Design of Graphical User Interface (GUI) on IoT-based remote laboratory for Programmable Logic Controller (PLC) practicum and pneumatic simulation

L A Pratama* dan J Kustija
Departemen Pendidikan Teknik Elektro, Universitas Pendidikan Indonesia, Jl. Dr. Setiabudhi 207, Bandung 40154, Indonesia

*lutfiakbarp21@upi.edu

Abstract. The remote laboratory system allows users to conduct lab work via a remote-controlled lab computer. Each computer has a different interface to interact with the operating system. This affects the comfort and effectiveness of users in accessing remote laboratories. Based on this, we need a support system in the remote laboratory to maximize user time in practicing. The interface on the computer lab becomes an important factor in supporting practicum that uses computers. GUI (Graphical User Interface) is an interface that allows users to interact with the operating system graphically. Unlike the CLI (Command Line Interface), which uses the command line, the GUI uses a graphical menu, thus allowing users to more easily and efficiently interact with the operating system. The main features of the GUI are made up of three components. The components in question include time management features, application selection, and notifications. If already connected to the GUI, users can easily open the applications provided, such as CX-Programmer, Pneumatic Festo Fluidsim, and the camera. Also, the GUI functions as a timer and notifies users. This can be seen from the automatic connection and disconnection between the user and the computer lab if the practicum time is up.

1. Introduction
Remote Laboratory is an integrated web system that can be accessed through an integrated interface by a web browser [1]. The Remote Laboratory system allows users to conduct lab work via a remote-controlled lab computer. Remote laboratories have been proven to provide significant benefits, for example, including increased time for user access to equipment, institutional resources and security issues [2].

PLC Practicum and Pneumatic Simulation use computers as practicum media, but each computer has a different interface to interact with the operating system. This certainly raises various problems. Based on this, we need a support system in the remote laboratory to maximise user time in practising. The interface on the computer lab becomes an important factor in supporting practicum that uses computers as media. GUI (Graphical User Interface) is a system interface that allows users to be able to interact with the operating system or computer graphically [3].

The design of the GUI on a computer lab can make it easier for users to interact with the operating system when doing lab work. The user has a simple interaction with the interface and empowers the user by providing an interactive and intuitive interface for the application program [4,5]. On this occasion,
the author aims to create a GUI design on an IoT-based remote laboratory for PLC practicum and pneumatic simulation.

2. Methods

The research method carried out in this study is an experimental method which includes the study of literature, design and testing. The first stage is to conduct a literature study by opening several scientific journals related to the GUI.

The second stage is the design stage, planning begins with making a moving actuator on the PLC trainer using a conveyor, pneumatic solenoid valve and proximity sensor. Then the GUI design is done as a liaison between the lab computers with the practicum media.

The design of the GUI design was made using Microsoft Visual Studio .NET (VS.NET). GUI will be active or appear according to a predetermined schedule and users can do the practicum after gaining access. The software used is the CX-Programmer software for PLC practicum and Fluidsim Pneumatic software for Pneumatic simulation practicum. In this system, there is also a Logitech C270 Webcam camera to supervise practicum media. This step begins with designing the Flowchart first.

![Flowchart](image)

**Figure 1.** GUI design flowchart.

After the design of the flowchart is complete, the trial continues by applying the GUI design on the computer server and practicum testing.
3. Results and discussion

3.1. Design results

The remote laboratory system only allows users to do lab work through a computer lab that is controlled remotely from any location, through a web browser.

Unlike the remote laboratory systems in general, this tool adds a support system for users to interact with the lab computer. The support system is an interface that allows users to interact with the operating system/computer lab more quickly and efficiently.

The interface in the operating system is divided into two types, namely, GUI (Graphical User Interface) and CLI (Command Line Interface). Unlike the CLI, which uses the command line, the GUI uses a graphical menu, thus allowing users to run with the operating system more efficiently.

The design and implementation of a GUI require a significant level of expertise and long development time [6]. The reason for applying the GUI design in a computer lab is that it makes it easier for users to facilitate the operating system and optimize lab time by providing an interactive interface for application programs.

Following the design steps:

This GUI design is made according to flowchart and uses Visual C# programming language with .NET Framework as Common Language Infrastructure in Microsoft Visual studio [7,8]. The reason for choosing the .NET Framework is because the .NET Framework makes it easy to build software applications [9]. The results of the GUI design can be seen in Figure 3.

![Figure 2. Architecture design of GUI at remote laboratory.](image-url)

![Figure 3. Display of GUI design.](image-url)
The following features are found in the design of the GUI design:

3.1.1. Time management. The time management feature functions as a countdown in practising. The practicum time is determined according to the agreement between the user and the instructor. Display this feature can be seen in Figure 4.

![Figure 4. Display of time management in the GUI.](image1)

This feature begins by creating a Private Class Timer for setting time intervals. The class parameters are set so that it creates a counter every one second. After completing the interval, proceed with setting the countdown variable for the second, minute and hour settings using the decrement technique in Microsoft Visual Studio. In addition to the above features, the time setting can also function as the opening of the main window of the GUI automatically when it has entered the practicum time. The main window opening technique uses the Scheduler Task which regulates when the main window will open automatically.

3.1.2. Practicum application

![Figure 5. Display practicum application on the GUI.](image2)

The user can open the application by pressing the available menu as in Figure 5. This process is created by running the programs that are available on each menu. The program in question is "Start application.exe" or can also be accessed by calling the director "start C: \ Program Files (x86) \ OMRON \ CX-One \ CX-Programmer.exe" in coding Microsoft visual studio.

![Figure 6. Display closing the GUI.](image3)

If the user presses the FINISH button or the time is up, then all practicum applications will be forced out. This display can be seen in Figure 6. This process is done by calling the command prompt on Microsoft Visual Studio that has included the program to close the application. The program in question is "Taskkill / F / Im application.exe".

3.1.3. Notifications. Notification feature functions as notification regarding practicum activities. Notifications that are notified in the form of reminding practicum time. Display notifications can be seen in Figure 7.
Figure 7. Display Notification on the GUI.

The contents and activation of the notification popup are adjusted to the time variable in the Private class timer or according to the time available in the time management feature. When the specified time variable is reached, a notification popup will appear to notify the practicum activity. This process is done by calling the PopupNotifier library available at Microsoft Visual Studio. Popup settings include popup.image to display images, popup.titletext for titles, popup.contentext that is set to change according to the remaining time possible.

3.1.4. Close program. The function of the close program feature is to close the GUI main window when the lab is finished or when the time is up. Users can close the main window GUI by pressing the EXIT button when they have finished the practicum, or the GUI will close automatically when the countdown time has run out. This process is made by running a program in Microsoft Visual Studio that is tailored to the private class timer. The program in question is calling the keyword "this.close ()" to close the main GUI window.

3.2. Testing

The following are the results of GUI Design that have been made and accessed by users.

3.2.1. Test the PLC practicum. Users can do the PLC practicum following the worksheet that has been provided. In the lab, computer PLC software is installed and has been integrated with the GUI. The software available is CX-Programmer; users can open it by pressing the button on the main window GUI, which is available in the lab application features. The PLC practicum display with CX-P can be seen in Figure 8.

Figure 8. Display of PLC practicum.

When conducting a PLC practicum and transferring the program to a PLC, the user can see the results of the practicum with a camera that leads to the lab equipment.

3.2.2. Test of pneumatic simulation practicum. Pneumatic simulation practicum using Pneumatic Festo Fluidsim software [10]. Users can access the Pneumatic Festo Fluids by clicking the Festo Fluidsim menu on the GUI main window. Display pneumatic simulation practicum can be seen in Figure 9.
3.2.3. Camera trial. The camera component allows users to supervise practicum equipment. Users can access the camera by clicking the camera menu on the main window GUI. The camera trial display can be seen in Figure 10.

4. Conclusions
Based on the results and discussion, it can be concluded that the design of GUI (Graphical User Interface) in IoT-Based Remote Laboratory for PLC (Programmable Logic Controllers) Practicum and Pneumatic Simulation was successfully made according to the plan. The trial results show that every part and process runs well where the user can access the software easily. The features provided are running well such as proper notifications according to practicum time, app selection goes well and appropriate time management.

References
[1] Tokdemir G and Bilgen S 2008 Remote lab effectiveness assessment model 2008 11th International Conference on Optimization of Electrical and Electronic Equipment pp 234-239
[2] Lindsay E D, Murray S and Stumpers B D 2012 A toolkit for remote laboratory design and development Int. J. Online Eng. 8(1) 14–19
[3] Nakajima H, Masuda T and Takahashi I 2013 GUI ferret: GUI test tool to analyze complex behavior of multi-window applications Proc. IEEE Int. Conf. Eng. Complex Comput. Syst. ICECCS 2 163–166
[4] Yu D C, Jamal N, Wang W, Zang X G and Brandt J R 1994 A Graphical User Interface for Design, Simulation and Analysis of Power Plant Electrical Auxiliary Systems IEEE Trans. Energy Convers. 9(2) 263–269
[5] Billard E A and RiedmillerA E 1997 Laboratory exercises using a graphical user interface to a
queueing simulator *IEEE Trans. Educ.* 40(2) 154–157

[6] Kushiro N, Sato K, Komiya N, Nakata M, Inoue M and Tsubota H 1996 Agent based graphical user interface architecture for consumer products *IEEE transactions on consumer electronics* 42(4) 1044-1049

[7] Louridas P 2007 Declarative GUI programming in Microsoft Windows *IEEE Softw.* 24(4) 16–19

[8] Ling T H Y, Wong L J, Tan J E H and Lee C K 2015 XBee wireless blood pressure monitoring system with microsoft visual studio computer interfacing *2015 6th International Conference on Intelligent Systems, Modelling and Simulation* pp 5-9

[9] Moise D L, Wong K and Sun D 2004 Integrating a reverse engineering tool with Microsoft Visual Studio.NET *Proc. Eur. Conf. Softw. Maint. Reengineering, CSMR* 8 85–92

[10] Stipišić L, Kosić D and Kurtela M 2012 Design and simulation of turbine governing hydraulic system with FluidSIM software *2012 Proceedings of the 35th International Convention MIPRO* pp 800-805