IJCSN - International Journal of Computer Science and Network, Volume 8, Issue 1, February 2019
ISSN (Online): 2277-5420
www.IJCSN.org
Impact Factor: 1.5

Automated Low-Cost Home Sensing with E-mail Capability: A new Perspective to Remote Home Sensing for Surveillance

[1] **John Francis Ogbonoko,** [2]**Abdulkadir Dauda,** [3]**Haruna Umar Adoga**

[1]Department of Computer Science, Federal University Lafia,
Nasarawa, Nigeria

[2]Department of Computer Science, Federal University Lafia,
Nasarawa, Nigeria

[3]Department of Computer Science, Federal University Lafia,
Nasarawa, Nigeria

Abstract - This research focuses on the application of Internet of Things in remote home sensing with the design and implementation of a system built around WiFi infrastructure. It also focused on the strength of the microcontroller - Raspberry Pi in building IoT systems. The developed system is made up of two parts; the hardware part that is powered by Raspberry Pi. This provides the appropriate interface for connecting the sensors, pi-camera module, and other hardware devices of the system. The second part which is more of the programming aspect implemented using Python is the distance measurement, image capture, and email application that provides the interface for home surveillance and monitoring against burglary, and for appropriate actions to be taken by the home owner. The system is scalable, adaptable, and cost effective compared to the commercially available systems found in the market for the same purpose. It is also easy to build by anyone if the implementation procedures outlined in this work are followed through correctly.

Keywords – *Internet of Things, Home sensing, Raspberry Pi, Python, Sensor Networks, Computer Networks*

1. Introduction

The internet has grown rapidly from its conventional use for access to global information and communication to one that makes devices smart (intelligent), and also enables connection between these devices and with human beings to create a smart environment such as in homes, health care, agriculture, education, cities etc. This growth in information and communication technologies (ICT) and in the domain of embedded systems has given rise to a new field of technology referred to as the Internet of Things (IoT) [1, 2]. IoT is a major driver of the fourth industrial revolution, and it is premised on existing and well established ICT standards.

The overwhelming benefits of IoT is that it extends internet to virtually every object for example, temperature and power meters, refrigerators, microwave ovens, washing machines, machines, vehicles, and equipment involved in industrial processes [3]. This enables such objects to establish connection and communicate between themselves and with people [4] using embedded Wireless

Network Sensors (WNS) and Radio-Frequency Identification (RFID) [5, 3, and 6]; RFIDs allow short-range communication using RFID tags and RFID readers. Home sensing is an aspect of the Internet of Things (IoT) that is gaining wide acceptability as evidenced by inventions leading to products and services that are deployed to promote comfort and wellbeing in homes. For example, monitoring of the sick or the elderly people in homes to ascertain their location at a particular time within the home, monitor their body temperature, and blood pressure. Another important benefit of using IoT systems for home sensing is that it allows for remote monitoring of equipment, persons, pets, and surrounding environment, which reduces onsite human involvement.

The safety of the home environment is very crucial in the protection of lives and properties. This is why individuals, private and public organizations spend a huge amount of money to provide security for their buildings. The cost of these standard devices is usually very high and limiting affordability only to the wealthy. But with the IoT concept, similar devices with equivalence of throughput as the standard devices makes it possible for everyone to have low-cost or affordable home sensing IoT systems deployed for use. Some existing IoT products or projects define home sensing in terms of home surveillance, health and wellbeing monitoring, home automation etc. which are mostly driven by environmental parameters.

This work majorly explored the strengths of the Raspberry Pi in building a low-cost home sensing device, and to provide an understanding of the concept of developing IoT systems from a computational point of view. It was also necessary to bear in mind the need to be explicit in describing the design and implementation procedures for those who may be interested in equipping themselves with requisite IoT skills for self-empowerment.

2. Literature Review

As the interconnectivity between billions of devices increases, new products and services continue to emerge serving various purposes in our everyday life. This creativity brought by IoT forms the basis for home automation in general and sensing of homes in particular, thereby providing home owners with useful information about their home environment even when at remote locations. This helps them to keep track of activities in the home as most devices are tagged with RFIDs and sensors that transmit information to the home owners. Underpinning most of these home sensing and automation systems is the Raspberry Pi. In this section, the authors looked at some case study systems that have been developed around Raspberry pi.

Observing the limitations of some existing home surveillance systems such as CCTV, which requires the home owner at the location under surveillance to have a clue about what is happening within the environment, [7] designed and implemented a surveillance monitoring system using Raspberry pi and PIR sensor. This system provided the home owner with real-time information at remote locations on an Android smart phone over internet connectivity. Though, the author did not state how the system was tested and what programming language was used.

In some cases, these applications combine the functionalities of home monitoring and automation over a web interface using parameters generated from temperature sensing and voltage measuring devices as implemented in [8]. The researcher designed a model that could feed users with data on a mobile device from within the home. The objective was to reduce the level of human interaction in controlling home appliances such as a fan. The fan was controlled using readings acquired from a temperature sensor. This model worked well in achieving the objectives of the research. An advancement of this work was carried out in a wireless-based system proposed by [9] and [10]. The proposed system would have capabilities for data collection bothering on the Activities of Daily Life (ADL) of the residents. For [9], the system model was trained on the activities of the residents and to provide feedback on the state of their health or activities. Such feedback could be used as a basis for immediate intervention in the event of a health emergency like heart attack or a fall. The system recorded 98% accuracy; however, there were observed limitations such as cold start-ups and duration in training the model on the resident’s activities. A solution to the cold start-up problem would be to use a more effective machine learning technique and providing more data to enhance the system’s performance, while the second problem could be addressed by using a more effective algorithm that increases the system’s precision on data. [10] Used sensors for data acquisition and GPS for location tracking within a building. The system worked without any human intervention and provided users with information on their daily routines.

[11, 12] used a voice recognition system and Raspberry pi as a sensor web node for home automation. The work in [12] used voice recognition to control appliances using Siri enabled mobile device and Raspberry Pi. The system recorded 93.3333% accuracy which was measured in terms of, gender voice recognition, command recognition, and the time taken for the system to respond to a command. Observed gap was in testing the system to recognize commands using foreign accents. [11] Focused on fire prevention using fire detecting sensors to detect any anomalies that could lead to fire outbreak in a home. The use of an analog temperature sensor required a conversion of analog data to digital signal; the researcher however, did not mention how this was achieved. A good example would be to mention the use of MCP3002 or MCP3008 Analog-to-Digital Converter.

3. Analysis of the System

3.1 Functional Requirements of the System

As mentioned earlier, the design of the home surveillance system is to suit the purpose of monitoring the home

IJCSN - International Journal of Computer Science and Network, Volume 8, Issue 1, February 2019
ISSN (Online): 2277-5420
www.IJCSN.org
Impact Factor: 1.5

environment against burglars. Such systems help to keep the environment safe by providing home users and security agencies with information on the activities of burglars. The design consideration is simple and easy to develop within a learning environment.

The system has the following functionalities:

- Detect human motion from a distance
- Specify the distance of the detected motion from the source
- Capture an image when human motion is detected
- Save all captured images in folder located on the system
- Send email notification with an image attachment
- Connect to a Wi-Fi network

3.2 Hardware Requirements of the System

Various hardware solutions are available in the market for developing personal IoT systems so, and can be readily purchased from a nearby outdoor electronic or an online electronic shop. The hardware materials used in this research are cheap and easy to play with. In this section we discussed some of the hardware materials used in terms of their unique characteristics and advantages.

Raspberry Pi Model B: Raspberry Pi is an open source credit-card size, single-board programmable computer introduced by the Raspberry Foundation in 2012 [11] to encourage low-level programming skills [13], and for developing cost-effective systems especially in the domain of home sensing and automation [11, 12].

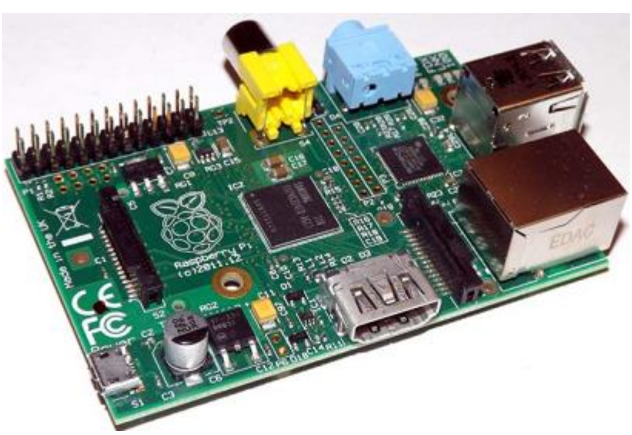

Fig.1 Raspberry Pi Model B [13].

The Raspberry Foundation recommends the Rasbian Pi operating system for use on Raspberry Pi; however, it supports other operating systems such as Linux, Plan 9, RISC OS [14], and Microsoft Windows 10. It has some interesting features that make it useful for a wide range of applications and for interfacing with other hardware peripherals. Some of the features of Raspberry Pi Model B are listed below [15, 16].

- 2 USB ports
- 26 General Purpose Input/Output (GPIO) pins
- HDMI port
- Ethernet port
- 3.5mm audio jack for audio out
- Camera interface (CSI)
- 512MB RAM
- Micro SD card slot
- Composite video
- Camera Serial Interface
- Display Serial Interface

Here are some of the advantages of Raspberry Pi Model B:

- It is a programmable computer. That is, it is in itself a computer
- It has networking capabilities as it comes with a built-in Ethernet port
- No knowledge of electronics is required
- Suitable for intensive projects where networking is required
- A good knowledge of Linux is required

HC-SR04 Ultrasonic Ranging Module Sensor: The HC-SR04 ultrasonic motion sensor uses ultrasonic waves to measure the distance from the source (sensor) to a target object. One of its advantages is the accuracy and stability it offers in measuring distances using ultrasonic waves. It is capable of measuring distances between 2cm – 400cm. It has both a transmitter and a receiver, and between them, a burst is transmitted back and forth [17].

Fig. 2 HC-SR04 Ultrasonic Ranging Module Sensor [18].

The distance between the transmitter and the receiver can be calculated thus, Distance, d = (speed x time)/2. Where, speed is the speed of sound at sea level.

Fig. 3 Ultrasonic Module in Operation [17].

It has four pins V_{CC} for 5V power input, Trig pin for Trigger input, ECHO pin for Echo output, and GND pin for Ground. The electronic parameter obtained from [18] is shown in the table below.

Table 1: Electronic Parameter for HC-SR04 Ultrasonic Ranging Module Sensor

S/No	Parameter	Description
1	Working voltage	DC 5-Volt
2	Working current	15mA
3	Working frequency	40Hz
4	Maximum distance coverage	4 meters
5	Minimum distance coverage	2cm
6	Measuring angle	15 degree
7	Trigger input signal	10uS TTL pulse
8	Echo output signal	Input TTL lever signal and the range in proportion
9	Dimension	45*20*15mm

Raspberry Pi Camera Module: The module is used for capturing videos and still images when its 15-pin Ribbon Cable of the module is plugged onto the 15-pin MIPI Camera Serial Interface (CSI) of the Pi. It has the following [19]:

- 2592 x 1944 pixels for still picture resolution characteristics
- 20 x 25 x 9mm size
- 5MP Omnivision 5647 camera module
- 15-pin ribbon cable
- It is compatible with all versions of Raspberry Pi
- Weight of 3g

Fig. 4 Raspberry Pi Camera Module [19].

3.3 Software and Programming Language Consideration

Based on the capabilities and features of Raspberry Pi discussed above, it can be explored using several programming languages; however, Python is the default programming language that comes with the Rasbian Linux distribution providing a text editor "IDLE" - this can be used for writing and running Python scripts. Python is easy to learn and provides a rich library that makes it fun for beginners. This rather interesting programming language was used for the programming aspect of the implementation.

4. Description of Home Surveillance Modules

The various modules that provide a complete design of the system and how they function are discussed below.

4.1 Motion Sensing

The module is responsible for sensing objects and measuring their distances from the source. In this case, it sensed human body motion. In the activated mode, the value of the PIR sensor ECHO pin changes from zero to one in the event that motion is detected within the angle of the sensor's detecting zone. The distance of the object from the source is then displayed.

4.2 Image Capturing

The image capturing module is responsible for motion capturing when human motion is detected. How this works is that if body motion is sensed within a specified distance from the sensor, the capture method of the pi-camera library in the Python script is called and an image is captured.

4.3 Email Notification

The image captured in the preceding section is stored in a folder on the Raspberry Pi. To enable the home owner access to captured images, the email notification module is activated and an email is sent to the home owner's email account with an email attachment. This will allow for the identification of the intruder or burglar.

5. Tools and Materials

Raspberry Pi Model B, jumper wires, 300R and 470R resistors, HC-SR04 ultrasonic motion sensor, Pi camera module, Wi-Fi dongle, USB power cable, and a solderless breadboard.

6. System Design and Implementation

6.1 Hardware Design Concepts

The hardware configuration setup is shown in Fig. 6. Four jumper wires were plugged onto the HC-SR04 ultrasonic motion sensor as follows: red - Vcc pin, blue - Trig pin, yellow - Echo pin, and black - GND pin. Two wires were used to connect the 5V pin (pin 2) and GND pin (pin 6) of the Raspberry Pi to the positive and negative rails of the breadboard respectively. The 300 Ohm resistor was

IJCSN - International Journal of Computer Science and Network, Volume 8, Issue 1, February 2019
ISSN (Online): 2277-5420
www.IJCSN.org
Impact Factor: 1.5

connected between the Echo pin of the HC-SR04 and the GND pin of the Raspberry Pi while, the 470 Ohm resistor was connected between the negative rail of the board and the GND pin of the Pi. The pi-camera ribbon was plugged firmly unto the connector which is located between the HDMI and Ethernet ports. When the setup was completed, it was connected to a power source using the USB adapter.

It is important to note that the Echo pin of the HC-SR04 sensor has a rating of 5V. There is need for this value to be reduced to avoid damage to the circuit by using a voltage divider. The divider reduces the output voltage (V_{OUT}) to 3.3V from the input voltage (V_{IN}) of 5V of the ECHO pin. To achieve this, two resistors were connected in series.
Eq. (1) below are useful for this purpose [1],

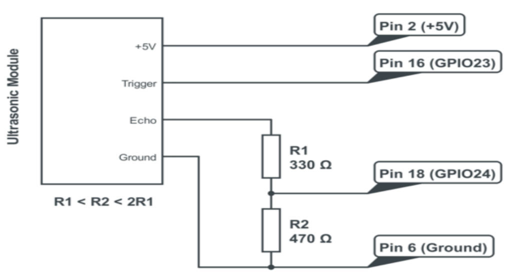

Fig. 5 Ultrasonic Module Circuit [20].

$$V_{OUT} = V_{IN} \text{ x } R_2/(R_1 + R_2)$$

$$V_{OUT}/V_{IN} = R_2/(R_1 + R_2) \qquad (1)$$

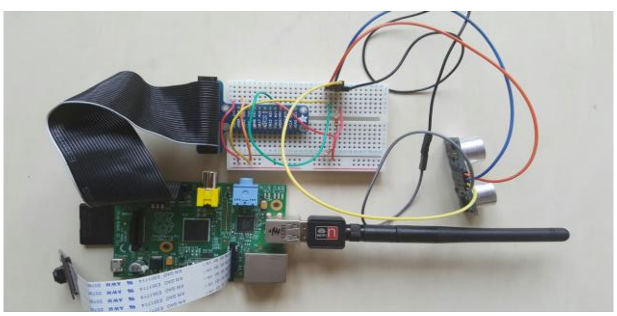

Fig. 6 Hardware Setup for Raspberry Pi Home Surveillance System.

6.2 Software Design Concepts

Motion Sensing and Image Capturing: The sensor is able to detect objects within the radius of coverage of the ultrasonic wave. It works in such a way that it measures the distance of the object from the source. For this work, the system was required to measure the distance between the human motion and the sensor (source). Two variables were declared; **'stopTime'** and **'startTime'** to measure the time interval between the start time and the stop time; that is, when the ECHO pin changes from LOW to HIGH. The two variables are declared as follows;

startTime = time.time()
stopTime = time.time()

The difference in the time interval is multiplied with the sonic speed (speed of sound).

```
#If the distance travelled is less than 30cm, then the Pi-Camera module should
#be activated and capture an image of the burglar
if distance < 30:
    print("Hi")
    currentTime = datetime.now()
    with picamera.PiCamera() as camera:
        camera.resolution = (1024, 768) #image resolution
        camera.start_preview()
        # Camera warm-up time
        time.sleep(2)
        camera.capture('image.jpg') #name of image is set
        print("We have taken a picture.")
```

Fig. 7 Section of Python Script Responsible for Capturing Images.

Another approach was to specify a directory path where the images would be saved. For example, specifying the image path as **picPath = "/home/pi/Desktop/images/"**. Also, specifying the image name using the current date and time of the Raspberry Pi. This gives each image a unique name. For instance, specifying the image name as **picName = currentTime.strftime("%Y.%m.%d-%H:%M:%S")+'.jpg'**. The variables "picName" and "picPath" were concatenated and passed as arguments to the capture method of the camera as **camera.capture(picPath + picName)**. This was to ensure that while the image was saved in the path specified, it was being assigned the current date and time as a unique name, thereby creating a kind of dynamism in the naming.

To capture images based on the distance measured, a condition was set for which image should be captured. In Fig. 7 above, the condition for image capture was given to be any distance less than 30cm.

IJCSN - International Journal of Computer Science and Network, Volume 8, Issue 1, February 2019
ISSN (Online): 2277-5420
www.IJCSN.org
Impact Factor: 1.5

Email Notification: this required creating a Gmail account to serve as the sender's email and installing two packages; Simple Mail Transfer Protocol (SSMTP) and mail utilities (mailutils). SSMTP is a lightweight server that allows for sending an email, while mailutils is a set of libraries for handling emails. To install the two packages, the command **"sudo apt-get install ssmtp mailutils"** was used in the LX Terminal of the Raspberry Pi. The following editing was carried out in the try block of the code snippet shown in Fig. 8;

- SMPT port was set to 587 as "server = smtplib.SMTP('smtp.gmail.com',587)"
- the receiver's email was set as toaddr = 'receiver@gmail.com' (any valid receiver's email may be used)
- the sender's email which was the one created earlier was set as,

  smtpUser = 'sender@gmail.com'
- Provide the password to the Gmail account created as smtpPass = '**************'

```
toaddr = 'johnogbonoko@gmail.com'    #Email address of receiver
smtpUser = 'piprojectraspberry@gmail.com' # Email address of sender
smtpPass = 'zxvhgwqtrroeslgz' #Sender's password
subject = 'Photo ' + f_time #Subject of email

msg = MIMEMultipart() #intermediate base class for MIME messages that are multipart
msg['Subject'] = subject
msg['From'] = smtpUser
msg['To'] = toaddr
msg.preamble = "Photo @ " + f_time

fp = open('image.jpg', 'rb') #Opens the file and reads the binary file
img = MIMEImage(fp.read())
fp.close() #Closes the file
msg.attach(img) #attaches the image to the message
print("Getting ready to send  an email")
try:
    #An smtp server connection is created here to enable sending an email
    server = smtplib.SMTP('smtp.gmail.com',587) #Setting the mailhub to port 587
    server.ehlo()
    server.starttls()
    server.ehlo()
    server.login(smtpUser, smtpPass)
    #s.send_message(msg)
    server.sendmail(smtpUser, toaddr, msg.as_string())
    server.quit()
```

Fig. 8 Code Snippet for Sending Email.

From the script above, the image file specified in the code snippet was passed to the open method as an argument. The second argument is the mode specifying how the file should be opened, which in this case is to read the binary file. The *sendmail()* performs the operation of sending an email. It has three arguments; the sender's email, the receiver's email, and the message.

7. Results

7.1 Distanace Measurement and Image Capture

The developed system was tested for sensing human motion using ultrasonic wave within the distance between the source (sensor) and the target object (human being). They system showed accuracy and stability in measuring the distance, hence, the varying distances as seen in Fig. 9 did not in any way affect the systems' ability in capturing the image of the object. It first measures the distance, and then captures the image. The measured distances and image capture notifications were displayed in the LX Terminal of the Raspberry Pi. As many times as distances are measured, images are captured. The images captured are stored in a folder on the Raspbian Operating System of the Raspberry Pi as shown in Fig. 10.

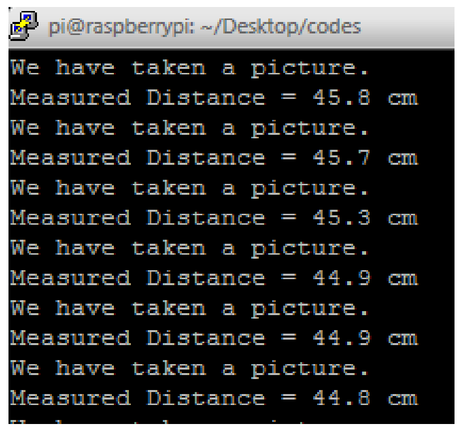

Fig. 9 Distance Measurement and Image Capture by Ultrasonic Motion Sensor.

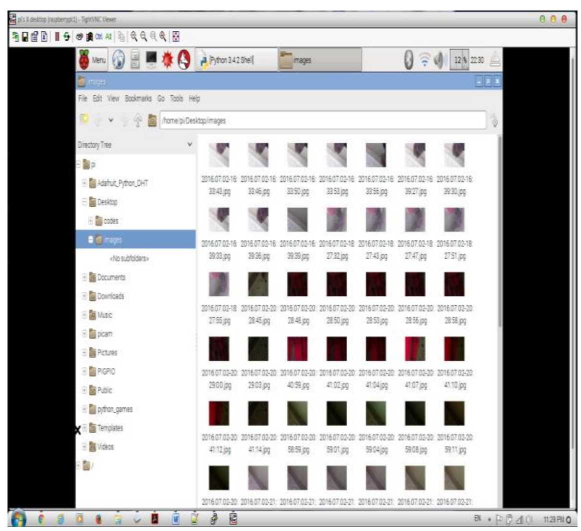

Fig. 10 Raspberry Pi Image Folder.

7.2 Email Notification

The receiver's email is that of the home owner. When an image is captured, it is stored in an image folder, and then sent as an email attachment. The home owner can view remotely the email attachment on their mobile devices. A sample of email notification is shown in Fig. 11.

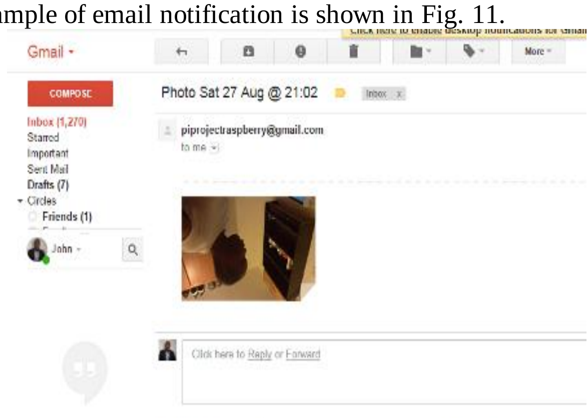

Fig. 11 Email Notification with Captured Image.

8. Conclusions

There is still much to be desired as far as home sensing for surveillance and monitoring is concerned. In the work carried out, the authors have successfully deployed Raspberry Pi and Python programming language to build a low-cost home sensing system. However, it is important to note that the system is scalable to accommodate a database that holds the distances measured, links to the images captured in the image folder, links to the email notification, and a web interface for controlling the system. It is possible to also extend functionality to a web interface that provides users with an option to take appropriate action, for example, to alert the nearest police station for arrest of burglars or intruders. Another consideration would be to have a module that scans images captured to be matched with images in a crime database for identification of burglars and intruders.

Internet of Things (IoT) have revolutionized the concept of home sensing and evolved in an age that is witnessing tremendous developments that blends sensor networks and computer networks. This blend played a major role in enabling a system that worked effectively to sense motion, capture images, and send an email notification in this work. With increasing reduction of cost and size of sensors and RFIDs, more objects can be made smart to generate data for human consumption and offer low-cost products and services, which was the focus of this work. It is all about promoting wellbeing by taking IoT to everyone in their homes.

References

[1] Challal, Y., Natalizio, E., Sen, S. and Vegni, A.M. (2015) 'Internet of Things Security and Privacy: Design Methods and Optimization', *Ad Hoc Networks*, 32, pp. 1–2. doi: 10.1016/j.adhoc.2015.05.010.

[2] Miorandi, D., Sicari, S., De Pellegrini, F. and Chlamtac, I. (2012) *Internet of Things: Vision, Applications and Research Challenges*, *Ad Hoc Networks*, 10(7), pp. 1497–1516. doi: 10.1016/j.adhoc.2012.02.016.

[3] Mashal, I., Alsaryrah, O., Chung, T.-Y., Yang, C.-Z., Kuo, W.-H. and Agrawal, D.P. (2015) 'Choices for Interaction with Things On Internet and Underlying Issues', *Ad Hoc Networks*, 28, pp. 68–90. doi: 10.1016/j.adhoc.2014.12.006.

[4] Bandyopadhyay, D. and Sen, J. (2011) 'Internet of Things: Applications and Challenges in Technology and Standardization', *Wireless Personal Communications*, 58(1), pp. 49–69. doi: 10.1007/s11277-011-0288-5.

[5] Gubbi, J., Buyya, R., Marusic, S. and Palaniswami, M. (2013) 'Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions', *Future Generation Computer Systems*, 29(7), pp. 1645–1660. doi: 10.1016/j.future.2013.01.010.

[6] Whitmore, A., Agarwal, A. and Da Xu, L. (2014) 'The Internet of Things—A Survey of Topics and Trends', *Information Systems Frontiers*, 17(2), pp. 261–274. doi: 10.1007/s10796-014-9489-2.

[7] PalSharma, D., Baldeo, A. and Phillip, C. (2015) 'Raspberry Pi Based Smart Home for Deployment in The Smart Grid', *International Journal of Computer Applications*, 119(4), pp. 6–10. doi: 10.5120/21053-3700.

[8] Addimulam, S. C. (2015) *Smart Home Control Using Raspberry Pi in Internet of Things Environment*. Available at: http://pqdtopen.proquest.com/doc/1738629611.html?FMT=ABS (Accessed: 20 June 2018)

[9] Fan, X., Huang, H., Qi, S., Luo, X., Zeng, J., Xie, Q. and Xie, C. (2015) 'Sensing Home: A Cost-Effective Design for Smart Home via Heterogeneous Wireless Networks', *Sensors*, 15(12), pp. 30270–30292. doi: 10.3390/s151229797.

[10] Mafrur, R., Khusumanegara, P., Bang, G.H., Lee, D.K., Nugraha, I.G.D. and Choi, D. (2015) 'Developing and Evaluating Mobile Sensing for Smart Home Control', *International Journal of Smart Home*, 9(3), pp. 215–230. doi: 10.14257/ijsh.2015.9.3.20.

[11] Vujović, V. and Maksimović, M. (2015) 'Raspberry Pi as a Sensor Web Node for Home Automation', *Computers & Electrical Engineering*, 44, pp. 153–171. doi: 10.1016/j.compeleceng.2015.01.019.

[12] Celebre, A.M.D., Dubouzet, A.Z.D., Medina, I.B.A., Surposa, A.N.M. and Gustilo, R.C. (2015) 'Home Automation Using Raspberry Pi Through Siri Enabled Mobile Devices', *2015 International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*. doi: 10.1109/hnicem.2015.7393270.

[13] Cox, T. (2014) 'Raspberry Pi Cookbook for Python Programmers' Birmingham: Packt Publishing.

[14] Wang, F., Hu, L., Zhou, J., Wu, Y., Hu, J. and Zhao, K. (2015) 'Software Toolkits: Practical Aspects of the Internet of Things—A survey', *International Journal of Distributed Sensor Networks*, 2015, pp. 1–9. doi: 10.1155/2015/534378.

[15] Srivastava, A. (2016) 'Raspberry pi 3 is out now! Specs, benchmarks & more' - *the MagPi magazine*. Available at: https://www.raspberrypi.org/magpi/raspberry-pi-3-specs-benchmarks/ (Accessed: 17 June 2018).

[16] *Model B Hardware General Specifications*. Available at: http://www.raspberry-projects.com/pi/pi-hardware/raspberry-pi-model-b/hardware-general-specifications (Accessed: 17 June 2018).

[17] Kartha, V., George, L., Shenoy, M. and George, E. (2015) 'Interfacing HC-SR04 Ultrasonic Sensor with Raspberry Pi'. Available at: https://electrosome.com/hc-sr04-ultrasonic-sensor-raspberry-pi (Accessed: 17 June 2018).

[18] *Ultrasonic Ranging Module HC - SR04*. Available at: http://www.micropik.com/PDF/HCSR04.pdf (Accessed: 29 June, 2018)

[19] *Raspberry Pi Camera Board (5MP, 1080p, v1.3)* (2015) Available at: https://www.modmypi.com/raspberry-pi/camera/raspberry-pi-camera-board-5mp-1080p-v1.3 (Accessed: 29 June, 2018).

[20] Hawkins, M. (2012) *Ultrasonic Distance Measurement Using Python – Part 1*. Available at: http://www.raspberrypi-spy.co.uk/2012/12/ultrasonic-distance-measurement-using-python-part-1/ (Accessed: 29 June, 2018).

[21] Ogbonoko, J. F. (2016). Creating a Laboratory and Computing Resources for Teaching Internet of Things: A Case of the Raspberry Pi Home. (M.Sc. Thesis, Department of Computer Science, University of Sheffield, United Kingdom), pp. 7, 10, 21-28, 38-39, 42-46.

First Author

Ogbonoko, J. F. holds a Bachelor of Science (BSc.) degree in Computer Science from the Benue State University, Makurdi, Nigeria. He also holds a Master of Science (MSc.) degree in Software Systems and Internet Technology from the University of Sheffield, United Kingdom. He currently lectures in the Department of Computer Science, Federal University Lafia, Nasarawa State, Nigeria. His research interests are in the area of Software Engineering, Internet of Things, and Big Data.

Second Author

Abdulkadir Dauda was born in Lafia, Nasarawa State of Nigeria on the 25th of October 1982. He obtained the Bachelor of Science degree in Computer Science from the Usmanu Danfodiyo University Sokoto, Nigeria in 2006. He worked with the Nigerian Judiciary as a Programme Analyst from February 2009 to April 2014 when he joined the Federal University Lafia as a Graduate Assistant. In 2015, he proceeded to the University of Bedfordshire, United Kingdom for his Masters of Science Degree which he completed in January 2017. He currently works as an Assistant Lecturer in the department of Computer Science, Federal University Lafia, Nigeria. His research interests are in the area of High-Performance Computing and Distributed Systems.

Third Author

Adoga, H. U. holds a Bachelor of Engineering (B.Eng.) in Electrical & Electronics Engineering from the University of Maiduguri, Nigeria, with specialization in data communications and networks. He also holds a Master of Science (MSc.) degree in Computer Science, from the University of Hertfordshire, England. He is currently a lecturer with the Department of Computer Science, Federal University Lafia, Nigeria. His research interests are in the areas of Software Defined Networking (SDN), IOT and Distributed Systems. He is a registered member of the Institute of Electrical and Electronics Engineers (IEEE), the Nigeria Computer Society (NCS), and the Nigeria Society of Engineers (NSE). As a CCNP professional, Haruna is also fascinated by design and configuration of computer networks.