INTERNET OF THINGS BASED DATA LOGGER SYSTEM FOR TEMPERATURE AND HUMIDITY MONITORING USING MICROCONTROLLER

Adewole Adekunle Olu & Onaneye Abiodun Adefolarin

Computer Engineering Department, Abraham Adesanya Polytechnic, Ijebu-Igbo, Ogun State, Nigeria

Electrical/Electronic Engineering Department, Abraham Adesanya Polytechnic, Ijebu-Igbo, Ogun State, Nigeria

Kunleadewole2008@gmail.com

ABSTRACT

The Internet of Thing (IOT) is communication paradigm, embracing connections between physical objects, real world objects and the virtual world. The Internet of Things is a global network that connects physical objects using cloud computing, web applications, and devices that are network dependent and it permits devices to speak with each other, access, store and retrieve data from the Internet, and interact with users, making intelligent, deep, pervasive and perpetually connected environments. Microcontroller based devices are designed to work with IOT Services. The Microcontroller is a versatile device which cannot solely be used to control devices, however can also be used to read data from all kinds of sensors. In this paper we incorporated the use of microcontroller (ESP8266 WIFI module) which acts as interface for temperature and humidity sensor (DHT11). The data collected from this sensor were sent through the internet to a web server which functions as Sensing as a Service cloud server. The data were logged on remote server in real time. Remote monitoring, management and visualization of sensors data were achieved through any internet connected mobile devices. Whenever sensor's data (temperature and humidity) updated on the cloud server is above certain value, an email alerts were sent to a preconfigured android phone connected to the internet.

Keywords: IOT cloud, Database, Sensors, Cloud computing, Sensing as a Service.

INTRODUCTION

Internet of Things is a communication paradigm, a form of communication between inactive objects, smart object with sensors and any other objects which are not mostly grouped as digital computers. These objects generate, exchange data with minimal or no human intervention. The frontier proposal of internet of things aimed at ensuring every gadgets called objects in everyday lives are connected and alive on the internet. These objects will continuously be sending data to the internet and at times be receiving data, and also be accessible from anywhere from the surface of the earth. This machine to machine communication will become pervasive, immersive and available at every point in human endeavor or activities, in offices, hospitals, schools, farms and any accessible places. In summary Internet of things is an internet of three things. (I) People to people, (II) People to machine (things), (III) Things (machine) to things (machine), interacting through internet (Keyur et al., 2016).

Cloud and IoT are two complementary technologies (Alessio et al., 2014). This new paradigm CloudIoT is expected to disrupt both current and future Internet (Zhou et al., 2013). Using cloud computing, connected devices require online cloud services, this services entail storage of huge amount of data generated by the objects connected to it. Cloud computing has unlimited capabilities in terms of storage and processing power, is a far more matured technology, and has most of the IoT problems partly solved .Cloud computing paradigm ensured data are stored safely. Cloud based platforms help to connect to the things around us so that we can access anything at any time and at any place in a user friendly manner using customized web portals and in built web applications. Hence, cloud acts as a forepart to access Internet of Things. Because applications that act with devices like sensors have special necessities of large storage, storage of huge data, there is need for immense computation power to alter the important real time processing of data, and high speed network to send and receive the data (Rao et al., 2012). These are the basic requirements for any IOT devices.

In everyday, temperature and humidity are important parameters that need to be monitored especially in many places like farms, green houses, hospitals, industries, server rooms, homes and offices. There have been different means devised for measuring temperature and humidity. There are transportable data loggers, this device is designed to accept data from temperature sensors and store the results on externally in EEPROM (Madukar et al.,2014) or store data on SD card for post process analysis or visualization (Ibrahim, 2008). Most data loggers acquire data which may be directly used in a computer only after transferring. This causes drawbacks as data cannot be processed in real time, and at times data could be lost due to theft or corruption.

Internet of things based data logger system for temperature and humidity monitoring using microcontroller can serve as a solution to the problem addressed above. Utilizing the facility of IOT cloud we are able to achieve online real time monitoring. This paper deals with the design and construction of this system using the microcontroller (esp8266) and Arduino platform as the Independent Development Environment (IDE) .The web applications were designed based on these technologies, PHP, MySQL, HTML, Javascript and Jquery programming languages. The web interface was used to visualize the data stored on the web server database in tabular form and also generate the graphical display from this. The design was a complete web-connected solution that shows all the major aspects of the Internet of Things, a project that continuously logs data to a secured IOT cloud.

SYSTEM DESCRIPTION

The Internet of Things (IoT) based Data Logger System for temperature and humidity monitoring are divided into two parts

I. Hardware design
II. Software design

The system which was used to log sensor data (temperature and humidity) to the internet consist of five major parts which are:

I. The sensing unit (DHT11)
II. The microcontroller unit (esp8266)
III. The power supply circuit
IV. Access point (Gateway)
V. Local Server (optional)

The general block diagram of the whole system is shown in fig.1 below.

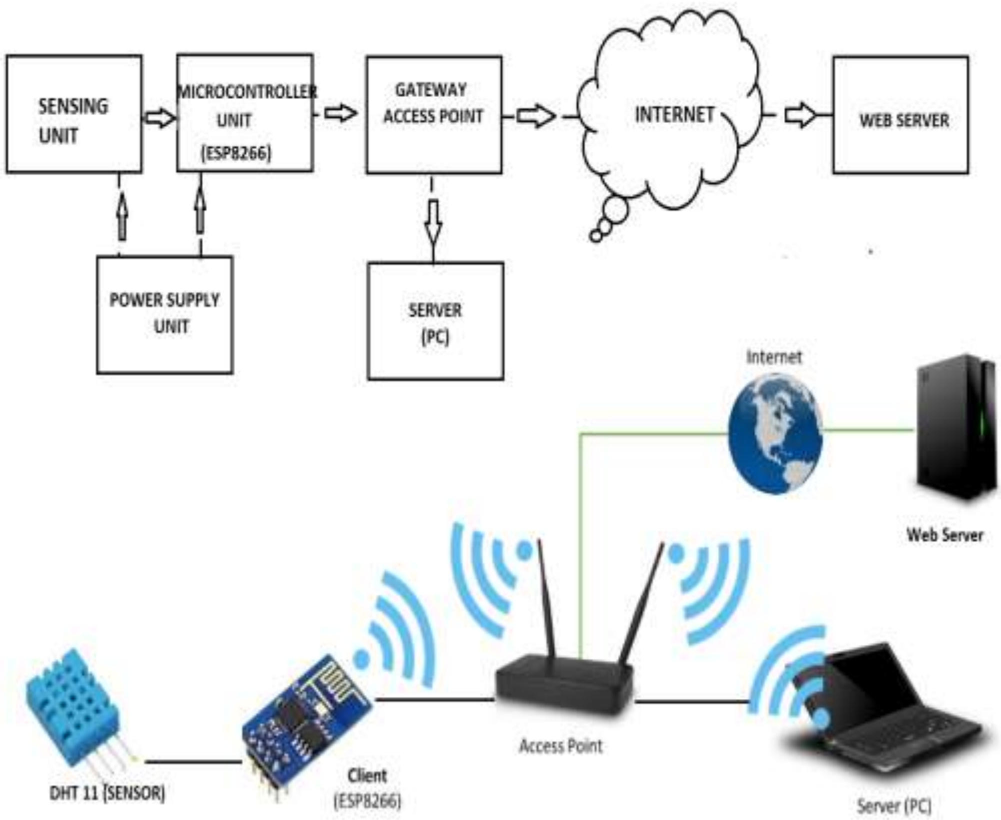


Fig.1 System building block

A. The sensing unit

The detecting unit is a temperature and humidity sensor as a single unit. The sensor DHT11 is a discrete value sensor, which uses a capacitive humidity sensor and a thermistor to gauge the encompassing air, and delivers a digital signal as output. It has a resistive type humidity estimation range of 20-90%RH, 0-50°C with precision value of ±2°C and 1%RH resolution. Its working voltage ranges from 3v to 5v.

B. The microcontroller unit

The ESP8266 is a System on a Chip (SoC) integrated circuit. It consists of a 32-bit microcontroller unit (MCU) and a Wi-Fi transceiver. It has 11 GPIO pins (General Purpose I/O pins), and an analog input. It can be program like any normal Arduino or any other microcontroller. It can communicate through Wi-Fi. it can connect to Wi-Fi network, connect to the Internet, host a web server with real web pages, and it let smart phone connect to it. When ESP8266 is connected into a Wi-Fi network, one IP address is allocated which is accessible in its local network. The ESP8266 can operate in three different modes: Wi-Fi station, Wi-Fi access point, and both at the same time.

There are many different modules available for this microcontroller, standalone modules like the ESP- series with fewer pin (8 pins) which is used in our research project. The main features of ESP8266 microcontroller are as follows; it is 802.11 b/g/n complaint, low power 32-bit MCU, with Integrated 10-bit ADC , TCP/IP protocol stack, and Wi-Fi 2.4 GHz, which support WPA/WPA2.

The digital sensor is interfaced with the digital I/O of ESP8266 microcontroller as shown in figure 2 below. Thus the data collected by the sensor (DHT 11) is transmitted to the microcontroller unit. ESP8266 has transmitting and receiving capability on it serial port. Thus, this port is used when programming using Arduino IDE.

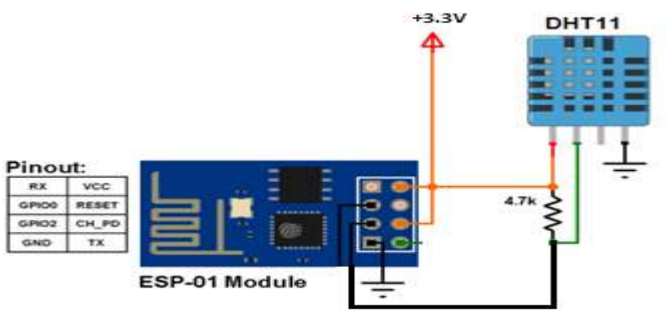


Fig.2 Connection between the digital sensor and the microcontroller

C. The power supply circuit

The power supply section is a regulated DC power supply of +3.3 Volts. The power supply circuit is shown in fig. 3 below. The power requirements of ESP8266 is 80mA, 3.3v. This power requirement is achieved using a step down transformer which convert the alternating voltage of 220v to 6v. The output of the transformer is then rectified using bridge rectifier. The output of this is now fed to the input of voltage regulator (LM1117), whose output is a fixed voltage of 3.3v

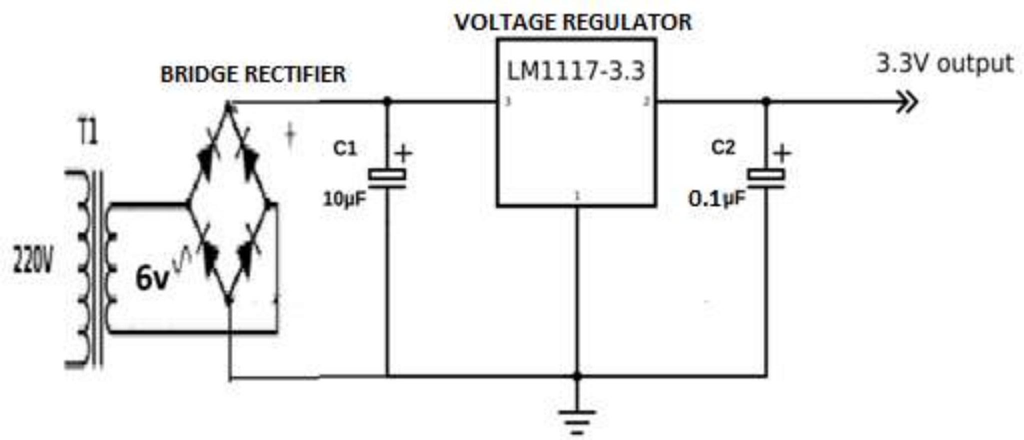


Fig.3 Conversion of 220v AC to 3.3V

D. Access Point

This device perform many different function, such as; it allows other Wi-Fi devices to connect to it or be part of the local network. It has an inbuilt router. The inbuilt router routes IP packets to the right sub-nets so that they will arrive at their destination, for example, if the computer sends a message that is meant for the ESP8266 over the Ethernet sub-net, the router will send the packet to the Wi-Fi sub-net, because it knows that's where the ESP8266 is. If the router is unable to see the addressee on the local network, inbuilt modem, and then to the Internet Service Provider over cable modem or mobile phone, to the Internet, where other routers will endeavor to get the packet to the destination.

E. Web server and web interface

A linux based web server collects data and store it in a database. These data can be accessed from the database from a website hosted on the server. The data from sensor with their time and data of arrival is stored the packet will be sent to the in database. User can interact with the database through web page as shown in figure 6(a) & 6(b).

HARDWARE IMPLEMENTATION

The implementation of the hardware was implemented by constructing the circuit on a breadboard. Following figure 4(a) shows the detail circuit diagram of the IOT based data logger with sensor attached and, Figure 4(b) shows actual hardware implementation.

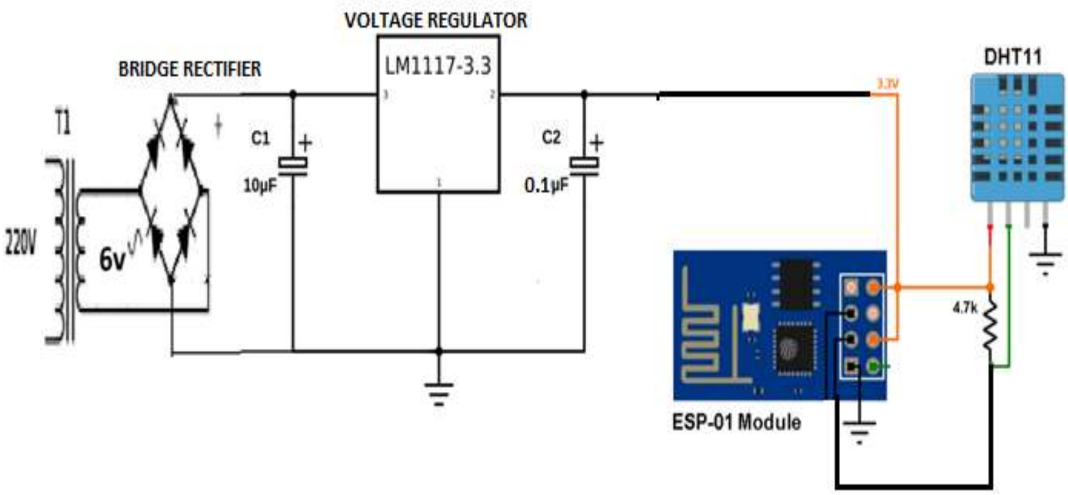


Fig.4a Complete Hardware Circuit

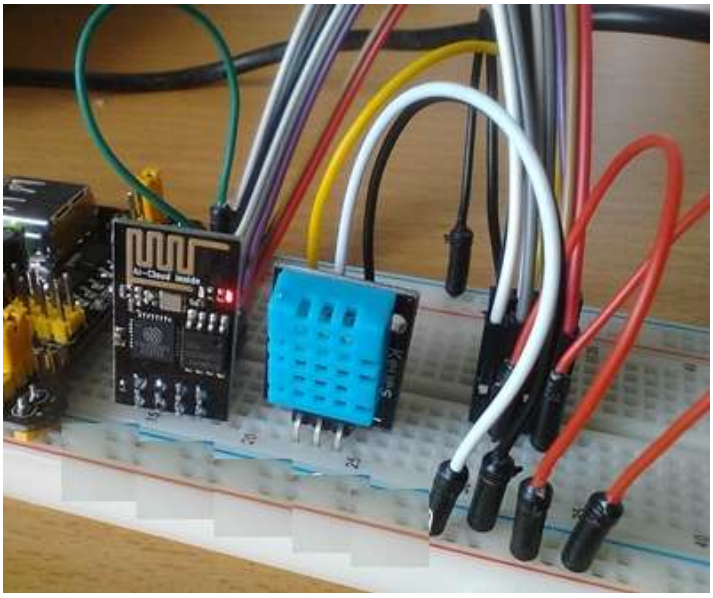


Fig.4b Hardware Implementation

SOFTWARE IMPLEMENTATION

The software implementation plays a major role while retrieving the sensor data and updating it to the web server. Software design includes developing algorithm for the system, for the hardware (client) and web server (IOT cloud). Arduino Integrated Development Environment (IDE) is used for writing and compiling the codes for the Esp8266 microcontroller. Arduino IDE is an embedded programming platform which supports Atmega series of microcontrollers and some other microcontrollers and provides a complete programming environment. Arduino IDE also support programming of the Esp8266 microcontroller.

For development purpose two software tools were used. They are Arduino IDE and web development tools. The control program for Esp8266 was written in Arduino IDE as said earlier, using c language for writing the routine. The web development tools used comprises PHP, MySQL, HTML, Javascript and jquery in a single software package called WAMP server. The WAMP was installed on the local machine for testing the code before it was uploaded to online web server. The web application through which user can interact with the web server required PHP, MySQL, HTML, Javascript and jquery codes for visualization on remote internet connected devices. The general procedure or algorithm for the execution is shown in the flowchart in fig.5a below.

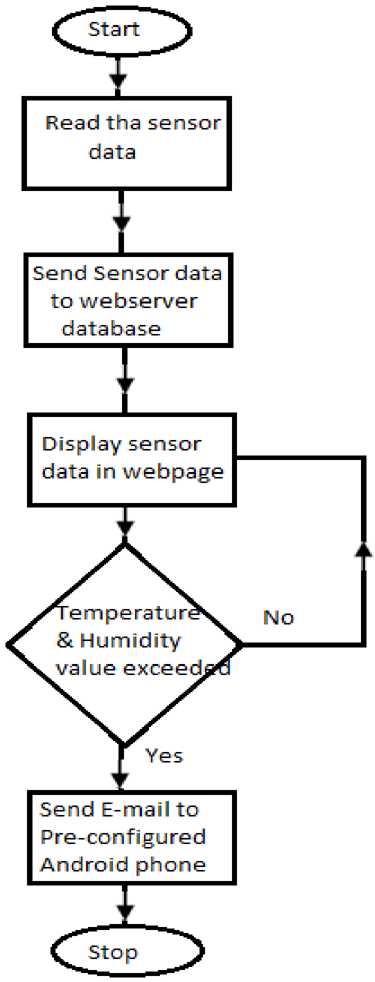


Fig.5a flowchart for the overall process for the system

A. Programming Esp8266 to insert DTH11 sensor data into database

A Program was created to insert DHT11 sensor data into MySQL database using REST api through the microcontroller Esp8266. The program flow for this, is as show in fig.5b. The Steps Involved in Programming Esp8266 in client mode are as follows:

I. Include the software library for DHT11 and Esp8266 Wifi
II. Define the pin that is used to accept sensor data on Esp8266
III. Declare Wifi username
IV. Declare Wifi password
V. Declare the host website e.g www.mysensordata.com
VI. Check for wifi connection and if available connect
VII. If connected obtain an IP address
VIII. Connect to port on database
IX. Send sensor data to web server

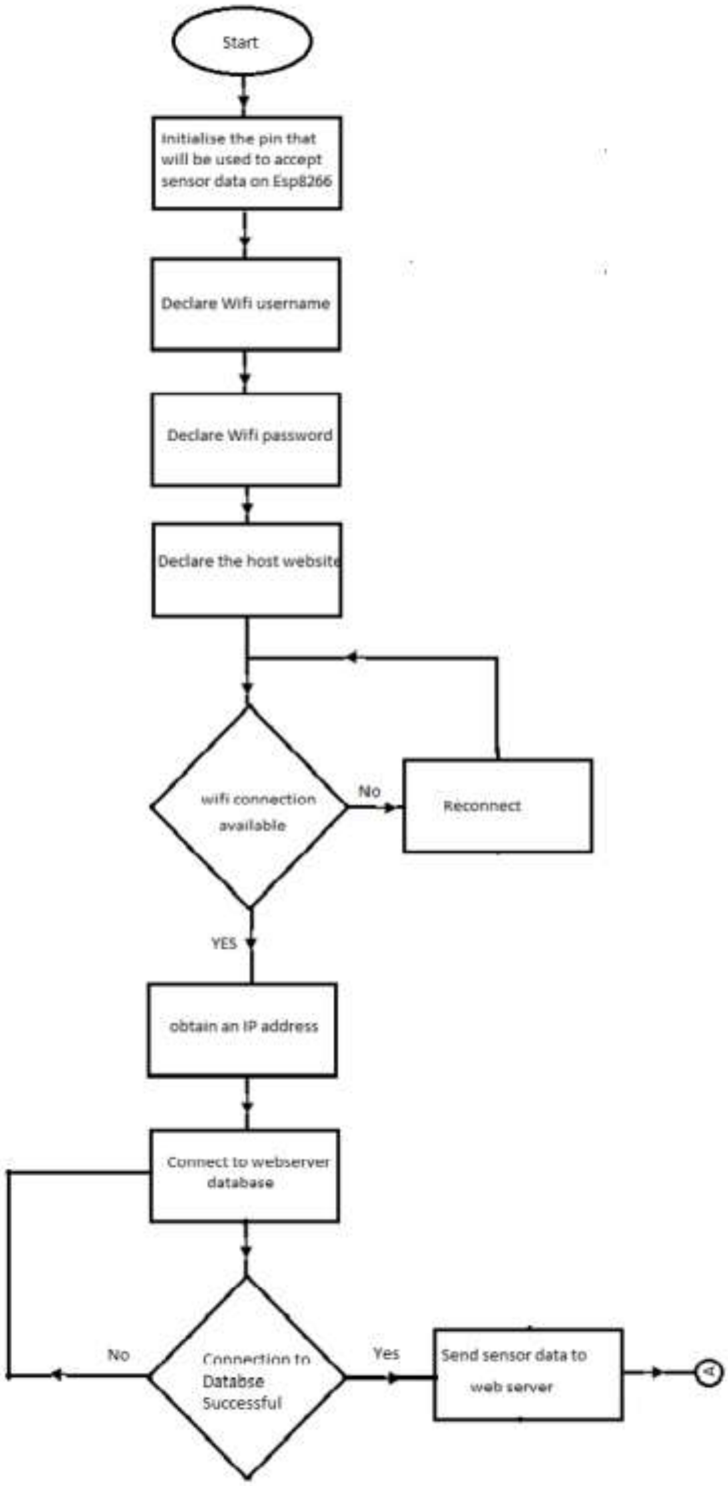


Fig.5b Program Flow Diagram for Esp8266 Microcontroller

B. IOT Cloud programming

The home page of the IOT was created using HTML and CSS, for styling and layout respectively . Since webpage is expected to be dynamic, PHP and Mysql is used for database driven website. MySQL is used to store data inside of database tables. PHP allow manipulation of web page content on the web server.PHP script are stored on the web server, where they are processed. A PHP script runs on the web server and can alter and generate HTML code, therefore web server is able to generate dynamically HTML web pages for database viewing. PHP also stores and retrieve data from the database and incorporate the data into html codes that is generated. In designing the IOT cloud different PHP api were created to insert, delete, read, update data in database. All these scripts were written in PHP programming language. FIG.5c shows the program flow diagram for web server(IOT Cloud) application.

Programming the IOT cloud involves the following steps

Step 1: Settings things up

I. an account was created on web server to host the PHP api, MySQL database and webpages.
II. On the hosting platform, a new MySQL database was created with new password and username.

Step 2: Creating PHP api

I. A new table was created for data channel to insert the temperature and humidity sensor data in the database.
II. PHP REST api was created to insert, delete, read and update data in database.

Step 3: Testing api

I. The api was tested using postman tool to test, insert, delete, read data in databas

STEP 4: Reading temperature and humidity

I. Temperature and humidity data were extracted from the MySQL database using REST api on a web page using HTML, CSS, Javascript and jquery codes.

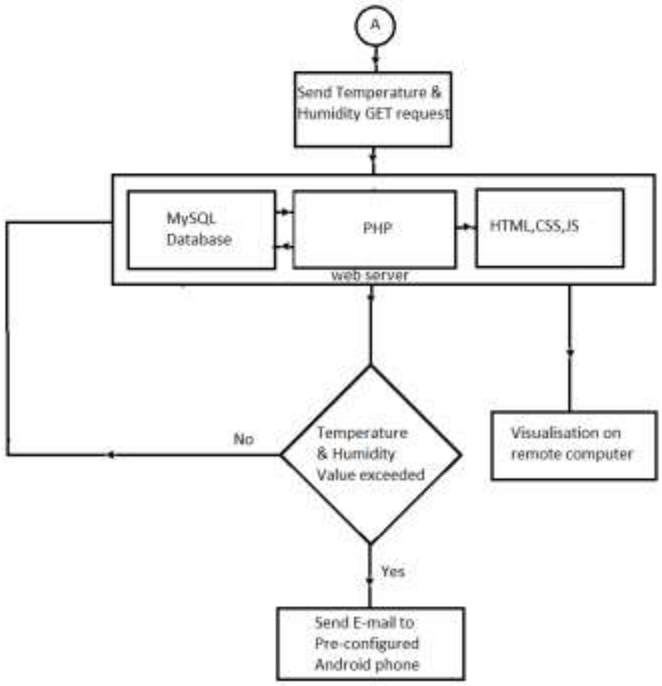
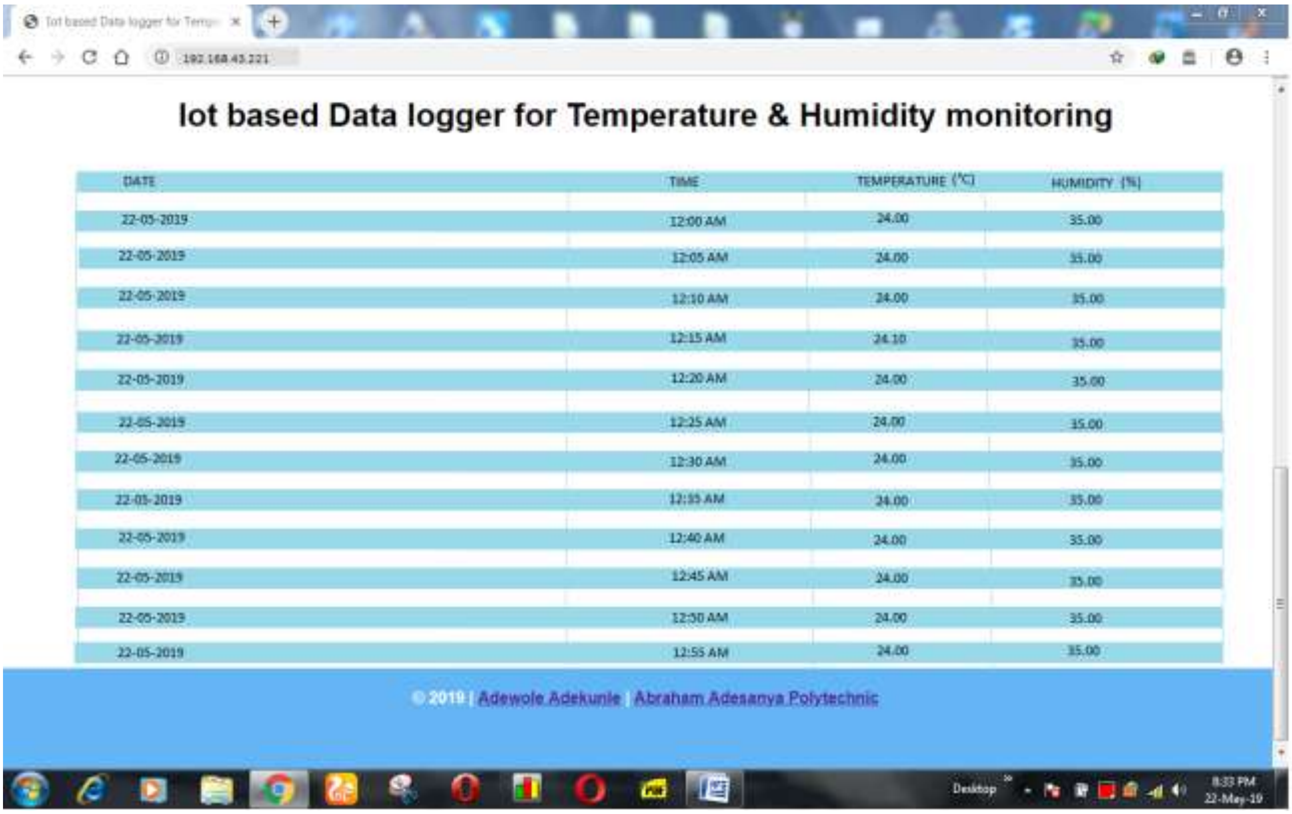


Fig.5c Program Flow Diagram for web server(IOT Cloud) application

RESULTS AND DISCUSSION

The web application on web server collects the data from the sensor and store it in the database. These data can then be viewed through an internet connected computers or mobile devices. The web browser displays the various data, temperature and humidity value in real time, when the user goes to the homepage of the IOT based data logger system for temperature and humidity monitoring, and automatically plots the given data in a graphical representation when a generate graph button is clicked on the web page. The humidity and temperature data captured in the database with time is displayed in a tabular form. The sensor data in the database can be downloaded for offline processing or further analysis if there is need for this. The data can be visualized and analyzed as needed. When the temperature and humidity value exceeded certain limit an E-mail was sent to a preconfigured android phone connected to the internet. The result from this project is shown in fig below 6a & 6b. If one want more storage space to store more data one can own the web server or pay for more domain space on web server. The web page can also be improved upon by designing as per user requirements, which may be complex and costlier. This can be done by professional software developers.



Iot based Data logger for Temperature & Humidity monitoring

DATE	TIME	TEMPERATURE (°C)	HUMIDITY (%)
22-05-2019	12:00 AM	24.00	35.00
22-05-2019	12:05 AM	24.00	35.00
22-05-2019	12:10 AM	24.00	35.00
22-05-2019	12:15 AM	24.10	35.00
22-05-2019	12:20 AM	24.00	35.00
22-05-2019	12:25 AM	24.00	35.00
22-05-2019	12:30 AM	24.00	35.00
22-05-2019	12:35 AM	24.00	35.00
22-05-2019	12:40 AM	24.00	35.00
22-05-2019	12:45 AM	24.00	35.00
22-05-2019	12:50 AM	24.00	35.00
22-05-2019	12:55 AM	24.00	35.00

© 2019 | Adewole Adekunle | Abraham Adesanya Polytechnic

Fig.6a Home page of IOT based data logger showing various data

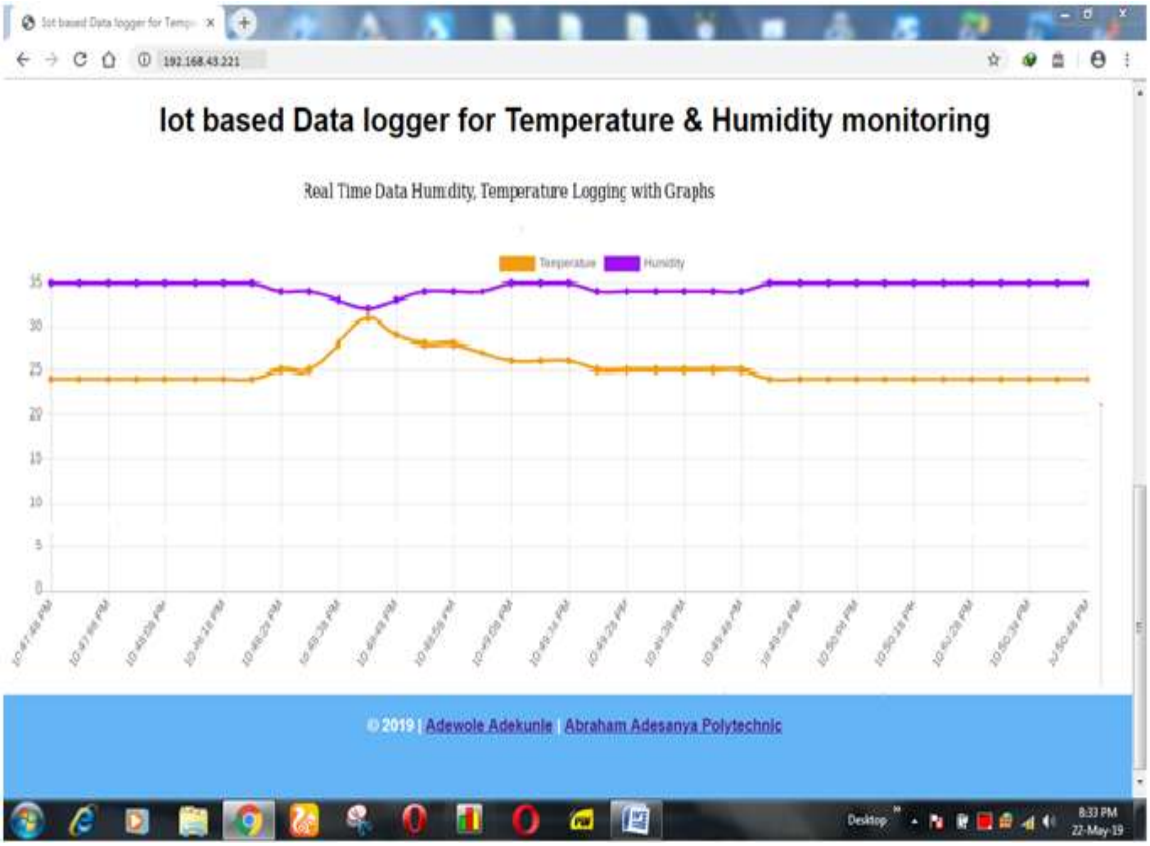


Fig.6b Graphical display of humidity and temperature value

CONCLUSION & RECOMMENDATION

The iot based data logger for temperature and humidity monitoring has been implemented and successfully carried out. It functions as expected, and well executed. The project was successful and we are able to visualize online the values of different temperature and humidity of a room at a particular time, and the days in which this data were capture into the online web database. When the temperature or humidity exceeded certain value the server sent an E-mail alert to a preconfigured Android phone connected to the internet. The data are logged into the database at five minutes interval throughout the day. The logged data information (temperature and humidity) can easily be access globally on a computer or mobile phone with internet services using web browser. The temperature and humidity data stored on the database can be downloaded on local computer for further processing and analysis, if there is need for this. The downloaded data can be imported into microsoft excel worksheet for graphical interpretation to generate graph or chart. Decisions can be made based on this, to know the conditions of the indoor environment where these data were captured.

The advancement in technology has made this research findings to be practical, as there is availability of internet in many places. Thus the project can easily be deployed in many areas where there is need for measuring or monitoring of temperature and humidity in any environment. Further research may improve on the graphical interpretation using online tools on web server instead of downloading the data.

REFERENCES

Alessio B., Walter D., Valerio P., & Antonio P. (2014). "Integration of Cloud computing and Internet of Things: A survey", *Proceedings of the 2014 IEEE 2nd International Conference on Future Internet of Things and Cloud (FiCloud-2014),* Pages: 684-699.

DHT11 Datasheet. Retrieved from http://www.micropik.com/PDF/dht11.pdf

ESP8266 Serial Wifi Transceiver Module Datasheet. Retrieved from https://www.robotshop.com/mediafiles/pdf/esp8266-serial-wifi-transceiver-module-datasheet.pdf

Ibrahim, D. (2008). "Microcontroller And SD-Card Based Multichannel Data Logger". Electronics World. Page 26-31. Retrieved from www.electronicsworld.co.uk

Keyur K., Patel I., & Sunil P. (2016). "IOT Based Data Logger for Monitoring and Controlling Equipment Working Status and Environmental Conditions", *International Journal of Innovative Research in Computer and Communication Engineering*, Pages: 5103 – 5112.

Madukar, S. Chavan, Suryawanshi V.S, Sankpal S. S. (2014). "AVR Microcontroller Based Temperature Control System with Real Time Data Logger", *International Journal of Ad vance Research in Volume 2*, Pages: 51-57.

Rao, B., Saluia, P., Sharma, N., Mittal, A., & Sharma. S. (2012). "Cloud computing for Internet of Things & sensing based applications", *IEEE 2012 Sixth International Conference on Sensing Technology*, pages 374–380.

Zhou, J., Leppanen, T., Harjula, E., Ylianttila, M., Ojala, T., Yu, C., & Jin, H. Yang L. (2013).
"Cloud things: A common architecture for integrating the Internet of Things with

Cloud computing", *Proceedings of the 2013 IEEE 17th International Conference on Computer Supported Cooperative Work in Design (CSCWD),* Pages: 684-699

Reference to this paper should be made as follows: Adewole Adekunle Olu & Onaneye Abiodun Adefolarin (2019), Internet of Things Based Data Logger System for Temperature and Humidity Monitoring using Microcontroller. *J. of Engineering and Applied Scientific Research,* Vol. 11, No. 2, Pp. 35-48