Low-Cost Gas Leak Detection and Surveillance System for Single Family Homes Using Wit.ai, Raspberry Pi and Arduino

https://doi.org/10.3991/ijim.v16i09.30177

Ximena Pérez-Palomino(✉), Karina Rosas-Paredes, José Esquicha-Tejada
Universidad Catolica de Santa Maria, Arequipa, Peru
xperezp@ucsm.edu.pe

Abstract—The current situation in the region of Arequipa (Peru) is an increase in crime and insecurity; companies that provide private surveillance services have increased the costs of equipment and services online. We propose is to implement a low-cost gas leakage and surveillance system for single-family houses, implemented with Raspberry Pi3, an Arduino board, SIM 900 module, sensors, actuators, and peripherals. The system alerts by sending a text message when an intruder enters the home or when there is a gas leak; it captures the webcam image that is sent to the homeowner's email. For voice command recognition, Wit.ai and Firebase are used for communication between the system and the mobile application. System functionality and usability tests were carried out, allowing us to know user satisfaction.

Keywords—IoT, surveillance, Raspberry Pi, Arduino, SIM 900 module

1 Introduction

Currently, due to the COVID-19 health emergency, around 10,000 people are unemployed in the city of Arequipa, Peru [1]. This is a determining factor for insecurity in the city, according to the National Institute of Statistics and Informatics, In January and March 2021 alone, there has been an increase in the number of crime reports in Arequipa (5428 crime reports registered) [2]. Furthermore, it is considered that home security is not only about alerting about the presence of an intruder, but also about creating a safe environment, as well as other risks that can affect home life, such as fires [3].

Therefore, several projects or prototypes have been developed to solve this type of problem, which can be implemented with minicomputers, boards, sensors, and actuators that are cost-effective and easy to acquire, in order to provide greater safety for hospitalized patients [4], domotic solutions [5], improve education [6] and the automation of things (IoT) like watering the garden [7], [8].

According to the proposed scenario, there are solutions such as alarm systems but with high prices, systems that are difficult to use, with similar sounds, and do not detect gas leakage. Therefore, a gas leak detection and monitoring system is proposed using a Raspberry Pi 3 Model B minicomputer, an Arduino UNO board, a microphone,

a webcam, sensors, and actuators to provide home security that is low cost and low power consumption, as well as being easy to use and implement.

2 Current situation

In the region of Arequipa, Peru, there are several companies that offer alarm system services, at high prices and with similar characteristics. Initially, a survey was carried out to find out the security needs of the population, using equation (1) provided by [9].

$$n = \frac{\mathrm{N} * Z^2 * p * q}{d^2 * (N-1) + Z^2 * p * q} \tag{1}$$

The population size (N) of the region of Arequipa in 2017, is 1 382 730 and the percentage of people (0.46) who own a house is 46%, according to the National Institute of Statistics and Informatics [10]. Therefore, we obtain:

$$\mathbf{n} = (1\,382\,730 * 1.645^2 * 0.46 * 0.54)/(0.1^2 * (1\,382\,730 - 1) + 1.645^2 * 0.46 * 0.54)$$

$$\mathbf{n} = 67.2$$

A survey of 67 inhabitants living in the region Arequipa was then conducted to find out their perceived level of insecurity and how much they would be willing to pay for a gas leak detection and surveillance system. (See Table 1).

Table 1. Survey of level of insecurity and payment for surveillance system

Question	Percentage	Description
Have you ever had your property stolen by unauthorized income?	63%	Were victims of robbery
	37%	Were not victims of robbery
How much are you willing to pay for a gas leak detection and surveillance system?	55.8%	Pay from $220 to $280
	38.5%	Pay from $280 to $830
	5.7%	Pay from $830 to $1390

The questions asked and the result in percentage can be observed. The first question allows us to determine the level of insecurity, where 63% were victims of robbery, a high percentage, which represents the insecurity in the city. The second question allows to determine how much a person can pay for a surveillance and gas leak detection system, 55.8% indicated that they would pay between 220 to 280 dollars. This price will be the basis for developing the system and procuring the sensors, actuators, and peripherals.

3 Analysis of tools used

Various tools and alternatives were evaluated to develop and implement the gas leak detection and surveillance system with the best equipment and at a low price.

The main Raspberry Pi models were compared, without considering previous versions that are not available in the local market: Raspberry Pi 2 Model B, Raspberry Pi 3

Model B, and Raspberry Pi 4 Model B. In Table 2, when comparing the models, Raspberry Pi 3 Model B was chosen, due to its higher procedural speed, Bluetooth connection, and Wi-Fi when compared to the previous version, and it has a more accessible price and lower electricity consumption when compared to the recent version. These versions have 4 USB inputs to connect different equipment or peripherals.

The Raspberry Pi minicomputer requires an operating system, in this case, Raspbian was used for being simple, stable, fast, and with an extensive development community [4]. Raspberry Pi can be programmed with Python and Java, the chosen programming language is Python because there are more libraries, the greater contribution from the community (forums) and it allows the use of GPIO pins to connect the digital world with the physical world [11], [12].

Table 2. Raspberry Pi model types

Model	Raspberry Pi 2 B	Raspberry Pi 3 B	Raspberry Pi 4 B
Price	40 Dollars	40 Dollars	70 Dollars
SOC	Broadcom BCM2836	Broadcom BCM2837	Broadcom BCM2711Bo
CPU Clock	700 MHz	1.2 GHz	1.5 GHz
RAM	1 GB	1 GB	1 GB
USB	4	4	4
Wi-fi	No	Yes	Yes
Consumption	820 mA	1400 mA	2.5 mA

It is required to use an Arduino board, to connect the components that do not work at 3.3v in Raspberry Pi and to connect the GSM/GPRS SIM 900 module for sending text messages. Models were compared: Arduino Nano, Arduino Mega, Arduino Leonardo and Arduino Uno. Table 3 shows the comparison between the Arduino models, in this case the Arduino Uno model was chosen, since compared to the Arduino Nano the SIM900 GSM module/GPRS does not fit on a shorter board. Compared to the Nano, Mega and Leonardo models, the Arduino Uno board is more commercial, and a large number of libraries has only been developed for this model. Arduino Uno is economical, compatible with GSM/GPRS module and is regular size.

Table 3. Arduino model types

Model	Arduino Nano	Arduino Mega 2560	Arduino Leonardo	Arduino Uno
Microcontroller	ATmega 328P	AVR ATmega 2560 8 bits	AVR ATmega 32u4 8 bits	AVR ATmega 838 8 bits
Input/output digital pins	14/14	54/54	20/20	14/14
Analog input/ output pins	8/0	16/0	12/0	6/0
Price	6 dollars	42 dollars	16 dollars	11 dollars
Compatible GSM/ GPRS Module	NO	YES	YES	YES

For speech recognition, the speech recognition software that can be used on Raspberry Pi are Google Cloud Speech API, Pocket Sphinx and Wit.ai. Which were tested and compared to determine which is the best for the proposal. In Table 4, the comparison between speech recognition software is displayed. When comparing Wit.ai with Google Cloud Speech API, it was observed that in the long term it was going to pay for the use of the service, by using a large number of characters. Compared to Pocket Sphinx, when this software was tested, speech recognition was low-quality, and the words being spoken were not understood by the system. For this reason, Wit.ai was chosen for having better speech recognition, works with several languages (English, Spanish, etc.), has an interactive interface with the developer, is easy to use, and is free [13].

Table 4. Voice recognition software

Voice Recognition Software	Google Cloud Speech API	Pocket Sphinx	Wit.ai
Internet connection	Yes	No	Yes
Price	4 USD per million characters	Free	Free
Open Source	No	Yes	No
Recognition Level	High	Medium	HIGH
Languages	80 to 110 languages	English/Spanish through bookstores	50 languages: English, Spanish, Chinese, etc.

4 System architecture

The gas leak detection and surveillance system have a home automation system structure, which is made up of sensors, actuators, and a control module.

There are several components necessary to develop the gas leak detection and surveillance system. The following block diagram (See Figure 1) shows the connection of the components and the architecture of the system

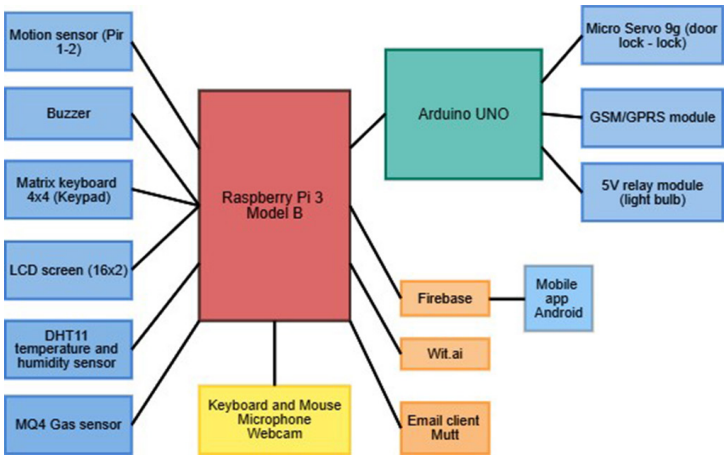

Fig. 1. System architecture

In the diagram, you can see sensors, actuators, and equipment that connect to Raspberry Pi. It has 40 pins, to which 2 PIR sensors (pins 11 and 12), a buzzer (pin 7), a matrix keyboard (pins 29, 31, 33, 35, 32, 36, 38, and 40), and a screen were connected LCD (pin 3 and 5), a DHT11 temperature and humidity sensor (pin 8) and an MQ4 gas sensor (pin 13). Also, Raspberry Pi has 4 USB ports, in which it connects: keyboard and mouse, an Arduino UNO board, a microphone with a Jack to USB port adapter, and a USB camera. Through a Wi-Fi connection, it communicates with Firebase, the cloud database, and the communication link between the system and the mobile application, with the Wit.ai voice recognition software and with the Mutt email client to sending emails.

Raspberry Pi is responsible for processing data and performing programmed actions.

The Arduino Uno board is made up of 6 analog pins and 14 digital pins. Servomotors are connected to pins 10 and 9, and the relay module (focus) is connected to pin 11. The SIM 900 GSM/GPRS module connects on top of the Arduino board and uses the Tx and Rx pins (pins 8 and 7) for communication and sending text messages. The connection between Arduino Uno and Raspberry Pi is through the USB port, through serial communication.

5 System development

In order for the system to be interactive with the user, the entry of options through a control panel is considered, which consists of a matrix keyboard, in which the following options can be entered: option A: activate full alarm (surveillance and detection gas leak); option B, deactivate the complete alarm; option C, activate only gas leak detection; option D, disable gas leak detection; option *, shows temperature and humidity when the systems are not active, and option #, to activate the voice recognition system.

The surveillance system works like the alarm systems on the market, to give it greater value, other features were added. The system works when an intruder enters the property, the alarm emits a sound, sends an alert text message to the owner, turns on a spotlight from 6:00 pm to 4:00 am (depending on the time), takes a picture, saves it in a Raspberry Pi folder with the date and time, sends it to the owner's email, and issues a message advising that the doors will be closed as a prevention method. So that many messages or emails are not sent, after 30 seconds have passed after detecting a presence, the system opens the lock, and if it detects a presence again, it performs the previous procedure (See Figure 2).

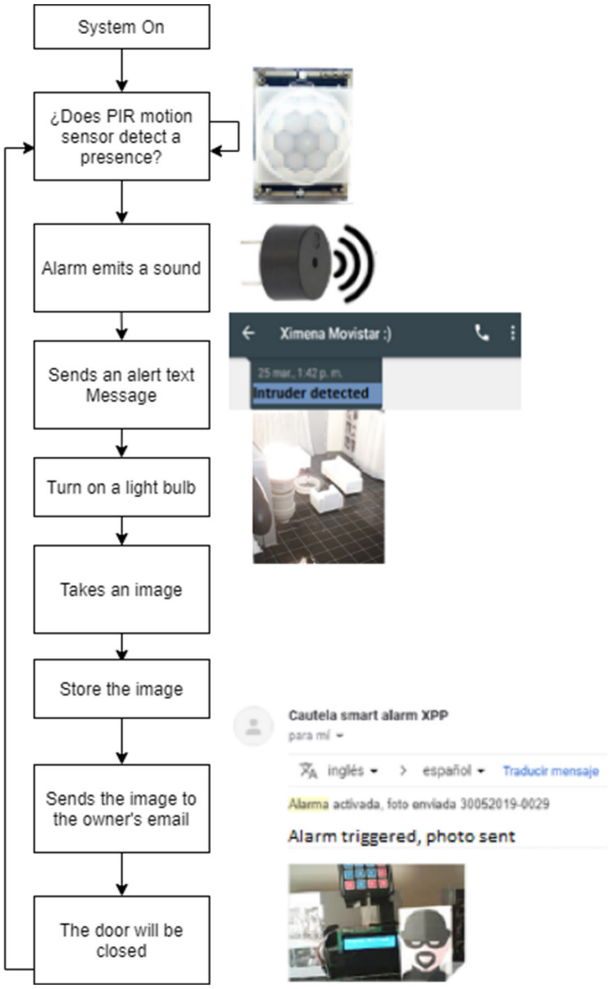

Fig. 2. Surveillance system

The gas leak detection system works when the gas sensor detects the presence of methane gas, then it sounds an alarm (buzzer), sends an alert text message, and displays an alert message on the screen. If the temperature is higher than 35 degrees, it shows the temperature as an on-screen alert message. In order not to send multiple text messages, it waits for 15 seconds, when the time passes and the system detects a gas leak again, it performs the procedure described above.

To activate or deactivate the gas leak detection and surveillance system by the keypad, the user is prompted to enter a password. The user has three attempts to enter the correct password, otherwise, the system hangs for 20 seconds.

To use the voice recognition system, the user can enter the options # via the matrix keypad. The commands that can be spoken are: “activate alarm”, “deactivate alarm”, “activate gas”, “deactivate gas”, “tell me temperature”, “tell me humidity”, “turn on

light" and "turn off light". For the enable or disable options, the user is prompted to say the keyword to enter the requested option. The user has three attempts to enter the code word, if a valid word is not spoken, the system returns to the main menu. To enter the other commands the system must be disabled, if you say "tell me temperature" or "tell me humidity" the system displays the temperature and humidity in that room of the house where the sensor is located, in this case in the kitchen. When you say "turn off light" or "turn on light", the light bulb will emit light or turn off, as the case may be. For the speech recognition system, the speech recognition software Wit.ai was used, which has the "Speech to Text" service [13]. When a command is spoken, it is recorded and sent to the Wit.ai servers, where the audio is converted to text, the text is returned to the system and compared to which the command entered. Speech recognition is a type of artificial intelligence, which it establishes communication between man and an intelligent device by means of human language [14].

The system has a mobile application (See Figure 3) that allows the user to manage and control the system from anywhere in the world with an Internet connection. Through the mobile application you can change the password used to enter the control panel (activate or deactivate), change the keyword (voice recognition system), and the email to which the image is sent. In the application you can view the status of the alarm (activated or deactivated), you can activate or deactivate any of the systems and you can observe the temperature and humidity of the home in real-time. Communication between the system and the mobile application is done through Firebase. Firebase is a platform that enables a cloud database and other services to develop web and mobile applications [15]. In the interim of using the Firebase database, when the user makes any changes to data, it is reflected both in the system and in the mobile application.

Fig. 3. "Cautela" mobile application [16]

6 Results

To test the gas leak detection and surveillance system, a one-story house model was implemented with the main door and 5 rooms. Sensors, actuators, and equipment necessary for the operation were installed in this model. (See Figure 4).

Fig. 4. Prototype gas leak detection and surveillance system

The proposed system has a cost of 268 dollars. In Table 5, prices for the equipment, sensors, actuators, cables used, and the development of the mobile application are presented.

Table 5. Gas detection and surveillance system costs

Equipment	Price ($)
Raspberry Pi 3 Model B	42
Arduino UNO	10
Module Shield GSM/GPRS SIM 900	14
Peripherals (keyboard, LCD screen, buzzer, microphone, camera)	40
Actuators (Led, Spotlight, Micro servo 9g)	11
Sensors (PIR, Dht11 temperature, MQ4 gas)	13
Cables	7
Mobile app	131
Total	**268**

Functionality and usability tests were carried out in different scenarios, which are divided into tests of the surveillance system, the gas leak detection system, and the voice recognition system.

Table 6 shows the tests carried out on the surveillance system, the tests carried out in the morning vary with respect to those at night, in relation to shipping times. At night when the light bulb turns on, it increases the time it takes to send the text message, causing the other options to delay as well. For the system to detect the presence of an intruder again, an average of 53 seconds must pass according to the tests carried out.

Table 6. Time surveillance system test

Test	Detect Presence	Turn on the Light	Time to Send a Text Message (Second)	Time to send an Email (Second)	Close the Door	Mode Home
P1 10:40 am	Yes	No	9 secs.	20 secs.	14 secs.	Full
P2 11:20 am	Yes	No	9 secs.	20 secs.	17 secs.	Empty
P3 7:16 pm	Yes	Yes	17 secs.	19 secs.	21 secs.	Empty
P4 7:27 pm	Yes	Yes	16 secs.	17 secs.	20 secs.	Empty
P5 8:07 pm	Yes	Yes	16 secs.	19 secs.	20 secs.	Empty

Table 7 shows the tests performed on the gas leak detection system, the time it takes to detect gas depends on the time the sensor is turned on, this type of sensor needs 50 seconds to heat up. In the first test made to the sensor it took longer, because the system had just been turned on. In the other tests the system is stable, therefore gas detection is faster.

Table 7. Tests performed on the gas leak detection system

Tests	Detect Gas Leak	Display Message on LCD Screen	Time to Send a Text Message	Mode Home
P6	Yes	50 secs.	70 secs.	Empty
P7	Yes	4 secs.	9 secs.	Full
P8	Yes	5 secs.	10 secs.	Full
P9	Yes	4 secs.	9 secs.	Full

7 Conclusions and future work

The proposed gas leak detection and surveillance system is a good alternative to provide greater security to the home, easy to implement, has an intuitive mobile application, has a voice recognition system, and has better features than other surveillance systems that exist on the market as detecting a gas leak or alerting the owner by text message. Despite its size, the Raspberry Pi minicomputer has great processing power which allowed the development of this project, being necessary to integrate the Wit.ai voice recognition system, since it works with ambient noise and identifies the voice of a person without much effort. To store data, the Firebase cloud database was used, which allowed the system to be manageable and controllable from a mobile phone with an Internet connection. Finally, by using SIM 900 GSM/GPRS module, it was possible to send alerts through text messages, without depending on an Internet connection. The tests were carried out on a single-family home model and showed that the system works in different scenarios. The system is modular and scalable, that is, it allows adding additional sensors, actuators, and peripherals to provide greater characteristics, taking into account electricity consumption so as not to overload the Raspberry Pi or the Arduino board.

As work future in this time of COVID-19, a digital thermometer could be used that would work with the MLX90614 sensor. This sensor would measure people's temperature as they enter the home, as an automatic prevention method.

8 Acknowledgment

To Universidad Catolica de Santa Maria, for providing the necessary equipment and the space provided to develop the project

9 References

[1] El Correo Newspaper, "Young people are the most affected with unemployment due to the pandemic," https://diariocorreo.pe/edicion/arequipa/jovenes-son-los-principales-afectados-con-el-desempleo-causa-de-la-pandemia-942559/. Revised 15 Jan, 2021.

[2] Peru Instituto Nacional de Estadística e Informática, "Technical Report Statistics on crime, public safety and violence," No 2 October 2021. https://www.inei.gob.pe/media/MenuRecursivo/boletines/estadisticas_de_criminalidad_seguridad_ciudadana_abr-jun2021.pdf. Revised 15 Jan, 2021.

[3] Instituto Nacional de Defensa Civil, "What are the main causes of a fire in the home?" https://rpp.pe/prevenir-es-simple/cuales-son-las-principales-causas-deincendio-en-el-hogar-noticia-1088572. Revised 15 Jan, 2021.

[4] J. Alpaca, J. Esquicha, and K. Rosas, "Sistema de Atención a pacientes hospitalizados utilizando Raspberry Pi con Cámara Megapixel y Open CV," 15th LACCEI International Multi-Conference for Engeineering, Education and Techonology, 2017. https://doi.org/10.18687/LACCEI2017.1.1.223

[5] I. S. Areni, A. Waridi, I. Amirullah, C. Yohannes, A. Lawi, and A. Bustamin, "IoT-based of automatic electrical appliance for smart home," International Journal of Interactive Mobile Technologies, vol. 14, no. 18, pp. 204–211, 2020. https://doi.org/10.3991/ijim.v14i18.15649

[6] B. Pérez-Camacho, J. Gonzales-Calleros, and G. Rodríguez-Gómez. Design of Home Energy Management System Using IoT Data Flow. HCI-COLLAB 2019, CCIS 1114. pp. 165–176, 2019. Springer, Mexico. https://doi.org/10.1007/978-3-030-37386-3_13

[7] J. Esquicha-Tejada and J. Copa-Pineda, "Alternatives of IoT irrigation systems for the gardens of Arequipa," International Journal of Interactive Mobile Technologies, vol. 15, no. 22, pp. 4–21, 2021. https://doi.org/10.3991/ijim.v15i22.22653

[8] J. Esquicha-Tejada and J. Copa-Pineda, "Integration of an IoT system—Photovoltaic system to optimize the consumption of drinking water in the irrigation of gardens in the City of Arequipa," Proc. LACCEI Int. Multi-conference Eng. Educ. Technol., no. July 2020, pp. 27–31, 2020. https://doi.org/10.18687/LACCEI2020.1.1.212

[9] P. López-Roldán and S. Fachelli, "Metodología de la Investigación Social Cuantitativa." Universitat Autònoma de Barcelona, 2015.

[10] Instituto Nacional de Estadística e Informática, "Perú: Perfil Sociodemográfico." [Online] Disponible en: https://www.inei.gob.pe/media/MenuRecursivo/publicaciones_digitales/Est/Lib1539/libro.pdf

[11] C. Shalini and I. V. Prakash, "IoT based industrial sensor monitoring and alerting system using Raspberry Pi," IOP Conference Series: Materials Science and Engineering, vol. 981, no. 4, 2020. https://doi.org/10.1088/1757-899X/981/4/042010

[12] Y. Irawan, R. Wahyuni, M. Muhardi, H. Fonda, M. L. Hamzah, and R. Muzawi, "Real time system monitoring and analysis-based internet of things (IoT) technology in measuring outdoor air quality," International Journal of Interactive Mobile Technologies, vol. 15, no. 10, pp. 224–240, 2021. https://doi.org/10.3991/ijim.v15i10.20707

[13] A. D. Stoica et al., "The Impact of Romanian Diacritics on Intent Detection and Slot Filling," in 2020 22nd IEEE International Conference on Automation, Quality and Testing, Robotics—THETA, AQTR 2020—Proceedings, 2020, pp. 0–5. https://doi.org/10.1109/AQTR49680.2020.9129947

[14] A. H. Al-Ajmi and N. Al-Twairesh, "Building an arabic flight booking dialogue system using a hybrid rule-based and data driven approach," IEEE Access, vol. 9, pp. 7043–7053, 2021. https://doi.org/10.1109/ACCESS.2021.3049732

[15] M. A. Mokar, S. O. Fageeri, and S. E. Fattoh, "Using firebase cloud messaging to control mobile applications," in Proceedings of the International Conference on Computer, Control, Electrical, and Electronics Engineering 2019, ICCCEEE 2019, 2019, pp. 2–6. https://doi.org/10.1109/ICCCEEE46830.2019.9071008

[16] "XimenaPerezP/Alarm-Application-", GitHub, 2019 [Online]. Available: https://github.com/XimenaPerezP/Alarm-Application-

10 Authors

Ximena Pérez-Palomino is a Systems Engineer from the Universidad Catolica de Santa María, UCSM (2016). She is currently a Professor at UCSM in the Institute of Informatic and Tecnical Analist in Bantotal. Researcher in Data Business Inteligence and Information Technologies. Email: xperezp@ucsm.edu.pe

Karina Rosas-Paredes is a Systems Engineer from the Universidad Catolica de Santa Maria. Master in Information Systems and Higher Education from the Universidad Catolica de Santa Maria. Doctor in Systems Engineering from the Universidad Nacional Federico Villarreal. She is CCNA CISCO certified. Coordinator of the Engineering Area of CICA-UCSM. Researcher in Information Technologies. Director of Innovation and Development Vice Rectorate of Research Universidad Catolica de Santa Maria, Peru. Email: kparedes@ucsm.edu.pe

José David Esquicha-Tejada is a Systems Engineer from the Universidad Catolica de Santa Maria, UCSM (2008). He holds a Second Specialty Professional Degree in Systems Auditing and Information Security at UCSM (2019). Master in Strategic Telecommunications Management at the Miguel de Cervantes European University (2013). He is a candidate for a Doctor of Environmental Sciences and Renewable Energies at the Universidad Nacional de San Agustín de Arequipa. He is currently an Assistant Professor at UCSM in the Faculty of Physical and Formal Sciences and Engineering. His research interests include the Internet of Things (IoT) and educational technology. Email: jesquicha@ucsm.edu.pe

Article submitted 2022-02-12. Resubmitted 2022-03-07. Final acceptance 2022-03-08. Final version published as submitted by the authors.