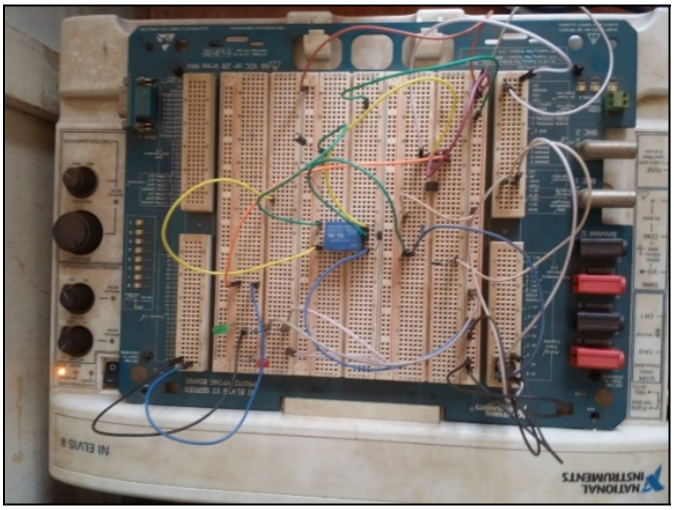

Figure 3 Half wave circuit on NI Elvis II board

A standard test xml file with the lab specifications inclusive of the lab name, the amount of time for accessing the lab, the oscilloscope channels, the switches, power supplies and authentication details for a particular student was effectively used for accessing the ilab experiment.

```xml
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<!--The xml data required for starting the lab!-->
<Settings>
  <Setting Name="DateTime" Value="04/01/2019 2:25 PM" Type="Info"/>
  <Setting Name="Channel0" Value="/wave0" Type="Channel" Index="0"/>
  <Setting Name="Channel1" Value="/wave1" Type="Channel" Index="1"/>
  <Setting Name="S1" Value="S1" DevicePath="DI0" Type="Switch" />
  <Setting Name="S2" Value="S2" DevicePath="DI1" Type="Switch" />
  <Setting Name="Lab Name" DevicePath="" Value="ss" Type="Info" />
  <Setting Name="Lab Author" DevicePath="" Value="ss" Type="Info" />
  <Setting Name="Lab Url" DevicePath="" Value="10.120.4.155" Type="Info" />
  <Setting Name="Api Port" DevicePath="" Value="9000" Type="Info" />
  <Setting Name="Device" DevicePath="" Value="Dev1" Type="Info" />
</Settings>
```

Figure 4 XML file with lab parameters

Authorized licensed use limited to: Makerere University Library. Downloaded on April 23,2022 at 13:36:32 UTC from IEEE Xplore. Restrictions apply.

Based on the details within the xml file, the appropriate lab controls were loaded onto the lab interface dynamically with the data socket communications attached to them. This created a software that was automatically adapted to scale for different iLabs without having to redesign the software to work for each individual iLab experiment. The structure of the xml file is shown in Figure 4.

6. Results

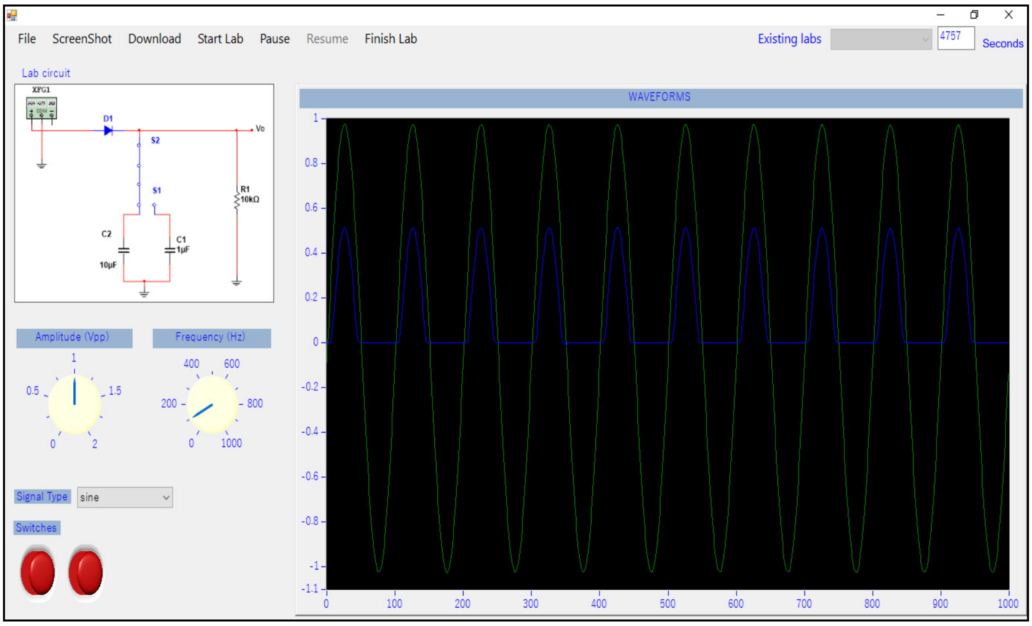

Figure 5 Lab client interface with XML file loaded with DSTP wave data

The result of the development process was an open source desktop application client software that was fully functional with all intended ilab functionalities ranging from adjustable signal types subjected to an engineering circuit, being able to observe both output and output waveforms within the user interface and controlling digitally triggered points of engineering circuits by use of toggle switches as shown in Figure 5 above.

Real time data acquisition was established through the data socket transfer protocol. This ensured more accurate lab results analysis in comparison to the previous lab architectures that used the LabVIEW web publishing protocol that relied on buffered results.

Being a software independent of LabVIEW technologies both in implementation and access of lab experiment data by the client, and purely dependent on the .NET framework, it was an achievement that greatly solved the problem of interface redesign by use of LabVIEW for each individual ilab experiment which had severely delayed deployment of iLabs at the faculty of engineering at the University for students.

7. Business Benefits

The pre-existing architectures were expensive to setup due to the costly laboratory equipment required to have them running. In comparison, the remote laboratory setup with the physical laboratories hosted at the Faculty of Engineering is cheap as it facilitates access to laboratories over the Internet at a fee without purchasing actual laboratory equipment.

The customer target group are the tertiary institutions like universities and secondary schools, who would employ this remote laboratory access to teach fundamental engineering concepts and science subjects. This would then bridge the gap between theoretical knowledge and actual physical aspects of engineering design.

Authorized licensed use limited to: Makerere University Library. Downloaded on April 23,2022 at 13:36:32 UTC from IEEE Xplore. Restrictions apply.

In turn the adoption of science related subjects would increase at higher institutions of learning, which is inline with the National Development Plan of Uganda.

This is in progress with pilot universities such as Busitema University and secondary schools actively participating in the Science and Technology Innovation competition organized annually by the iLabs@MAK project.

The software can be sold to various engineering firms for use in industrial monitoring and automation processes as it would provide remote engineering and sensing capabilities which could facilitate company progress towards advancing their industrial processes in order to improve productivity along manufacturing lines hence generating income for the research project.

In addition, the software is copyrightable which would earn a loyalty fee to the research project as it could be leased out for use by other organizations in their applications making it a secondary income source for the research project.

8. Conclusions

A fully functional software was achieved at the end of the development process with an installer package to install it on a windows computer through a setup. It supports real time data acquisition for lab instrument data through the user to the client interface.

With this new standard for access of iLab experiments, the learning experience of students will be greatly improved. Further research work should be done to implement software that runs on other operating system platforms in addition to the windows platform. This will accommodate the vast student community within the faculty and potential partner engineering firms.

The research however was limited to a few tests with a few student's laptops and a few lab computers hence the information regarding the user experience on different computing environments was not sufficient enough to make a proper conclusion that the software was reliable to out-perform the old lab client architectures. This calls for further tests and research work through test deployments to acquire vast user feedback.

It is recommended to formalize more partnerships with engineering firms to customize the software to their industrial automation processes at an agreeable amount to be able to generate a feasible income for the research project and other universities and secondary schools to adopt to this technology to blend it in their learning eco system for science related subjects to motivate students to take on science courses at advanced levels of education.

References

[1] A. F. Ahmed Naddami, "Online laboratory in Digital Electronics using NI Elvis II +," Thessaloniki, Greece, 2014.

[2] O. B. Akinawale, A. T. Asiimwe, C. Mwikirize and S. Stevens, "Online Laboratories: Enhancing the Quality of Higher Education in Africa," 2011.

[3] A. F. M. G. A. P. Ahmed Naddami, ""Importance of a network of online labs in magrebian countries"," *Proceedings of the IEEE,* pp. 77 - 78, 2014.

[4] V. e. a. Haward, ""The iLab Shared Architecture: A web services infrastructure to build communities of Internet Accessible Laboratories."," *Proceedings of the IEEE,* vol. 96, no. 6, pp. 931 - 950, 2008.

[5] Z. L. M. Tao W., "DataSocket technology and its application in equipment remote condition monitoring and fault diagnosis," *International Journal on Smart Sensing & Intelligent Systems,* pp. 491 - 508, 2016.

Copyright © 2020 The authors

Authorized licensed use limited to: Makerere University Library. Downloaded on April 23,2022 at 13:36:32 UTC from IEEE Xplore. Restrictions apply.
View publication stats