International Journal of Engineering & Technology, *8 (1.9) (2019) 110-113*

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Prototype Design of Client Server-Based Software to Control Client Computer in Computer Laboratories

Elis Hernawati*, Tedi Gunawan, and Aris Hermansyah Suryadi

Diploma of Information Management, Telkom University, Bandung, Indonesia
**Corresponding author E-mail: elishernawati@tass.telkomuniversity.ac.id*

Abstract

In the practicum activities in the laboratories of School of Applied Science, Telkom University, the computer server is used by lecturers and the client computers are used by the students. When lecturers are teaching, the students often do not pay attention to the lecturers. The students operate the computers to play game, to access Facebook, or to do other activities that have nothing to do with the learning process. This paper describes the process of prototype design of client server-based software to control the client computers on the Local Area Network in the computer laboratory, so that the lecturers can control the activities of client computers used by the students. This software is designed to control the mouse, keyboard, and monitor screen of the client computers from the computer server. The method used to build the prototype of this software is the prototyping model. Meanwhile, the programming language used is Visual Basic. The results of this study are used to help programmers in building software to control clients in the computer laboratories, so that the software can help lecturers in the laboratory in monitoring the client computer used by students.

Keywords: *Prototype, Software Design, Client Server, LAN, Visual Basic*

1. Introduction

Computer laboratory is an important infrastructure for the teaching and learning process, especially the practicum. In the computer laboratory a network system called Local Area Network (LAN) is installed. The LAN architecture consists of a computer server as the controller and several client computers. In the practicum activities in School of Applied Science, Telkom University, the computer server is used by lecturers and the client computers are used by students. When lecturers are teaching, the students often do not pay attention to the lecturers. The students operate the computers to play game, to access Facebook, or to do other activities that have nothing to do with the learning process. This causes the lecturers need to be able to control the students' computers from the computer server. To solve this problem, software installed on the server should be built to control the client computer. The software controlling the client computer to be built is client server-based software. The software is built by programmers by using Visual Basic programming language. To facilitate programmers in building the software, a prototype design of the software is required as the early stage of software development.

This paper describes the process of prototype design of client server-based software to control the client computer. The software has three functions, namely to enable or disable client's mouse and keyboard, to make client's monitor screen normal or blank, and to check client's activities through a task manager. The process of prototype design includes designing architecture of the software, the use case diagram, and the mockup interface.

Remote access systems can be used to control a network management. By using remote access administrators can easily control and check the client computer, such as disabling client's mouse and keyboard, making client's monitor screen blank, and checking client activities through a task manager. Visual Basic is one of programming languages that can be used to create a remote access system. Various kinds of programs can be made by using Visual Basic, such as programs that integrate database, network, office automation, and web application. Javier García et al. (2009) presented the main characteristics of a Remote Laboratory, analyzed the software technologies to implement the client and server sides in a WebLab, and correlated these technologies with the characteristics to facilitate the selection of a technology to implement a WebLab. Franco Davoli et al. (2006) reported the architecture developed and the results achieved within the three-year term LABNET project, which has addressed these issues within an integrated laboratory environment, aimed at providing unified access to heterogeneous equipment for a multiplicity of users with potentially different capabilities and skills. Ankush Mittal et al. (2008) stated that the framework provides an instructor-friendly remote monitoring of lab, effective evaluation, and grading methodology. The system also provides a student- friendly remote login, software access, and problem resolution through effective help from the teacher. The framework is built on a bandwidth-efficient client–server model-based backend system written in Java. Alessandro Ferrero et al. (2003) showed how this goal has been achieved by implementing a Java-based client-server architecture that is presently used as a subsidiary laboratory for the measurement subjects in the Electrical Engineering curriculum at the Politecnico di Milano. Ingvar Gustavsson (2003) presented the remote laboratory at Blekinge Institute of Technology in Sweden (BTH). It is neither a SCADA application nor a simulation. The three main objectives of the laboratory are:

1. to offer remote laboratory experiments to on-campus students as well as to distance learning students as part of conventional courses in electrical engineering;

2. to design remote laboratory exercises that are almost identical to local ones; and

3. to use the equipment and the premises more efficiently.

2. Materials and Methods

Prototyping model is used in this study. This model consists of three phases, namely: [1] listen to user, [2] build prototype, and [3] user test. Prototyping is done repeatedly until the software built is in accordance with the needs of the user.

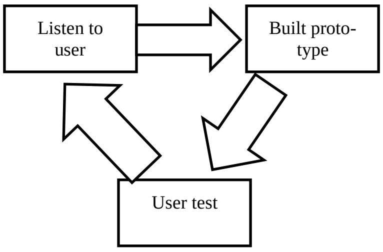

Fig 2: Prototyping Model

A. Listen to user

In this phase, communication with the users (which are lecturers who teach practicum) is done to find out what is needed in building the prototype.

B. Build prototype

In this phase, the process to build the prototype is done, which is designing the architecture of prototype, use case diagram, and user interface. This process is done repeatedly and periodically until the prototype is in accordance with the needs of the users.

C. User test

In this phase, a test to the prototype is done to ensure whether the prototype is in accordance with the needs of the users or not.

3. Result

This research aims to build a prototype of client server-based software, to control the client computers in the computer laboratory of School of Applied Science, Telkom University. Figure 1 represents the architecture of software design by using the hierarchical tree, which describes the structure of software.

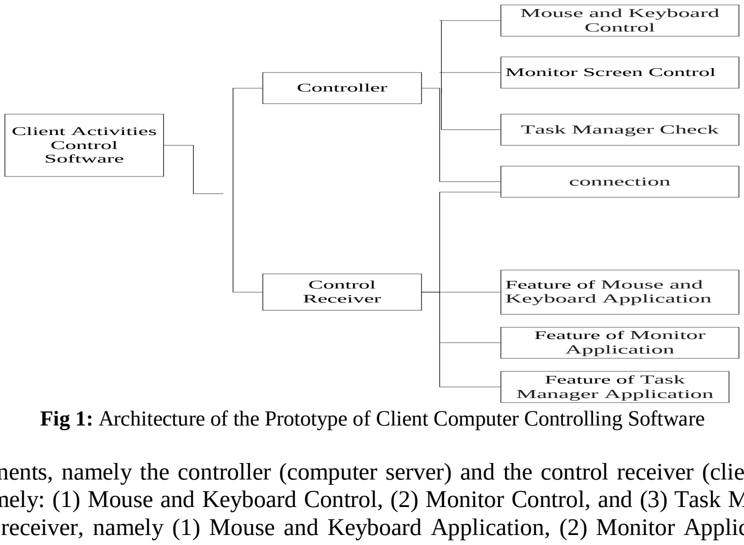
Client Activities Control Software
Controller
Control Receiver
Mouse and Keyboard Control
Monitor Screen Control
Task Manager Check
connection
Feature of Mouse and Keyboard Application
Feature of Monitor Application
Feature of Task Manager Application

Fig 1: Architecture of the Prototype of Client Computer Controlling Software

The software has two components, namely the controller (computer server) and the control receiver (client computer). There are three features on the controller, namely: (1) Mouse and Keyboard Control, (2) Monitor Control, and (3) Task Manager Check. There are also three features on the control receiver, namely (1) Mouse and Keyboard Application, (2) Monitor Application, and (3) Task Manager Application. However, these three features on the control receiver are hidden, in order that the client is not aware of the features. The features function to respond to commands from the controller. The software will work if the controller (server) and the control receiver (client) has been connected each other.

3.1. Sofware Use Case Diagram

Use case diagram describes the functions in the software and the users that interact with the system. The controller is the actor that directly interacts with the system to control the client computer (the control receiver). When the control receiver (client computer) receives a command from the controller, it will automatically run the functions instructed by the server without displaying any visual of application interface. Figure 2 describes the use case diagram of software prototype to control the client computer.

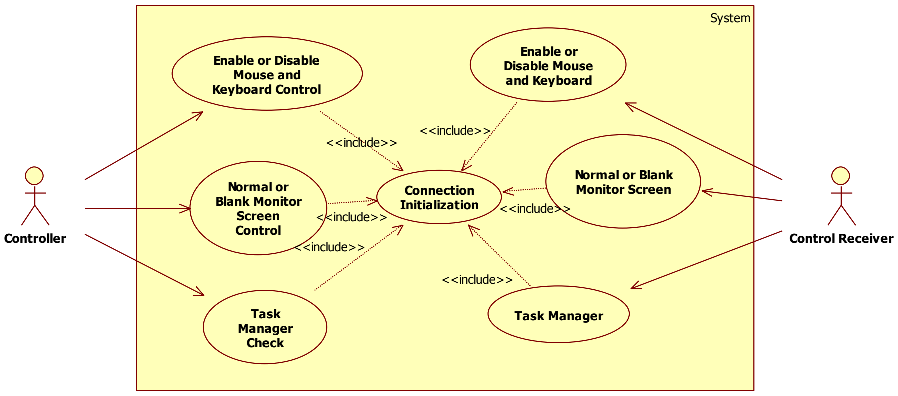
System
Enable or Disable Mouse and Keyboard Control
Enable or Disable Mouse and Keyboard
<<include>>
<<include>>
Controller
Normal or Blank Monitor Screen Control
Connection Initialization
Normal or Blank Monitor Screen
<<include>>
<<include>>
<<include>>
Control Receiver
<<include>>
Task Manager Check
Task Manager

Fig 2: Use Case Diagram of the Prototype of Client Computer Controlling Software

The controller (server) can perform the functions of enable or disable mouse and keyboard control, normal or blank monitor screen control, and task manager check, after the function of connection initialization with the client is activated. The control receiver (client) can respond to the three functions of the server after it is connected with the server.

The following is the explanation of the order of activities in each process described in the Activity Diagram.

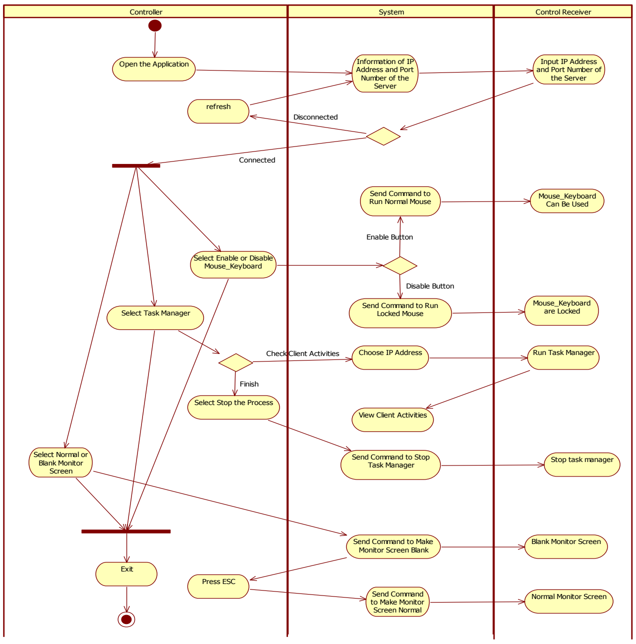
Controller
System
Control Receiver
Open the Application
Information of IP Address and Port Number of the Server
Input IP Address and Port Number of the Server
refresh
Disconnected
Connected
Send Command to Run Normal Mouse
Mouse_Keyboard Can Be Used
Enable Button
Select Enable or Disable Mouse_Keyboard
Disable Button
Send Command to Run Locked Mouse
Mouse_Keyboard are Locked
Select Task Manager
Check Client Activities
Choose IP Address
Run Task Manager
Finish
Select Stop the Process
View Client Activities
Select Normal or Blank Monitor Screen
Send Command to Stop Task Manager
Stop task manager
Send Command to Make Monitor Screen Blank
Blank Monitor Screen
Exit
Press ESC
Send Command to Make Monitor Screen Normal
Normal Monitor Screen

Fig 3: Activity Diagram of the Prototype of Client Computer Controlling Software

When the controller opens the application, the system will give information to IP and Port Server. The control receiver inputs the IP and Port Server. If the connection fails, the controller can perform a “refresh”.

After the process of connection initiation succeeds, the controller can choose to use the three existing functions:

1. The function of Enable or Disable keyboard & Mouse

If this function is selected, the system will send a command to open or lock the mouse and keyboard, so that the client’s mouse and keyboard can or cannot be used until the time limit specified in the program.

2. The function of Normal or Blank monitor screen

If this function is selected, the system will send a command to make the client’s monitor screen blank, until the controller presses the “escape” button to make the client’s monitor screen back to normal.

3. The function of Task Manager Check

If this function is selected, the controller can check the activities on the client computer. If the controller wants to see the activities on a client’s computer, the system will choose the client’s IP address that is inputted from the controller and the server’s task manager will display the client activities. If the controller wants to stop the process of checking the client activities, it can select the function of stop the process and the system will send a command to stop the task manager.

The Mockup Design of the software is written by using Visual Basic 6 programming language. It can be seen in the picture below.

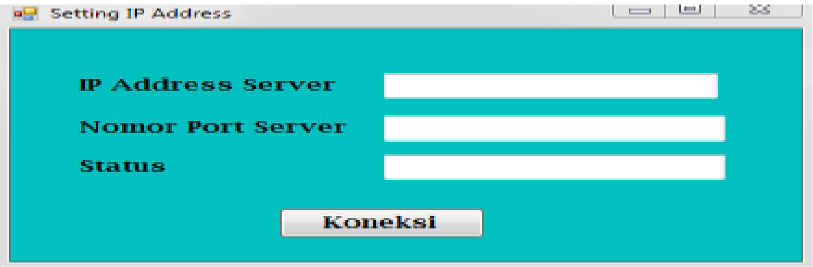

Fig 4: Mockup of Setting IP Address on the Client Computer

The Mockup Form of Setting IP Address is created on the client computer. It is used to initiate a connection between the client computer and the computer server. The process of connection initiation is set when the software is installed for the first time, in order that it can be connected directly to the server. The setting process is done by inputting the IP address and the port number of the server and pressing the connect button. If it is successfully connected, the status "connection" will appear in the textbox.

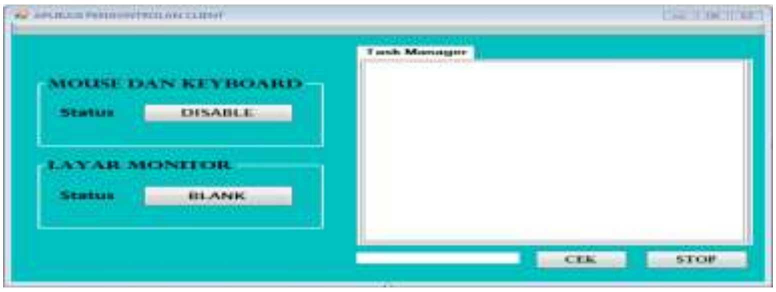

Fig 5: Form of Client Control on the Computer Server

The Mockup Form of client computer control is created on the server. This form is used to control the client computer. This form has four command buttons, namely the “Disable” button on the keyboard and mouse frame that is used to disable client’s keyboard and mouse, the “Blank” button on the monitor screen frame that is used to make client’s monitor screen blank, the “Cek” (Check) button under the taskbar windows that is used to display the client activities on a task bar that the IP address has been inputted, and the “Stop” button that is used to stop the process of checking the client activities on the task bar.

4. Conclusion

Based on the discussion above, the conclusions are:

1. The prototype of client server-based software to control the client computer that has been built can be used to help the programmers to do the code generation process that will result in a client activities control software. The software that will be built can help lecturers in the practicum activities to control and check client computers in the laboratory.

2. The prototype design includes the design of client control form that has the functionality buttons, namely Enable or Disable Keyboard and Mouse, Normal or Blank Monitor Screen, and Task Manager Check.

References

[1] Abastante, F., & Lami, I. M. (2012). Quality function deployment (QFD) and analytic network process (ANP): an application to analyze a cohousing intervention. Journal of Applied Operational Research, 4(1), 14-27.

[2] A. Stokes, T. Collins, J. Maskall, J. Lea, P. Lunt, and S. Davies, “Enabling Remote Access to Fieldwork: Gaining Insight into the Pedagogic Effectiveness of ‘Direct’ and ‘Remote’ Field Activities,” J. Geogr. High. Educ., vol. 36, no. 2, pp. 197–222, 2012

[3] Bouchereau, V., & Hefin, R. (2000). Methods and techniques to help quality function deployment (QFD), benchmarking. An International Journal, 7(1), 8-19. MCB University Press, pp. 1463-5771.

[4] Büyüközkan, G., Fiyzioglu, O., & Rual, D. (2007). Fuzzy group decision-making to multiple preference formats in quality function deployment. Computer Industry, 58(5), 392-402.

[5] B.-S. J. B.-S. Jeon, E.-J. K. E.-J. Ko, and G.-H. L. G.-H. Lee, “Wireless LAN network management system,” 10th Int. Conf. Telecommun. 2003 ICT 2003, vol. 2, no. 4, pp. 615–620, 2003.

[6] C. E. C. Bastidas, “Enabling remote access to computer networking laboratories for distance education,” in Proceedings - Frontiers in Education Conference, FIE, 2011.

[7] Chien-Jung, L., Chih-Hung Hsu, & Fang Ou-Yang. (2012). A hybrid mining and predicting system based on quadratic exponential smoothing model and grey relational analysis for green supply chain. Academic Journal, 11(8), 336-348.

[8] Chiu, M. C., & Kremer, G. E. O. (2011). Investigation of the applicability of Design for X tools during design concept evolution: a literature review.

[9] J. Melorose, R. Perroy, and S. Careas, “The Remote Framebuffer Protocol,” Statew. Agric. L. Use Baseline 2015, vol. 1, pp. 1–39, 2015.

[10] R. Hamid and S. A. Mohammed, “Remote Access Laboratory System for Material Technology Laboratory Work,” Latest Trends Eng. Educ., pp. 311–316, 2010.

[11] V. Mateos, L. Bellidoa, V. A. Villagrá, T. Richter, and A. Gallardo, “Access control for shared remote laboratories,” J. Res. Pract. Inf. Technol., vol. 44, no. 2, pp. 111–128, 2012.

[12] Y. Lin, T. K??m??r??inen, M. Di Francesco, and A. Yl??-J????ski, “Performance evaluation of remote display access for mobile cloud computing,” Comput. Commun., vol. 72, pp. 17–25, 2015.