Use of IoT to Monitor Self-Isolated SARS-COV-2 Patients

Shirin Sultana Shaik
Master's in Computer Science
University of Massachusetts, Lowell
Lowell, USA
shirinsultana_shaik@student.uml.edu

***Abstract*— COVID-19 - the pandemic that hit all over the world in 2020 turned into a challenge for all medical and front-line workers around the globe. Even with no symptoms, virus-infected individuals were required to isolate themselves to keep the virus from spreading. This affected patient care as they were independent without proper supervision from a medical practitioner. Remote supervised monitoring can reduce patient visits to the hospital and the spread of the virus, reducing the workload of medical staff. Infected individuals might benefit significantly from medical surveillance systems built on the Internet of Things (IoT) that could be used to track and stop the spread of the virus. This study proposes an IoT-based device to remotely monitor the patient's physiological data like blood pressure, heartbeat, and glucose level with the help of wireless body sensor networks and a gateway that can be used for data acquisition and transmission. This paper intends to construct a system that can show the results of capturing the conditions and locations of several patients in real-time using the ThingSpeak open-source software program for the sake of COVID-19 patient care. This technology would help to capture real-time data and other necessary information from various locations of the infected patients. ThingSpeak is an IoT analytics platform service that allows you to aggregate, visualize and analyze live data streams in the cloud, as it can provide instant visualizations of data posted by your devices to the platform.**

Keywords*— *remote medical surveillance, IoT-based sensors, COVID-19 patient care, Internet-of-Things, real-time

I. INTRODUCTION

In March 2020, the World Health Organization (WHO) announced the spread of the COVID-19 virus as a pandemic, indicating that it has spread globally and the number of infected patients is increasing rapidly. The Internet of Things (IoT) is a technology that can provide patients with a remote monitoring system to measure physiological parameters such as blood pressure, heart rate, and glucose level. Therefore, applying this technology can significantly reduce the burden on medical personnel while enhancing their efficiency. This technology will help to capture real-time data and other necessary information from various locations of the infected individuals. In addition, the availability of location information will make it easier for physicians and paramedics to assess the appropriate course of action and assist patients in doing required tasks autonomously. Accordingly, analyzing and displaying accurate patient location data with sensor measurement results on a single dashboard is necessary. This is feasible with the help of ThingSpeak, a cloud-based IoT analytics platform service that enables the collection, visualization, and analysis of live data feed. ThingSpeak provides instant visualizations of data posted by your devices to the platform.

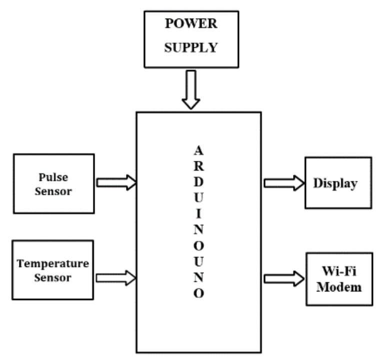

Fig. 1. Block Diagram of how Monitoring with IoT is done

II. THE PROPOSED SYSTEM SPECIFICATIONS

A. Hardware

1) Arduino Uno

Fig. 2. a. Arduino Uno R3 Front ; b. Arduino Uno SMD

2) Communication

The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1). A Software Serial library allows for serial communication on any of Uno's digital pins.

3) AI thinker Wi-Fi module

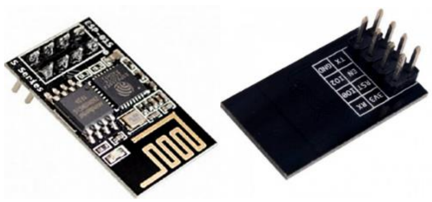

Fig. 3. AI thinker Wi-Fi module

4) Pulse sensor module

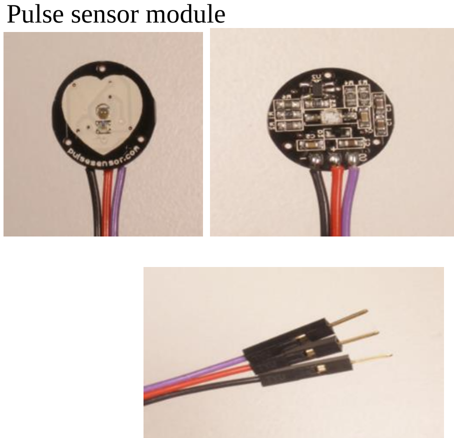

Fig. 4. Pulse sensor module

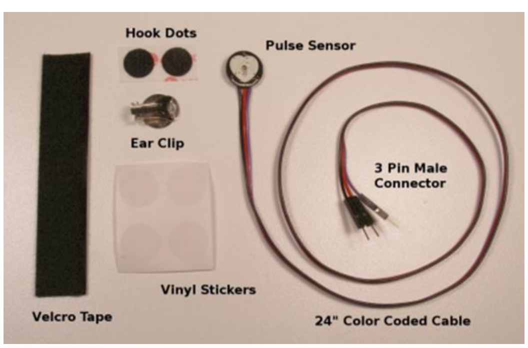

Fig. 5. Pulse sensor kit

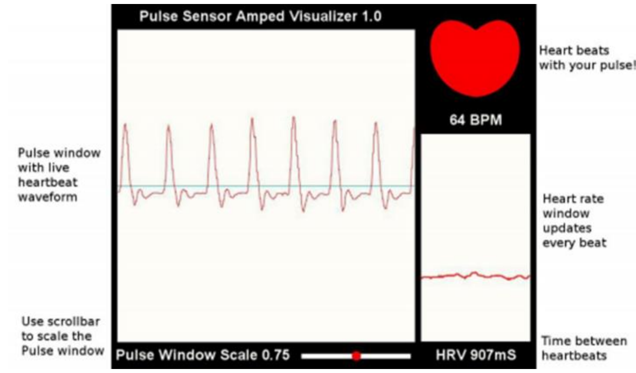

Fig. 6. Data visualization software

5) Temperature sensor module

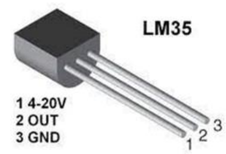

Fig. 7. Temperature sensor

6) Adaptor

Fig. 8. 12v lamp adaptor

B. *Software*

1) Arduino IDE

The Arduino Integrated Development Environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in functions from C and C++. The open-source Arduino Software (IDE) makes writing and uploading code to the board easy. This software can be used with any Arduino board. It is used to write and upload programs to Arduino-compatible boards and, with the help of third-party cores, other vendor development boards. The Arduino IDE supplies a software library from the Wiring project, which provides many standard input and output procedures.

2) ThingSpeak

ThingSpeak is an open-source Internet of Things application and API to store and retrieve data from things using the HTTP and MQTT protocol over the Internet or via a Local Area Network. You can also analyze and visualize your data with MATLAB or other software, including making your applications.

III. THE PROPOSED METHODOLOGY

A. *Working*

The real-time health monitoring system is comprised of an application server that can be used for data storage, analytics, and visualization, a pulse sensor for measuring heart rate, and a temperature sensor connected to an Arduino microprocessor, which precisely measures the physiological parameters of the patients, including their body temperature, heart rate, ECG, SpO2 levels, blood pressure, and glucose levels. The readings are then communicated to an Android application on a smartphone through a Bluetooth module, which acts as an IoT gateway to gather sensor data and add location information before uploading the information to the server. These sensors measure analog data sorted and amplified by the microprocessor. Physicians may also diagnose illnesses by using the data obtained remotely from the patient. An Android-based mobile application that communicates with a web application is developed for effective patient-physician real-time communication.

This architectural design enables a location-tracking system to collect data from various sensors within the patient's body and then transmit the measurement data to the IoT gateway. The sensor, which may consist of many measuring devices, is linked to an Android device that serves as an Internet of Things gateway.

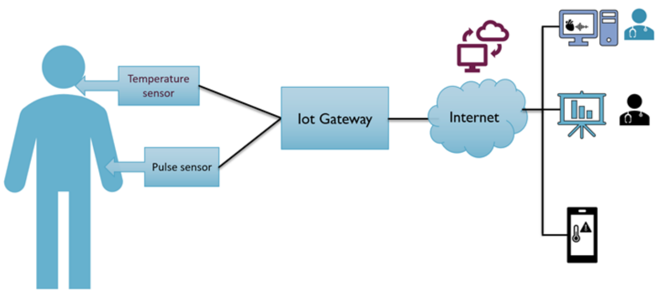

Fig. 9. Flow diagram of the proposed methodology

B. *Wireless Body Sensor Networks (WBSN)*

Measurement of Heart rate levels is one of the measurement points that must be carried out in routine examination procedures for a self-isolation patient with COVID-19. Measuring heart rate at home can be used to detect “silent hypoxemia” early in COVID-19 patients. The data from the heart rate measurement results in this study were obtained from direct measurements using a pulse sensor as part of the wireless body sensor networks (WBSN). In addition, the location data sent along is obtained from an Android device with GPS and Location API. The proposed system integrates a wireless sensor with a smartphone application that acts as an IoT gateway where the sensor can collect the user’s physiological parameters. The Android application connects to the internet to send the data to the server after adding the patient’s ID and location coordinates. The system was built using open-source ThingSpeak software for data storing and indexing. The sensor used for testing was a home pulse oximeter available in the market. The sensor device and IoT gateway connect using Bluetooth standard communication protocol.

C. *IoT Gateway*

After the data is received from the sensor and collected on Android, the mobile application will send the data to the server with the patient ID and location coordinates. The delivery process can take place in real-time or be delayed according to the availability of the internet connection. Application on Android devices connects to applications on the server using HTTP or HTTPS connection.

D. *Configuring ThingSpeak*

This cloud platform may be utilized to monitor a patient's submitted physiological data, which is done by inputting all the recorded values on the "My Channels" page of the software. Healthcare practitioners can access all the patient data from the charts created upon inputting readings and creating a channel in the software.

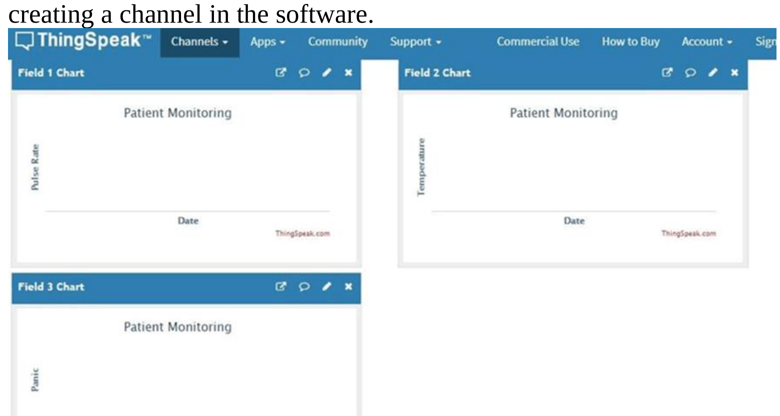

Fig. 10. Patient data in charts on ThingSpeak software

G. *Code to configure the setup*

```
#define USE_ARDUINO_INTERRUPTS true
#define DEBUG true
#define SSID "sla"     // "SSID-WiFiname"
#define PASS "fabian69" // "password"
#define IP "184.106.153.149"      // thingspeak.com ip

#include <SoftwareSerial.h>
#include "Timer.h"
#include <PulseSensorPlayground.h>     // Includes the PulseSensorPlayground Library.
Timer t;
PulseSensorPlayground pulseSensor;

String msg = "GET /update?key=8ZY29TXZ7XG5180Q";
SoftwareSerial esp8266(10,11);

//Variables
const int PulseWire = A0;       // PulseSensor PURPLE WIRE connected to ANALOG PIN 0
const int LED13 = 13;          // The on-board Arduino LED, close to PIN 13.
int Threshold = 550;           //for heart rate sensor
float myTemp;
int myBPM;
String BPM;
String temp;
int error;
int raw_myTemp;
float Voltage;
float tempC;
void setup()
{
   Serial.begin(9600);
  esp8266.begin(115200);
  pulseSensor.analogInput(PulseWire);
  pulseSensor.blinkOnPulse(LED13);       //auto-magically blink Arduino's LED with heartbeat.
  pulseSensor.setThreshold(Threshold);
 // Double-check the "pulseSensor" object was created and "began" seeing a signal.
   if (pulseSensor.begin()) {
    Serial.println("We created a pulseSensor Object !");  //This prints one time at Arduino power-up,  or on Arduino reset.
  }
  Serial.println("AT");
  esp8266.println("AT");
  delay(3000);
  if(esp8266.find("OK"))
  {
 connectWiFi();
  }
  t.every(10000, getReadings);
   t.every(10000, updateInfo);
}

void loop()
{

start: //label
    error=0;
   t.update();
    //Resend if transmission is not completed
    if (error==1)
    {
      goto start; //go to label "start"
    }
 delay(4000);
}
```

```
    delay(4000);
}

void updateInfo()
{
  String cmd = "AT+CIPSTART=\"TCP\",\"";
  cmd += IP;
  cmd += "\",80";
  Serial.println(cmd);
  esp8266.println(cmd);
  delay(2000);
  if(esp8266.find("Error"))
  {
    return;
  }
  cmd = msg ;
  cmd += "&field1=";    //field 1 for BPM
  cmd += BPM;
  cmd += "&field2=";  //field 2 for temperature
  cmd += temp;
  cmd += "\r\n";
  Serial.print("AT+CIPSEND=");
  esp8266.print("AT+CIPSEND=");
  Serial.println(cmd.length());
  esp8266.println(cmd.length());
  if(esp8266.find(">"))
  {
    Serial.print(cmd);
    esp8266.print(cmd);
  }
  else
  {
    Serial.println("AT+CIPCLOSE");
    esp8266.println("AT+CIPCLOSE");
    //Resend...
    error=1;
  }
}

boolean connectWiFi()
{
  Serial.println("AT+CWMODE=1");
  esp8266.println("AT+CWMODE=1");
  delay(2000);
  String cmd="AT+CWJAP=\"";
  cmd+=SSID;
  cmd+="\",\"";
  cmd+=PASS;
  cmd+="\"";
  Serial.println(cmd);
  esp8266.println(cmd);
  delay(5000);
  if(esp8266.find("OK"))
  {
    return true;
  }
  else
  {
    return false;
  }
}

void getReadings(){
```

```
  cmd+="\"";
  Serial.println(cmd);
  esp8266.println(cmd);
  delay(5000);
  if(esp8266.find("OK"))
  {
    return true;
  }
  else
  {
    return false;
  }
}

void getReadings(){
  raw_myTemp = analogRead(A1);
  Voltage = (raw_myTemp / 1023.0) * 5000; // 5000 to get millivots.
  tempC = Voltage * 0.1;
  myTemp = (tempC * 1.8) + 32; // conver to F
  Serial.println(myTemp);
  int myBPM = pulseSensor.getBeatsPerMinute();  // Calls function on our pulseSensor object that returns BPM as an "int"
                                                 // "myBPM" hold this BPM value now.
if (pulseSensor.sawStartOfBeat()) {              // Constantly test to see if "a beat happened".
Serial.println(myBPM);                          // Print the value inside of myBPM.
}

  delay(20);
    char buffer1[10];
     char buffer2[10];
    BPM = dtostrf(myBPM, 4, 1, buffer1);
    temp = dtostrf(myTemp, 4, 1, buffer2);
  }
```

Fig. 11. Code to set up the proposed system

E. Working Steps

Step 1: Open Arduino IDE and type the required code

Step 2: Put the API key of THINKSPEAK in the Arduino code to link the Arduino and THINKSPEAK.

Step 3: Connect the Arduino Uno with the Arduino cable to the PC.

Step 4: Hold the pulse sensor and temperature sensor at the same time.

Step 5: Check the serial monitor for the output.

Step 6: Now open ThingSpeak and the patient's readings can be seen on the charts.

IV. OBSERVATIONS

A. System Hardware Model

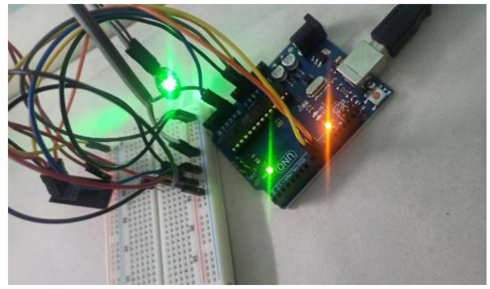

Fig. 12. Sensors connected to Arduino Uno

B. Theoretical Readings

Fig. 13. Body temperature and pulse rate readings along with time and location

V. CONCLUSION

In the current scenario, advanced information technologies have opened a new door to innovation in our daily lives. Out of these information technologies, the Internet of Things is an emerging technology that provides enhancement and better solutions in the medical field, like proper medical record-keeping, sampling, integration of devices, and causes of diseases. IoT's sensor-based technology provides an excellent capability to reduce the risk of surgery during complicated cases and is helpful for the COVID-19 type pandemic. In the medical field, IoT's focus is to help treat different COVID-19 issues precisely. It makes the surgeon's job easier by minimizing risks and increasing overall performance.

By using this technology, doctors can easily detect changes in critical parameters of the COVID-19 patient. This information-based service opens up new healthcare opportunities as it moves towards the best way for an information system to adopt world-class results and enables improving treatment systems in the hospital. Medical students can now be better trained for disease detection and well-guided for the future course of action. IoT's proper usage can help resolve medical challenges like speed, price, and complexity. It can easily be customized to monitor caloric intake and treatment like asthma, diabetes, and arthritis of the COVID-19 patient. This digitally controlled health management system can improve the overall performance of healthcare during the COVID-19 pandemic days.

V. FUTURE SCOPE

In the future, IoT will monitor the patient's vital signs in a real-time environment. This technology will digitally collect all detailed information to prevent ongoing issues regarding the treatment of the COVID-19 patient. There will be a significant enhancement in healthcare practice, using the latest technologies**,** and doctors will have to use them. IoT is a sophisticated developing technology with extensive applications in providing specialized medical care that opens up an effective way to analyze valuable data, information, and testing.

The future has applications in managing inventories used in the medical field and the medical supply chain for getting the correct item at the right time and location. IoT intelligent devices would be performing autonomously. There will be data storage with private and public clouds, and even software would also be on the cloud; thus, disease identification and follow-up could be made efficient. This disruptive innovation of the information system will facilitate intelligent healthcare service in the Medical 4.0 environment.

REFERENCES

[1] Rishabh Jain, "IoT-based patient health monitoring system using ESP8266 and Arduino," article to Circuit Digest, September 2018. https://circuitdigest.com/microcontroller-projects/iot-based-patient-monitoring-system-using-esp8266-and-arduino

[2] A. Thakallapelli, S. Ghosh and S. Kamalasadan, "Real-time frequency based reduced order modeling of the large power grid," Power and Energy Society General Meeting, 2016, Boston, MA, USA. https://ieeexplore.ieee.org/document/7741877

[3] Uddin, M. S., Alam, J. B., & Banu, S. (2017, September). Real-time patient monitoring system based on the internet of things. In 2017 4th International Conference on Advances in Electrical Engineering (ICAEE) (pp. 516-521). IEEE. DOI: 10.1109/ICAEE.2017.8255410

[4] Unknown, "Pulse Sensor Getting Started Guide," unpublished. https://www.generationrobots.com/media/DetecteurDePoulsAmplifie/PulseSensorAmpedGettingStartedGuide.pdf

[5] Lisa Jane Lishman, "Remote patient monitoring: the future of healthcare?," March 2021, in press. http://www.pharmatimes.com/web_exclusives/Remote_patient_monitoring_the_future_of_healthcare_1365643.