

A Smart Weather Monitoring System Using Internet of Things

Chaw Myat Nwe¹, Zaw Min Min Htun²

^{1,2}Electronic Engineering Department, Mandalay Technological University, Mandalay, Myanmar

Abstract: *The proposed system is an advanced solution for monitoring the weather conditions at a particular place and makes the information visible anywhere in the world. The technology behind this is Internet of Things (IoT). The system deals with monitoring the environmental conditions like temperature, humidity with respect to its measured time with a microcontroller interfaced with sensors and GSM module to send the information wirelessly to remote server and then plot the sensor data as graphical statistics.*

Keywords: Internet of Things (IoT), DHT 11, SIM 900A, Arduino UNO, PHP

1. Introduction

The weather conditions are required to be monitored to maintain the healthy growth in crops and to ensure the safe working environment in industries, etc. IoT device can be used to measure physical parameters pertaining to a physical object and upload them real-time to cloud storage where they can even be analyzed in real-time. Thus, the measured data can be observed from anywhere around the world using Internet-enabled devices. This can make monitoring possible even in difficult geographical terrains. It can also reduce the manpower requirement and thus the risk involved in visiting inhospitable sites.

The primary motivation behind taking up this project is the large utility of the wireless weather monitoring in varied areas ranging from agricultural growth and development to industrial development. The data from the sensors are collected by the microcontroller and also sends the sensors data in to the remote server by using SIM 900A modem. The systems mainly focused on monitor the up or down condition of the room temperature and humidity and send data to a remote server. It is the future technology of connecting the entire world at one place.

The traditional technologies like home automation, wireless sensor networks and control systems work efficiently and smarter due to involvement of IoT. By connecting this weather station to the internet, the IoT can be made much more extensive in predicting and knowing the weather data in particular place.

2. Overall Design

2.1 System Design

The fundamental components of an IoT device are: Control Unit, Power Supply, Input Devices, Output Devices, and Internet Mechanism etc. This system has got almost all things automated so that it get an advantage of the real time direct measurement of the parameters through GSM. Maintaining backup of sent data is easy and can be done within a few seconds. This model uses a DHT11, GSM

module (SIM900A), and an Arduino Uno. The data are stored inside database with timestamp. That data can be seen in visualization by the help of html, JavaScript and PHP language. The data updated from the implemented system can be accessible in the internet from anywhere in the world.

Fig.1 shows the system model of the wireless weather monitoring system.

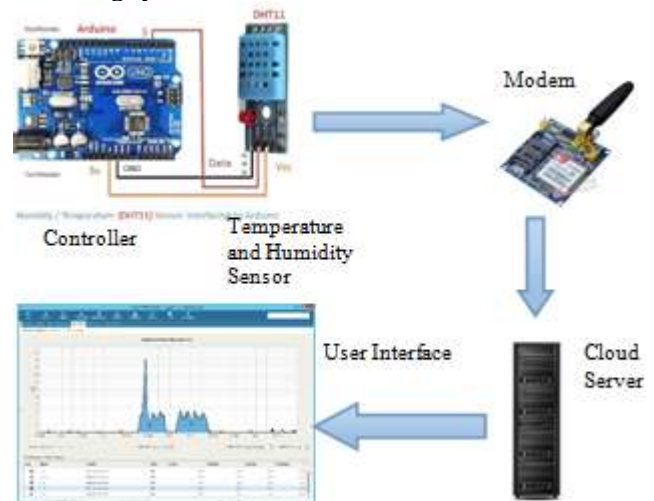


Figure 1: System Model

2.2. Block Diagram Basic

Block diagram of the weather monitoring system is shown in Fig.2. There are two portions in the system. One for weather sensing portion and the other is cloud server and monitoring portion. Humidity and Temperature from DHT11 sensor is to be fetched and sent to the distant server.

The measured data are sent to user with the help of GSM module (SIM 900a) via GPRS. This information is logged inside the database of that server and the server host a website to visualize these data in chart. User can easily monitor the condition of the weather sensed data with their respective time of conditions.

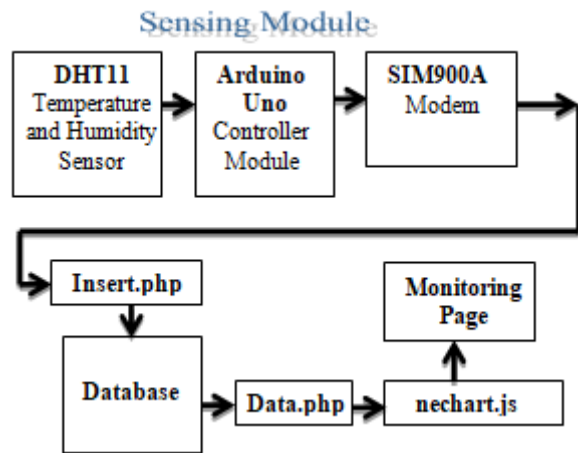


Figure 2: Block Diagram of the Weather Monitoring System

3. Components of the system

3.1. Hardware Components

3.1.1. DHT 11

The DHT 11 is a commonly used Temperature and humidity sensor. The sensor comes with a dedicated NTC to measure temperature and an 8-bit microcontroller to output the values of temperature and humidity as serial data. The sensor is also factory calibrated and hence easy to interface with other microcontrollers. The sensor can measure temperature from 0°C to 50°C and humidity from 20% to 90% with an accuracy of $\pm 1^\circ\text{C}$ and $\pm 1\%$. So if we are looking to measure in this range then this sensor might be the right choice [2].

3.1.2. GSM 900A

This GPRS/GSM module is a breakout board and minimum system of SIM900A Dual-band GSM/GPRS module. It can communicate with controllers via AT commands (GSM 07.07 ,07.05 and SIMCOM enhanced AT Commands). The interface is via UART/Serial. This module supports software power on and reset.

3.1.3. ARDUINO UNO

The Arduino UNO is a widely used open-source microcontroller board based on the ATmega328P microcontroller and developed by Arduino.cc. The board is equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits. The board features 14 Digital pins and 6 Analog pins. It is programmable with the Arduino IDE (Integrated Development Environment) via a type B USB cable. It can be powered by a USB cable or by an external 9 volt battery, though it accepts voltages between 7 and 20 volts. It features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter. The Arduino UNO is generally considered the most user-friendly and popular board or the Arduino board series [3].

3.2. Software Components

3.2.1. Arduino IDE

The Arduino integrated development environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in the programming language Java. It originated from the IDE for the languages Processing and Wiring. The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub `main()` into an executable cyclic executive program with the GNU tool chain, also included with the IDE distribution. The Arduino IDE employs the program `avrdude` to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware [1].

3.2.2 NetBeans IDE

NetBeans is an integrated development environment (IDE) for Java. NetBeans allows applications to be developed from a set of modular software components called modules. Applications based on NetBeans, including the NetBeans IDE, can be extended by third party developers [1].

3.2.3 Hypertext Preprocessor (or simply PHP)

PHP code may be embedded into HTML code, or it can be used in combination with various web template systems, web content management systems, and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications [6].

4. Software Design and Results of the System

The SIM900A GSM Module is turned into GPRS mode and commanded to make HTTP GET to `insert.php` file on the server by the following AT commands.

- AT+CSQ - To enquire signal strength
- AT+CGATT? - To check the status of Packet service attaches.
- '0' implies device is not attached and '1' implies device is attached.
- AT+SAPBR=3,1,"Contype","GPRS" - To see the connection type
- AT+SAPBR=3,1,"APN","CMNET" - To fill in local APN server
- AT+SAPBR=1,1 - To enable GPRS
- AT+CIFSR - To get the ip address
- AT+HTTPINIT - To initialize HTTP request
- AT+HTTPPARA - To prepare the http parameters
- AT+HTTPACTION - To submit http get request after initializing and preparing http parameters
- AT+HTTPREAD - To read the http response from the server

4.1. Data Collecting Process

Temperature and Humidity were sensed by DHT11 sensor and insert the data with Myanmar Standard Timestamp into Table. The flowchat of data storing process was shown in Fig.3.

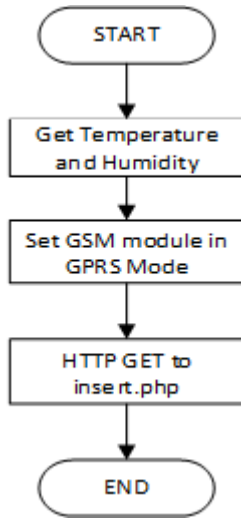


Figure 3: Flowchart for Data Storing Process

Firstly we need to define host, user, password and database. And then connect to the database by using defined values. If there is no error, humidity and temperature values with Myanmar standard timestamp are put inside the table by using HTTP GET REQUEST. The short code is shown in Fig.4.

```

6 $mysql = mysqli_connect($host, $user, $password, $database, 3306);
7- if (mysqli_connect_errno()){
8     printf("Connection failed: %s\n", mysqli_connect_error());
9     exit();
10- }else{
11     $humidity= $_GET['humidity'];
12     $temperature= $_GET['temperature'];
13     $sql = "insert into sensor1 values (CONVERT_TZ(CURRENT_TIMESTAMP, '+06:00', '+06:30'), '$humidity', '$temperature')";
14     $res = mysqli_query($mysql, $sql);
15- if ($res==TRUE){
16     echo "A record has been inserted.";
17- }else{
18     printf("Could not insert record: %s\n", mysqli_error($mysql));
19     }
20 }
    
```

Figure 4: Codes used for Data Storing Process

4.2. Data Monitoring Process

Fig.5 shows the flow chart for System Monitoring. It was direct to the source of JavaScript for jquery, jchar.min.js and calling javascript file to draw chat.

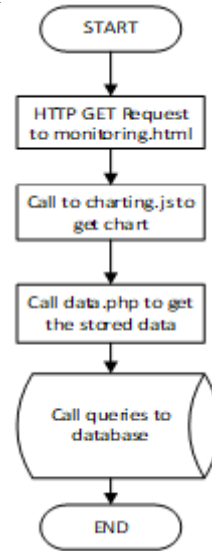


Figure 5: Flow chart for System Monitoring

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>Humidity and Temperature Monitoring </title>
5
6 <link href="css/default.css" rel="stylesheet">
7 <meta charset="utf-8"/>
8 </head>
9 <body>
10
11 <div class="chart-container" style="width: 700px">
12 <canvas id="chartcanvas"></canvas>
13 </div>
14
15 <!-- javascript -->
16 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
17 <script src="js/chart.min.js"></script>
18 <script src="js/line-db.php.js"></script>
19
20 </body>
21 </html>
    
```

Figure 6: Code used for System Monitoring

Inside the chart generator javascript file, define and push the data array of humidity and temperature values and its timestamps values for chart. Fig.7. shows the stored database in remote cloud server.

Field	Type	Null	Key	Default	Extra
TStamp	timestamp	NO	PRI	CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP
humidity	varchar(10)	YES		NULL	
temperature	varchar(10)	YES		NULL	

Figure 7: Database Table of the Environmental Datas

The chart generated to monitor the humidity and temperature values with respect to its timestamp is shown in Fig.8. The user can easily examine the last six values of the sensed data everyplace the internet can available.

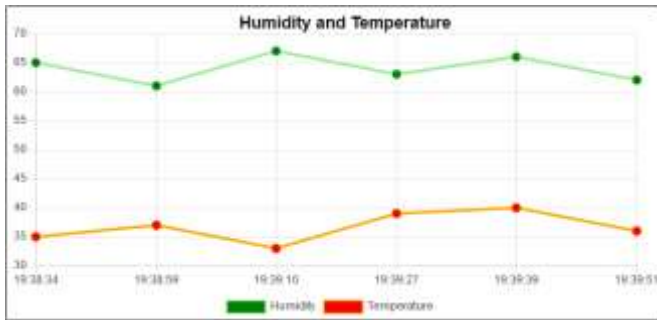


Figure 8: GUI Design for Weather Monitoring System



Zaw Min Min Hyun received the B.E. degrees in Electronic Engineering in 2005 from Mandalay Technological University and M.E. degree in 2012 from Bauman Moscow State Technical University. He is now the researcher for doctoral degree in Electronic Engineering. He is now doing his research related to wireless and arduino control.

5. Conclusion

This paper presents the research and implementation of a system for monitoring the environmental parameters using IoT scenario is accomplished. As the applications are limitless, other weather parameters can also be monitored easily with the addition of related sensors to the system architecture. The data can be stored online, which can be used to forecast weather and eventually analyze climate patterns, as well as for other meteorological purposes. The system is tested in an indoor environment and it is successfully updated the weather conditions from sensor data. It is also a less expensive solution due to usage of low power wired sensors and GPRS module.

6. Acknowledgment

The author would like to express my gratitude to all colleagues at Mandalay Technological University who have contributed to the preparation of this research work.

References

- [1] Charalampos Doukas, "Building Internet of Things with the Arduino", CreateSpace Publications, 2012
- [2] Robotics D, DHT11 Humidity & Temperature Sensor, 2010
- [3] Arduino Uno R3 Datasheet, Available: <http://arduino.cc/en/Main/arduinoBoardUno>
- [4] Susmitha P and Bala G.S, Design and Implementation of Weather Monitoring and Controlling System, International journal of Computer applications, 97 (3), 2014, 19-22.
- [5] "Design and Construction of a Low Cost Digital Weather Station" paper by Lanre Joseph Olatomiwa and Umoru Sam Adikwu from Federal University of Technology, Nigeria.
- [6] PHP, Available: <https://en.wikipedia.org/wiki/PHP>.

Author Profile



Chaw Myat Nwe received the B.E. and M.E. degrees in Electronic Engineering from Mandalay Technological University in 2000 and 2002, and doctoral degree in Electronic Engineering from Yangon Technological University in 2004 respectively. She is now working as a teaching staff in Division of Communication Engineering, Mandalay Technological University. She is now doing and supervising research related to wireless communication.