Personal Safety Device (PSD)Using Real-Time GPS & Environmental Communication System

Adnan Sarker Rafi
Department of Electrical and Electronics Engineering,
Ahsanullah University of Science and Technology,
Dhaka, Bangladesh
adnan.eee.20210105054@aust.edu

Miraz Hossain
Department of Electrical and Electronics Engineering,
Ahsanullah University of Science and Technology,
Dhaka, Bangladesh.
miraz.eee.20210105056@aust.edu

Topu Sarker Saiket
Department of Electrical and Electronics Engineering,
Ahsanullah University of Science and Technology,
Dhaka, Bangladesh.
topu.eee.20210105059@aust.edu

Fardin Hossoin
Department of Electrical and Electronics Engineering,
Ahsanullah University of Science and Technology,
Dhaka, Bangladesh
fardin.eee.20210105060@aust.edu

Jubayer Ahmed Fiham
Department of Electrical and Electronics Engineering,
Ahsanullah University of Science and Technology,
Dhaka, Bangladesh.
jubayer.eee.20200105040@aust.edu

Abstract—**The Personal Safety Device (PSD) is an innovative solution engineered to protect individuals in real-time. By integrating real-time GPS tracking, environmental monitoring, and efficient communication, the PSD empowers users to proactively mitigate potential threats and seek immediate assistance. The device utilizes the Sim808 module to ensure reliable cellular connectivity, facilitating the transmission of critical data, including GPS coordinates, environmental parameters, and distress alerts. Additionally, the incorporation of a DHT11 sensor enables precise measurement of ambient temperature and humidity, providing valuable environmental insights. The PSD's primary objective is to equip individuals with the necessary tools to navigate hazardous situations and obtain timely assistance. By combining precise location tracking with environmental awareness, the PSD delivers a comprehensive safety solution that can significantly reduce the risk of harm. This paper will explore the design, implementation, and evaluation of the PSD, underscoring its potential to revolutionize personal safety measures and enhance societal security.**

I. INTRODUCTION

The escalating rates of personal safety incidents necessitate the development of innovative solutions to safeguard individuals in real-time. This paper presents a novel Personal Safety Device (PSD) that leverages the power of real-time GPS tracking and environmental monitoring to provide comprehensive protection. By integrating advanced technologies, the PSD aims to empower individuals with the ability to track their location, monitor environmental conditions, and communicate distress signals efficiently.

At the core of the PSD lies a robust communication system that utilizes the Sim808 module to establish reliable cellular connectivity. This enables real-time data transmission, including GPS coordinates, environmental parameters, and distress alerts. To enhance the device's functionality, the DHT11 sensor is employed to monitor ambient temperature and humidity. This information is crucial for assessing environmental conditions and identifying potential hazards that could compromise safety.
Arduino Uno, a popular microcontroller platform, is ideal for various applications due to its comprehensive capabilities, user-friendly interface, and compact design. To initiate the system, specific commands are required to ensure proper boot-up and signal reception. While Arduino seamlessly integrates with a single sensor, challenges may arise when multiple sensors are connected. To address this issue, this paper explores the use of specialized library functions that optimize the handling of multiple sensors connected to Arduino Uno, ensuring accurate and reliable performance. The PSD's primary objective is to provide a comprehensive safety solution by combining accurate location tracking with environmental awareness. By equipping individuals with this information, they can make informed decisions, seek assistance promptly, and potentially prevent dangerous situations. This paper will delve into the design, implementation, and evaluation of the PSD, highlighting its potential to revolutionize personal safety measures.

II. LITERATURE SURVEY

The escalating prevalence of personal safety incidents has underscored the urgent need for innovative solutions to safeguard individuals in real-time. Personal safety devices, equipped with advanced technologies, have emerged as a pivotal response to this pressing concern. These devices, incorporating GPS tracking, environmental monitoring, and communication capabilities, offer comprehensive protection and facilitate timely assistance.

GPS technology has revolutionized location-based services, providing precise tracking and navigation capabilities. Numerous studies have explored the integration of GPS in personal safety devices, demonstrating its potential to deliver accurate location data, expedite emergency response, and enhance situational awareness. Researchers have delved into the synergy between GPS and various communication technologies to ensure reliable and timely transmission of location information.

Environmental factors can significantly influence personal safety. Studies have underscored the importance of monitoring environmental conditions, such as temperature and humidity, to identify potential hazards and mitigate risks. Sensors capable of measuring these parameters have been integrated into personal safety devices to provide real-time data and enable proactive measures to safeguard individuals.

Effective communication is paramount for personal safety devices to deliver timely alerts and facilitate assistance. Researchers have investigated various communication technologies, including cellular networks, Wi-Fi, and Bluetooth, to ensure reliable and efficient data transmission. Studies have explored the integration of these technologies with personal safety devices to enable seamless communication with emergency services and designated contacts.

While personal safety devices have demonstrated significant promise, several challenges persist. Addressing battery life, privacy concerns, and accuracy remains an ongoing area of research. Future studies aim to explore innovative approaches to enhance the functionality and effectiveness of personal safety devices, incorporating emerging technologies and addressing evolving safety needs.

III. PROPOSED SYSTEM

The framework integrates a comprehensive suite of components, including temperature sensor, LM 7805 voltage regulator, Mini 360 buck converter, Arduino Uno microcontroller, LCD display, batteries and a GSM modem.

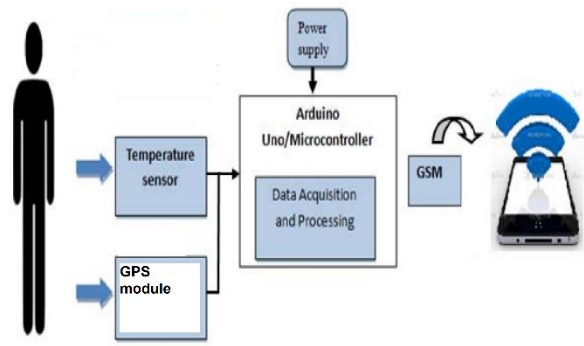

Fig. 1: Block diagram of proposed system

The system architecture, as depicted in Figure 1, comprises the components detailed in the preceding subsections. These components collectively constitute the smart board system, designed to monitor user location and surrounding environmental parameters. Sensors are employed to acquire raw data pertaining to user location and environmental factors, which are subsequently transmitted to the backend application via GSM technology.

A. Arduino UNO:

Fig.2: Arduino UNO

The microcontroller, serving as the central processing unit of the system design, executes the core logic. Popular options include Arduino, Beagle-Bone, and PIC. Among these, Arduino Uno was selected for its comprehensive specifications and ease of use. As illustrated in Figure 2, the Arduino Uno board, powered by the ATmega32 microcontroller, features 14 digital pins (6 of which can be used for pulse-width modulation), 6 analog inputs, a DC power jack, a reset button, an ICSP header, a 16 MHz ceramic resonator, and a USB interface.

B. Sensors

DHT11 sensor

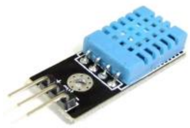

Fig.3: DHT11 sensor

The DHT11 sensor (Figure 3), a high-performance humidity and temperature sensor, serves as a critical component in the Personal Safety Device (PSD). By accurately measuring ambient environmental conditions, the DHT11 sensor provides valuable data for risk assessment and proactive measures. Its compact design and ease of integration facilitate efficient data acquisition and real-time environmental monitoring. The sensor was connected to the Arduino Uno microcontroller, adhering to the established protocol. The VCC pin of the DHT11 sensor was interfaced with the 5V pin on the Arduino Uno, while the GND pin was grounded. The data pin of the DHT11 sensor was connected to the D2 pin on the Arduino Uno, facilitating communication and data acquisition. This configuration enabled the reliable measurement of ambient temperature and humidity, providing essential environmental data for the PSD's safety features. The DHT11 sensor's contribution to the PSD is instrumental in ensuring comprehensive safety and situational awareness, empowering users to navigate potentially hazardous environments with confidence.

Sim808

Fig.4: Sim808

The Sim808 module (Figure 4), a versatile GSM/GPRS communication device, serves as a critical component in the Personal Safety Device (PSD). Its integration with the Arduino Uno is facilitated through a series of well-defined connections. The module's power supply is connected to the Arduino's 5V pin, while the ground pin is connected to Arduino's ground. The Sim808's TX pin is linked to the Arduino's RX pin, and vice versa, enabling bidirectional communication. Additionally, the module's RST pin is typically connected to a digital pin on the Arduino to facilitate software resets. This configuration enables the PSD to establish reliable cellular connectivity, facilitating the transmission of real-time GPS data, environmental parameters, and distress alerts.

C. LM 7805 voltage regulator

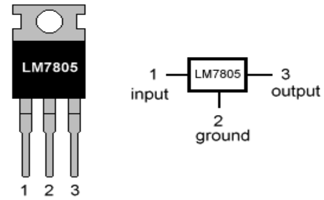

Fig.5: LM 7805

To provide a stable and regulated 5V power supply for the Arduino Uno, an LM7805 voltage regulator (Figure 5) is used in the system design. The regulator's input pin is connected to a higher voltage source, such as a 9V battery, while its output pin is connected to the Arduino Uno's 5V pin. A ground connection is made between the regulator's ground pin and the Arduino Uno's ground pin. This setup improves the system's performance and reliability by ensuring a consistent and dependable power source for the microcontroller and its peripherals.

D. Mini 360 buck converter

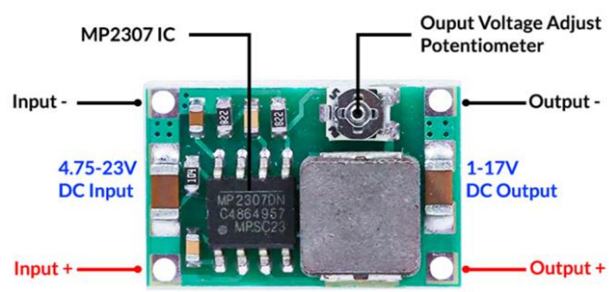

Fig.6: Mini360 buck converter

The Mini360 buck converter (Figure 6), a highly efficient DC-DC voltage regulator, plays a pivotal role in the Personal Safety Device (PSD). It is meticulously integrated into the circuit to convert the input voltage into a regulated output voltage, precisely meeting the power requirements of the Arduino Uno, DHT11 sensor, and Sim808 module. The circuit diagram, as shown in Figure 7, clearly delineates the interconnections among these components, ensuring reliable and efficient power distribution.

E. I2C LCD display 16x2

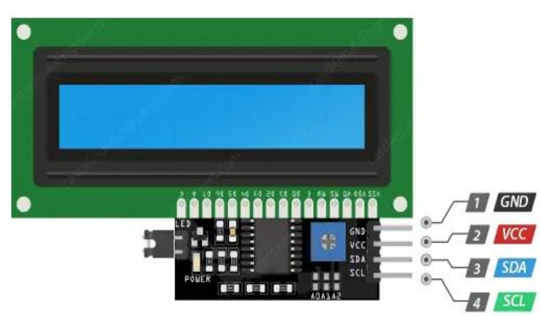

Fig.7: I2C LCD display 16x2

The I2C LCD display (Figure 7), a compact and efficient component, is seamlessly integrated into the system to provide real-time visual output. The display is connected to the Arduino Uno microcontroller via the I2C interface, utilizing the SDA and SCL pins. Employing the I2C protocol, the Arduino Uno communicates with the LCD display, transmitting data and instructions that govern the display's content. The LCD display, in turn, renders the received information on its 16x2 character grid, enabling the visualization of critical system parameters, such as GPS coordinates, environmental data, and status messages.

IV. Complete Hardware Setup

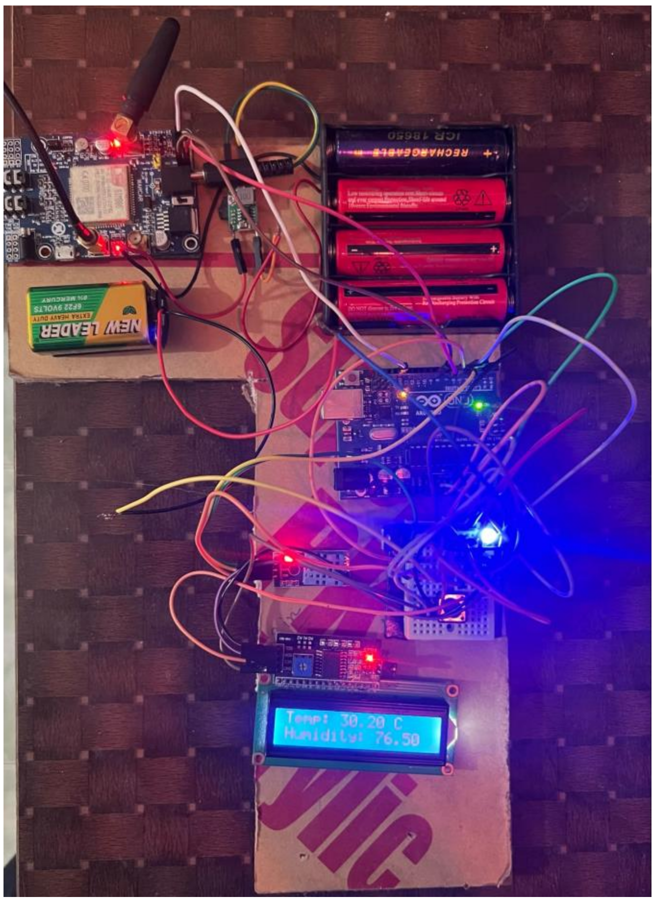

Fig.8: Complete hardware setup

Figure 8 illustrates the integration of the hardware components responsible for measuring heart rate, body temperature, and liquid level within the Personal Safety Device (PSD) system. These critical parameters provide valuable insights into the user's physiological state and environmental conditions, enabling the PSD to deliver comprehensive safety monitoring. The seamless integration of these components ensures accurate data acquisition and timely transmission for effective decision-making and emergency response.

V. Code Overview

The code leverages a collection of libraries to achieve its desired functionalities. The DFRobot_SIM808 library provides an interface for interacting with the SIM808 cellular module, enabling the retrieval of signal strength (getSignalStrength()) and the transmission of SMS messages (sendSMS()). Serial communication between the Arduino and the cellular module is established through the SoftwareSerial library. The DHT library facilitates the acquisition of temperature and humidity data from the DHT11 sensor (readTemperature(), readHumidity()). For controlling the Liquid Crystal Display (LCD), the LiquidCrystal_I2C library is employed. Furthermore, the code interacts with the GSM and GPS status LEDs using digitalWrite() function calls. This structured approach promotes modularity and code reusability.

The implemented code in the system is as follows:

```
#include <DFRobot_SIM808.h>
#include <SoftwareSerial.h>
#include <DHT.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

#define PIN_TX 7
#define PIN_RX 8
#define BUTTON_PIN 2
#define GSMLED 3
#define wl 4
#define GPSLED 6

#define DHTPIN 5
#define DHTTYPE DHT11
#define PHONE_NUMBER "+8801744132580"


SoftwareSerial mySerial(PIN_TX, PIN_RX);
DFRobot_SIM808 sim808(&mySerial);
DHT dht(DHTPIN, DHTTYPE);
LiquidCrystal_I2C lcd(0x27, 16, 2);

void setup()
{
  pinMode(BUTTON_PIN, INPUT_PULLUP);
  pinMode(GSMLED, OUTPUT);
  pinMode(wl, OUTPUT);
  pinMode(GPSLED, OUTPUT);

  mySerial.begin(9600);
  Serial.begin(9600);
  dht.begin();
  lcd.begin(16, 2);
  lcd.backlight();
```

```
  while(!sim808.init())
  {
    delay(1000);
    Serial.print("Sim808 init error\r\n");
  }
  Serial.println("Sim808 init success");
  Serial.println("Device ready");


  mySerial.println("AT+IPR=9600");
  delay(1000);
  while (mySerial.available())
  {
    Serial.write(mySerial.read());
  }


  mySerial.println("AT+CGPSPWR=1");
  delay(1000);
  while (mySerial.available())
  {
    Serial.write(mySerial.read());
  }
  mySerial.println("AT+CGPSRST=0");
  delay(1000);
  while (mySerial.available())
  {
    Serial.write(mySerial.read());
  }
}

bool getSignalStrength()
{
  int retryCount = 3;
  int signalStrength = 0;

  for (int i = 0; i < retryCount; i++)
  {
    if
(sim808.getSignalStrength(&signalStrength))
    {
      if (signalStrength > 0)
      {
        Serial.println("GSM Signal Good");
        return true;
      } else
      {
        Serial.println("GSM Signal Bad");
        return false;
      }
    }
    else
    {
      Serial.println("Attempt failed to get
signal strength");
      delay(500);
    }
  }

  Serial.println("Failed to get signal
strength after retries");
  return false;
}

String getGPSData()
{
  mySerial.println("AT+CGPSINF=0");
  delay(1000);

  String gpsData = "";
  while (mySerial.available())
  {
    char c = mySerial.read();
    gpsData += c;
  }

  if (gpsData.length() > 0)
  {
    Serial.println("Raw GPS Data:");
    Serial.println(gpsData);


    if (gpsData.indexOf("$GPGGA") != -1)
    {
      int comma1 = gpsData.indexOf(",");
      int   comma2   =   gpsData.indexOf(",",
comma1 + 1);
      int   comma3   =   gpsData.indexOf(",",
comma2 + 1);
      int   comma4   =   gpsData.indexOf(",",
comma3 + 1);
      int   comma5   =   gpsData.indexOf(",",
comma4 + 1);
      /*int  comma6   =  gpsData.indexOf(",",
comma5 + 1);
      int   comma7   =   gpsData.indexOf(",",
comma6 + 1);
      int   comma8   =   gpsData.indexOf(",",
comma7 + 1);
      int   comma9   =   gpsData.indexOf(",",
comma8 + 1);
      int   comma10   =  gpsData.indexOf(",",
comma9 + 1);*/


      String            latDeg             =
gpsData.substring(comma2 + 1, comma2 + 3);
      String            latMin             =
gpsData.substring(comma2 + 3, comma3);
      float   latitude   =  latDeg.toInt()   +
latMin.toFloat() / 60.0;
```

```
      String lonDeg = gpsData.substring(comma4 + 1, comma4 + 4);
      String lonMin = gpsData.substring(comma4 + 4, comma5);
      float longitude = lonDeg.toInt() + lonMin.toFloat() / 60.0;


      Serial.print("Latitude: ");
      Serial.println(latitude, 6);
      Serial.print("Longitude: ");
      Serial.println(longitude, 6);


      String googleMapsUrl = "https://maps.google.com/maps?q=" + String(latitude, 6) + "," + String(longitude, 6);
      Serial.println("Google Maps URL:");
      Serial.println(googleMapsUrl);

      return googleMapsUrl;
    }
  }
  return "";
}

void loop()
{
  static bool previousSignalStatus = false;
  bool currentSignalStatus = getSignalStrength();


  if (currentSignalStatus != previousSignalStatus)
  {
    digitalWrite(GSMLED, currentSignalStatus ? HIGH : LOW);
    previousSignalStatus = currentSignalStatus;
  }

  float h = dht.readHumidity();
  float t = dht.readTemperature();


  lcd.setCursor(0, 0);
  if (currentSignalStatus)
  {
    lcd.print("Temp: ");
    lcd.print(t);
    lcd.print(" C");
    lcd.setCursor(0, 1);
    lcd.print("Humidity: ");
    lcd.print(h);
    lcd.print(" %");
  }
  else
  {
    lcd.print("Signal Lost  ");
    lcd.setCursor(0, 1);
    lcd.print("              ");
  }

  if (digitalRead(BUTTON_PIN) == LOW)
  {
    delay(50);
    if (digitalRead(BUTTON_PIN) == LOW)
    {
      Serial.println("Button pressed");
      lcd.clear();
      lcd.print("Button Pressed");

      digitalWrite(wl, HIGH);

      String message = "";
      String googleMapsUrl = getGPSData();

      if (googleMapsUrl != "")
      {
        digitalWrite(GPSLED, HIGH);
        message += googleMapsUrl + " ";
      }
      else
      {
        digitalWrite(GPSLED, LOW);
        message += "Out of range. ";
      }

      message += "Temp: " + String(t) + "C, Humidity: " + String(h) + "%";
      sim808.sendSMS(PHONE_NUMBER, message.c_str());
      Serial.println("SMS sent");


      lcd.clear();
      lcd.print("Calling...");
      if (sim808.callUp(PHONE_NUMBER))
      {
        Serial.println("Calling");
        delay(30000);
        sim808.hangup();
        Serial.println("Call ended");
      }

      digitalWrite(wl, LOW);
      lcd.clear();
      while (digitalRead(BUTTON_PIN) == LOW)
      {
        delay(10);
```

```
      }
    }
  }

  delay(1000);
}
```

VI. Result

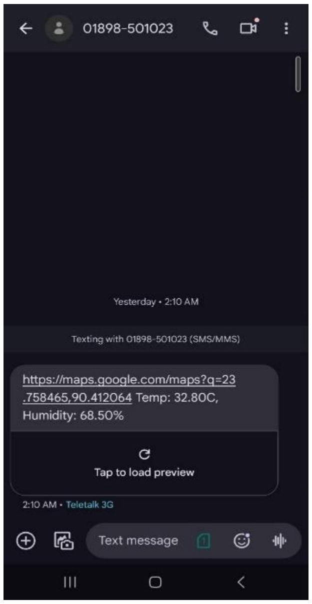

Fig.9: SMS received by our hardware system

The system effectively measures the specified parameters and transmits the collected data to designated recipients via SMS messages. Figure 9 illustrates the standardized format of the SMS alerts, ensuring clear and concise communication of the measured values. This timely data transmission empowers individuals to monitor their well-being and seek assistance when necessary, contributing to their overall safety and security.

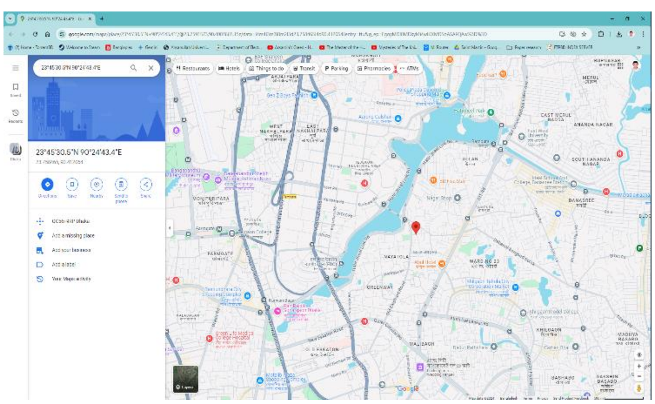

Fig.10: Web view showing the position using the red marker

The device's comprehensive suite of sensors and modules, including the Arduino Uno, DHT11, Sim808, and I2C LCD display, ensures reliable and efficient operation. Future enhancements could involve expanding sensor capabilities, exploring alternative communication technologies, and integrating artificial intelligence for more sophisticated data analysis and decision-making.

Conclusion

The Personal Safety Device (PSD) presented in this paper offers a comprehensive solution for personal safety, integrating real-time GPS tracking, environmental monitoring, and communication capabilities. The device effectively monitors user location, measures vital parameters, and transmits critical information via SMS alerts.

The PSD's integration of various sensors and communication modules, including the Arduino Uno, DHT11, Sim808, and I2C LCD display, ensures reliable and effective operation. Future enhancements could involve expanding sensor capabilities and exploring alternative communication technologies to further enhance the PSD's functionality.

Future enhancements could involve expanding the sensor suite to incorporate additional parameters, exploring alternative communication technologies for enhanced connectivity, and investigating methods to optimize battery life and device durability.

In conclusion, the PSD demonstrates a promising approach to personal safety, offering a valuable tool for individuals seeking to protect themselves and their loved ones.

Reference

[1] Balingasa, R. H., Bilog, M. T. C. R., Castillo, J. K. D., Perez, J. M., Terrible, A. F., & Caldo, R. B. (2015). Distress signal tracker using GPS and SMS technology: A prototype. LPU-Laguna Journal of Engineering and Computer Studies, 3(1).

[2] Kanani, P., & Padole, M. (2020). Real-time location tracker for critical health patient using Arduino, GPS Neo6m and GSM Sim800L in health care. In Proceedings of the International Conference on Intelligent Computing and Control Systems (ICICCS 2020) (pp. 1-6). IEEE Xplore Part Number: CFP20K74-ART. ISBN: 978-1-7281-4876-2.

[3] Digarse, P. W., & Patil, S. L. (2017). Arduino UNO and GSM based wireless health monitoring system for patients. In Proceedings of the International Conference on Intelligent Computing and Control Systems (ICICCS 2017)

[4] R. G. Landaeta, O. Casas, and R.P. Areny, "Heart rate detection from plantar bioimpedance measurements", 28th IEEE EMBS Annual International Conference, USA, 2006, pp. 5113- 5116.

[5] Ahmed, Salman, et al. "Wireless health monitoring system for patients." 2015 IEEE International WIE

Conference on Electrical and Computer Engineering (WIECON-ECE). IEEE, 2015.

[6] A compact Arduino NMEA (GPS) parsing library. (2019, Feb, 25). IoT TinyGPS. [Online]. Available: https://github.com/mikalhart/TinyGPS

[7] https://circuitdigest.com/microcontroller-projects/arduino-based-accident-alert-system-using-gps-gsm-accelerometer

[8] https://www.youtube.com/watch?v=470cmY1CCD8