Information Technology Institute

Java Programming
Group #8

Track: Embedded System
Intake: 42

Fire Alarm System

Presented to: Eng Eman Hesham

Introduction

We need to detect the temperature value of the environment such as the industrial environment or warehouse for the company. Depending on the threshold temperature value we take some actions and show the results on the Graphical User Interface GUI. Graphical User Interface has some features to deal with the event and help him/her to take an action.

Design

- **Hardware**
  we used Arduino NANO with DHT temperature and humidity sensors, Buzzer and Red LED. Scenario, we connect the DHT sensor with the digital input pins to Arduino Nano. Also, we connect the Buzzer and Red LED to the digital output pins to Arduino Nano.

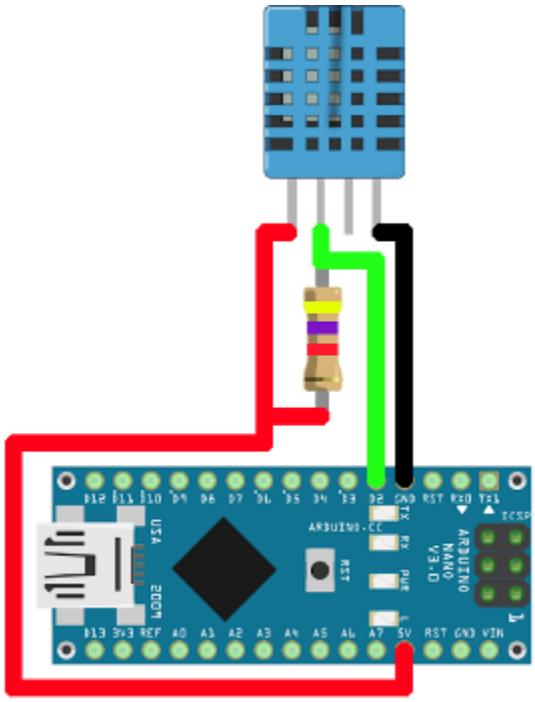

Figure 1: HW components

- **Software**
  We use the Arduino IDE to write the software code for our project. Then we connect the Arduino with NetBeans IDE to build the GUI and interact with it. We write the GUI code by using javaFx.

The Flow

Here we will discuss the flow of the project. This is the GUI and there are some buttons are Start, Test, Stop, Log and Exit. We will take about the role of each one.

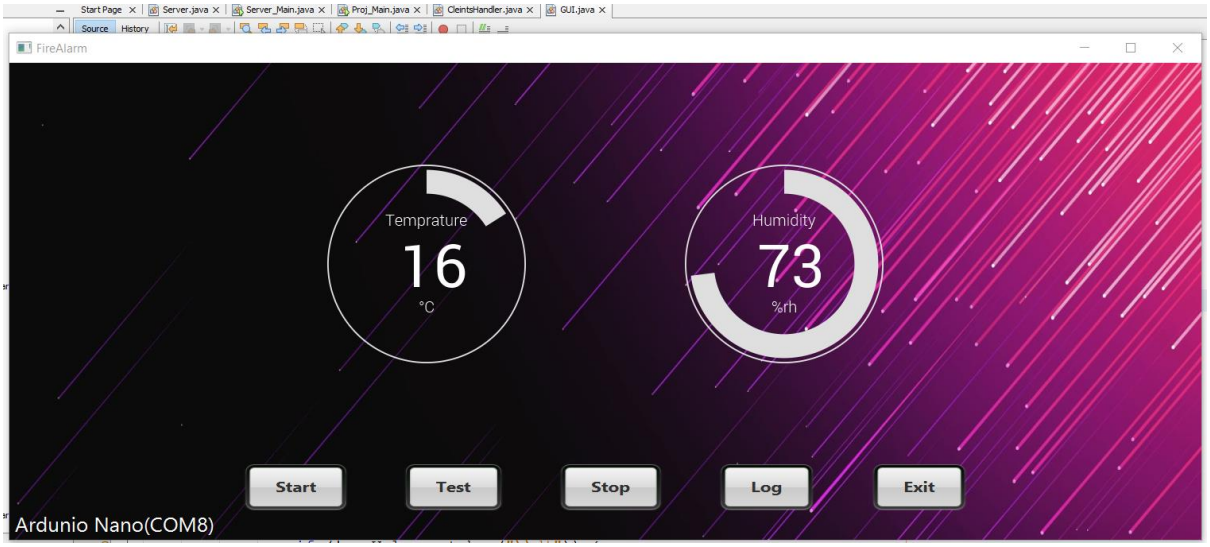

Figure 2: GUI

- **Test button**
  It is used to test the functionalities of the system. Like the alarm function and receiving and sending the data from DHT to GUI. The below figures show the functionalities.

```
/*
 * This function to activate the alarm system
 * by by flashing the buzzer and the Red LED
 */
void Alarm()
{
   digitalWrite(4, HIGH);
   digitalWrite(LED, HIGH);
   delay(500);
   digitalWrite(4, LOW);
   digitalWrite(LED, LOW);
   delay(500);
}
```

```
/*
 * This function will print the value of the temperature
 * and humidity on the Gui or the serial monitor
 * Therefore we will read a char to take an action if we like
 */
void WriteAndRead_SerialMonitor()
{
  /*
   * Printing the value of Temperature
   */
  Serial.println((int)temperature);

  /*
   * Printing the value of Humidity
   */
  Serial.println((int)humidity);
  //it just for receiving a byte from the Gui
  //or the serial monitor to change the state of the system
  incomingByte = Serial.read();
}
```

- **Stop button**
  It is used to stop all the functionalities of the system alarm and receiving and sending data.

- **Start button.**
  Activate the alert functionality after stopping it.

- **Log button**
  It is used to show the history of the reading values of the temperature and humidity.

- **Exit button**
  It is used to exit and stop the system functionalities.

Additional Work

We add some features like:

- **Humidity & Temperature**
  we measure the humidity beside the temperature value.
- **Server & Clients**
  we add a Server to send the data to multiple Clients. In addition to that the devices can control the hardware through the GUI. The results are shown below.

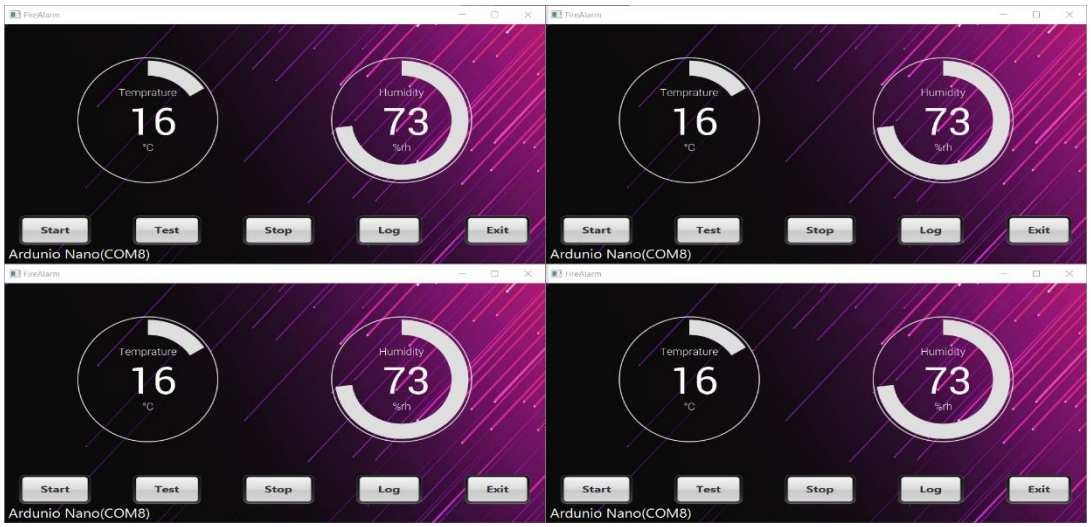

Figure 3: multiple clients connect throw the server

- **Log scene**

To show the history of the reading values of the temperature and humidity.

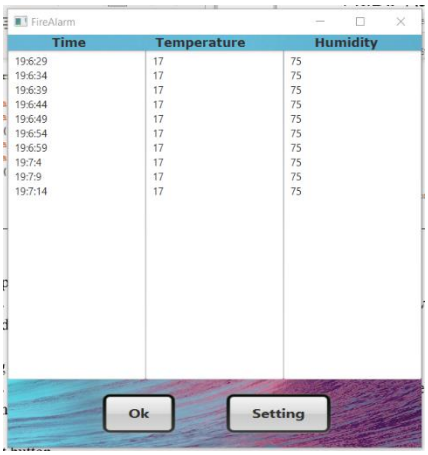

Figure 4:Loge scene

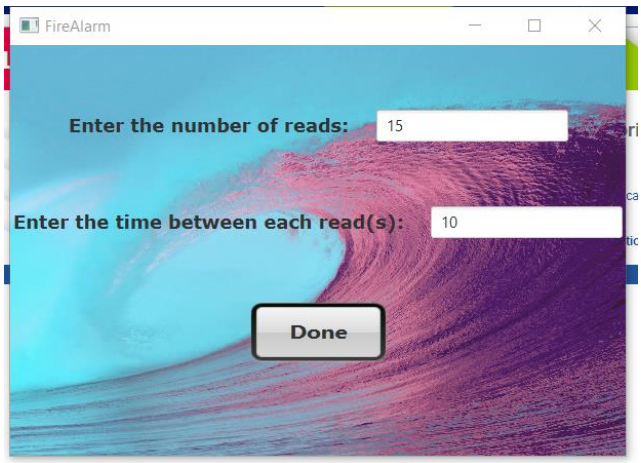

Figure 5: set Settings for the Log scene

- **Handling some scenarios**

  If the server is down or disconnected or If the HW is disconnected. The results are shown below.

- Server:

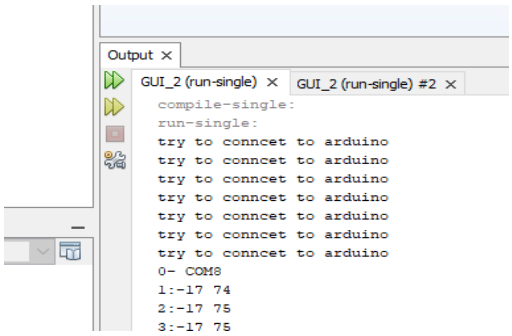

Figure 6: the console message that the server if the Arduino is disconnected

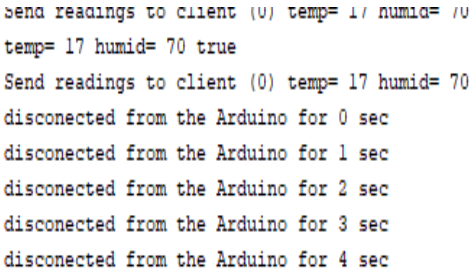

Figure 7: the console message that the server gets if the Arduino is disconnected

- Client:

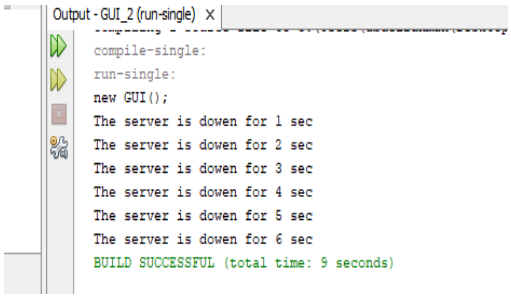

Figure 9: the console message that the client gets if the server is down

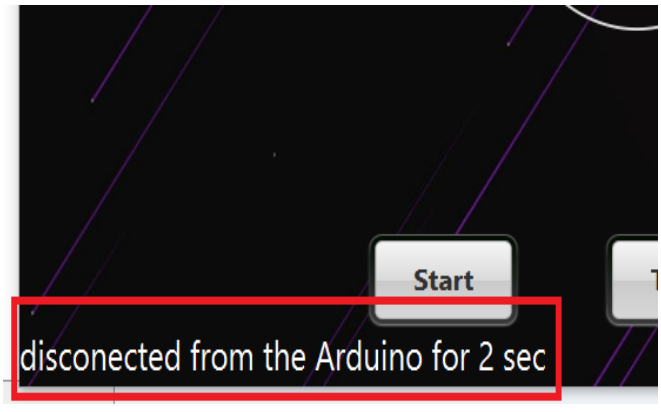

Figure 8: Clients message if the Arduino is disconnected

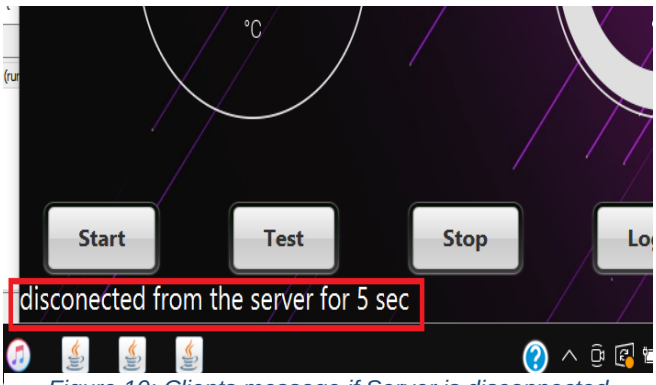

Figure 10: Clients message if Server is disconnected