Development of the Monitoring system to watch every Computer of Lab Remotely

Vishvajit.W.Dalimbkar
Department of Computer Science & Engineering.
V.V.P Institute of Engineering and Technology,
vishuashta@gmail.com

S.P.Deshmukh
Department of Computer Science & Engineering.
V.V.P Institute of Engineering and Technology,
Shambhuraj.deshmukh@gmail.com

Abstract— **Now a days we cannot monitor each and every student that he/she doing his/her practical's or not. It is not possible to see every student personally. Our aim is to develop software in which we are monitoring the live screens of all computers of the labs remotely and also we can see the status of each and every computer. In Remote Monitoring means system Administrator can secretly watch computers live screens and prevent the students from doing other thing instead of study Practical's. the status of the computer can be seen. With this Functionality it can able to see which system is connected in LAN and which system is not in the LAN. It provides the facility that if computer is connected in LAN then it is indicated by Green Signal and if the computer is not connected in LAN is indicated by red signal and if any computer is partially damaged then that is indicated by yellow signal. Administrator System having the Good Graphical User Interface that help a person to Access remote system easily and without acknowledged to the student.**

Keywords—status detection; live remote client monitoring; secretly monitoring.

I. Introduction

The System will be particularly useful if you are a administrator and want to monitor or control your students or to control your employees from your office, among other uses. From your console you will be able to see the remote screens you select. This system allows you to see live screens of remote computers secretly without disturbing it. This System also maintain the secrete that the person or remote system does not know that someone is monitoring his activities. We are using the client server based architecture. The server program is installed on administrator system and the client program must be installed on each system in the lab. on the remote machine there are no of labs Symbol are given if we select a particular lab then all the systems in the lab are visible on remote machine screen . We can watch the screen of any computer by clicking on it. Additionally you can also see the systems configurations and status of the system that is the system is currently damaged and which is active.Applications of this kind will help the staff/administrator teacher to have control of all computers in a networked lab. In this system we are providing the feature for Remote monitor to secretly watch the computers from the networked labs. The entire solution requires only the computer system to monitor the labs computers. We are using the client server based architecture.The server program is installed on administrator system and the client program must be installed on each system in the lab. on the remote machine there are no. of labs Symbol are given if we select a particular lab then all the systems in the lab are visible on Administrators machine screen. We can watch the screen of any computer by clicking on it. our system does not access the client or students system it just monitor activities performed by the Student secretly. The remote machine or the administrative person can also watch which system is currently active and which is damaged system and which is not active.

II. PROPOSED SYSTEM

We study Desktop Spy Monitor in the spy monitor problem is the it saves the keystrokes, screen shots, websites visit and transfer rate automatically and sends it to the administrator computer.

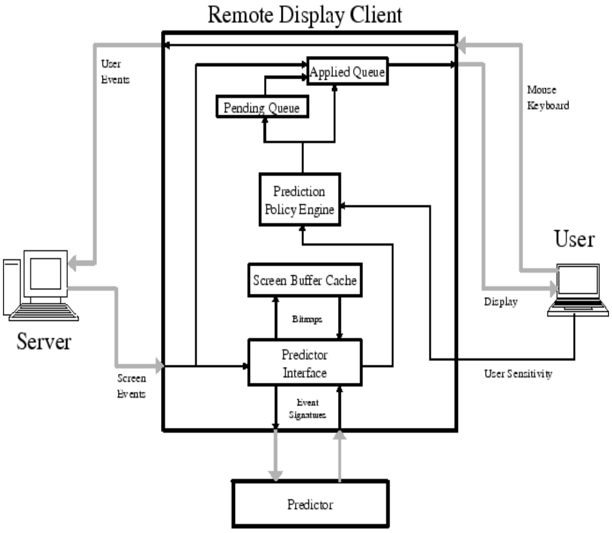
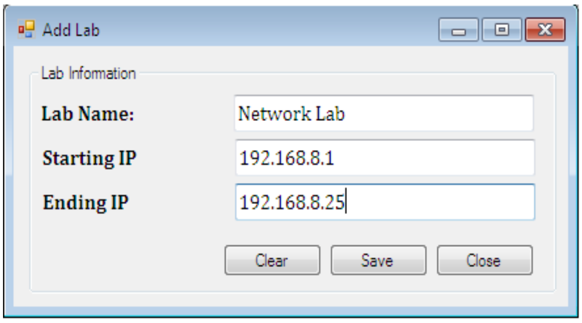


Fig. 1. Block Diagram

In a spy monitor you cannot see the live screen what exactly do the particular student so we are developing a system for removing the disadvantages of the existing system. We are monitoring the live screen of a particular computer connected in a network. And we also showing which computer are working state or which computer is a non working state by providing a signal. When we click a particular labs Symbol are given if we select a particular lab then all the systems in the lab are visible on remote machine screen. We can watch the screen of any computer by clicking on it. Additionally you can also see the systems configurations and status of the system .whether the system is currently active or not. We are give the authority to administrator to see the structure of a particular lab as well as a arrangement of a computer. When administrator has to monitor particular computers it simply click on that computer symbol then particular computer screen is visible on the administrators computer screen without knowing him.

III. OVERVIEW OF SYSTEM

Applications of this kind will help the staff/administrator teacher to have control of all computers in a networked lab. In this system we are providing the feature for Remote monitor to secretly watch the computers from the networked labs. The entire solution requires only the computer system to monitor the labs computers. We are using the client server based architecture. The server program is installed on administrator system and the client program must be installed on each system in the lab.



Lab Information

Lab Name: Network Lab

Starting IP 192.168.8.1

Ending IP 192.168.8.25

Fig. 2. Adding Lab

On the remote machine there are number of labs Symbol are given. if we select a particular lab then all the systems in the lab are visible on Administrators machine screen.

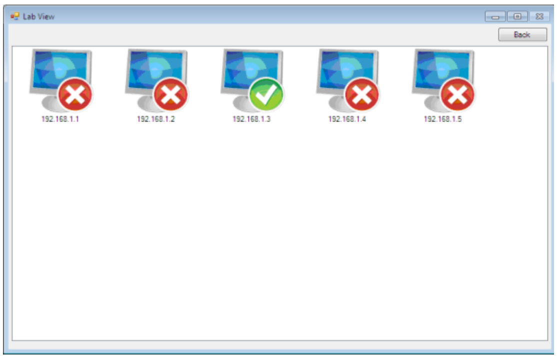


Fig. 3.Overview of lab

We can watch the screen of any computer by clicking on it. our system does not access the client or students system it just monitor activities performed by the Student secretly.The remote machine or the administrative person can also watch which system is currently active and which is damaged system and which is not active.

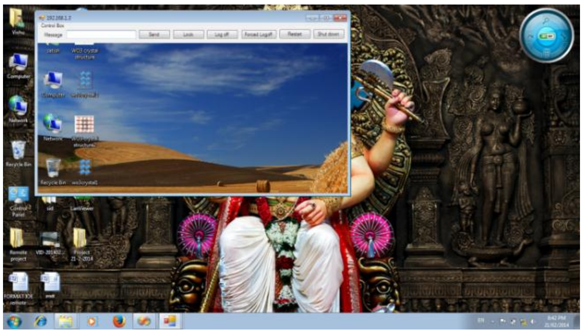


Fig. 4.Client side window

IV. IMPLEMENTATION DETAILS

A. Client Application Development

Remote Client this is used for develop client application. Client program application is run every client computer.this program is placed on the every computer.it take the current snap shot of the live screen and send to the server.this is the background process the client program exe run on a every computer on the background.and the exe file is mount on the strat up menu.

Image Recorder

One of the software features is the option to obtain screenshots. In the next sections we'll describe how it is done.

Create a Screenshot

A screenshot in C# can be easily created: Firstly, we save a bitmap in the size of the screen, then we create a Graphics instance from that image (in orderto restrain the size). The last stages are the copying and saving of the new image. The parameter is saved to create several screenshots.and send a screen shot to the server.

B. Time Intervals

One of the screenshots option is to create one every selected number of minutes defined by the user. achieve that we used time event handling.The first thing to do is creating an instance of the timer:

```
public System.Timers.Timer aTimer;
```

Then we initialize the field Elapsed to a delegate pointing to the screenshots function then we choose the time interval.

C. The Client Side

The communication with the server is done once the user chooses to connect to one such in a given IP address or name in the "Remoting" lab. a different thread is opened to deal with the communication with the server and the control goes to the server.in the client side request port and connectionport is used to communicate to the server.

```
this.Visible = false;
Program.NetworkThread = new Networking();
Thread thread = new Thread(new
ThreadStart(Program.NetworkThread.ListnerFunc));
thread.Start();
```

The thread is set to the function ListnerFunction in the class Networking. The function mentioned handles the requests (usually file transferrequests) sent by the server. The implementationis actually a big switch, for each different message sent by the server.In the client side the server ip address is stored so the client know's that who is the server.

```
ConnectionPort = 9876;

RequestPort = 6789;

 ServerIp = "192.168.1.1";
```

D. Controls used in Server

1. Label

A Label control is used as a display medium for text on Forms. Label control does not participate in user input or capture mouse or keyboard events.

2. Button:

A button accepts clicks. In Windows Forms we use aButton control that accepts click events and performs other actions in the user interface. This control provides a way to accept input and invoke logic based on that input.

3. Text Box

A TextBox control accepts user input on a Form. In this article, I will discuss how to create a TextBox control in Windows Forms at design-time as well as run-time

4. ListView

A ListView control provides an interface to display a list of items using different views including text, small images, and large images.

5. PictureBox

PictureBox provides a rectangular region for an image. It Supports many image formats. It has an adjustable size. It can access image files from your disk or from the Internet. It can resize images in several different ways.

6. Timer

The Timer control allows you to set a time interval to Periodically execute an event at a specified interval. It is useful when you want toexecute certain applications after a certain interval.

V. Working of the RFB protocol

RFB (“*remote frame buffer*”) is a simple protocol for remote access to graphical user interfaces. Because it works at the frame buffer level it is applicable to all windowing systems and applications, including X11, Windows 3.1/95/NT and Macintosh. The remote endpoint where the user sits (i.e. the display plus keyboard and/or pointer) is called the RFB client. The endpoint where changes to the frame buffer originate (i.e. the windowing system and applications) is known as the RFB server.

RFB is truly a “thin client” protocol. The emphasis in the design of the RFB protocol is to make very few requirements of the client. In this way, clients Can run on the widest range of hardware, and the task of implementing a client Is made as simple as possible. The protocol also makes the client stateless. If a client disconnects from a given server and subsequently reconnects to that same server, the state of the user interface is preserved. Furthermore, a different client endpoint can be used to connect to the same RFB server. At the new endpoint, the user will see exactly the same graphical user interface as at the original end point.

VI. TECHNOLOGY USED

.Net allows to retrieve almost any information about local network by using WMI - Windows Management Instrumentation - service with `ManagementObjectSearcher` class and in conjunction with Active Directory Service Interfaces (ADSI) - Active Directory hierarchy.

1.Get Domain Name location computer belongs to ManagementScope sets a scope for management operations - defines the WMI namespace in which management operations are performed. Once domain

name is known, we are able to retrieve list of computers within that domain.

2. Enumerate computers within a domain:Method returns instance of DirectoryEntry - node in in the Active Directory hierarchy. that corresponds to the given domain.

3. Enumerate users within a domain:

Desktop View Capturing Desktop Activity Capturing the activity on the desktop is fairly straight-forward. However, if you want to optimize the performance then you must use some tricks. A key performance metric for remote desktop applications is remote screen refresh rate. Two areas that I will concentrate on to increase refresh rates are:

- The algorithm used to generate the screen capture must be fast.
- The amount of data shuttled across the network must be minimized.

The WCF service is defined using the ViewerService contract. The three methods are:

- Ping – Used most for troubleshooting.
- PushCursorUpdate – Used to push new cursor content (the mouse has moved).

- PushScreenUpdate – Used to push new screen content (the screen has changed).

The Viewer Session class is used to encapsulate the content for each remote client that connects to the viewer. The Viewer Service contains a Dictionary of these sessions. The data flow is as follows:

1. The remote client calls either "push" method via their WCF proxy.
2. The byte array received by the viewer is "unpacked" into image and other metadata.
3. The unpacked data is updated in that client's ViewerSession.
4. The method UpdateScreenImage is called to merge the screen and cursor content.
5. The OnImageChange event is triggered allowing all listeners to update based upon the new data.

The Screen Capture instance provides the features to capture screen and cursor updates. The ViewerServiceClient instance provides the proxy to the WCF service. Updates are pushed to the WCF service via two threads. One is responsible for cursor updates and the other is responsible for screen updates. At times screen updates are fairly bulky. Having a separate cursor thread allows the cursor to be displayed without continuous updates and provides a much smoother viewer experience. The multiple threads do introduce the need for thread safety with respect to the WCF server resources. The following methods provide the push services.

TCP (Transmission Control Protocol)

It is one of the core protocols of the Internet protocol suite. The protocol guarantees reliable and in-order (correct order of packets) delivery of data from sender to receiver. To put it simply, it's reliable. The second aspect of TCP is that it is connection oriented.

VII. CONCLUSION

We introduce Software for monitoring lab's remotely. The application is placed on the Administrator computer. Experiments show that our solution sees monitoring student can only use his computer do something about his works, so the effective work time is longed and the knowledge is improved. Reduce effectively resource waste of your computer. With this monitoring software, student dare not chat, play computer game, or visit some sites at sixes and sevens, which will reduce greatly computer resource waste and thus the whole computers of computer lab run more smoothly.

VIII. REFERENCES

[1] Herbert Schildt,".net Complete Reference", Fifth Edition.

[2] VB.NET Developer's Guide .

[3] A Programmer's Introduction to Visual Basic.NET.

[4] Windows Spy Project Software System Lab,

Department of Electrical Engineering Technion 2008