Research Article

IoT System for the Continuous Electrical and Environmental Monitoring into Mexican Social Housing Evaluated under Tropical Climate Conditions

P. E. Mendez-Monroy,[1] E. Cruz May ⓘ,[2] M. Jiménez Torres ⓘ,[2] J. L. Gómez Hernández,[3] M. Canto Romero,[3] I. Sanchez Dominguez,[1] O. May Tzuc ⓘ,[4] and A. Bassam ⓘ[5]

[1]*IIMAS-Merida, National Autonomous University of Mexico, Parque Científico y Tecnológico de Yucatán, km. 55 Carretera Sierra Papacal-Chuburná, CP 97320 Sierra Papacal, Yucatan, Mexico*
[2]*Postgraduate in Renewable Energy, Autonomous University of Yucatan, Av. Industrias no Contaminantes, Merida, Yucatan, Mexico*
[3]*Bachelor of Renewable Energy, Faculty of Engineering, Autonomous University of Yucatan, Av. Industrias no Contaminantes, Merida, Yucatan, Mexico*
[4]*Faculty of Engineering, Autonomous University of Campeche, Campus V, Av. Humberto Lanz, Col. Ex Hacienda Kalá, C.P. 24085, San Francisco de Campeche, Campeche, Mexico*
[5]*Faculty of Engineering, Autonomous University of Yucatan, Av. Industrias no Contaminantes, Merida, Yucatan, Mexico*

Correspondence should be addressed to O. May Tzuc; oscajmay@uacam.mx and A. Bassam; baali@correo.uady.mx

This work presented the design, development, and implementation of a low-cost IoT system for real-time monitoring of electrical consumption and environment parameters inside social housing. The IoT monitoring system is composed of a set of remote measurement nodes that wirelessly sense and transmit records of the air temperature and relative humidity inside the house as well as the voltage, current, and power consumed by several electrical devices. A server coordinator composed of a Raspberry Pi was used for the interaction with the sensors through the Message Queue Telemetry Transport (MQTT) protocol and allowed the measured data packaging, fragmentation, transfer, storage, and cloud upload for remote multiuser visualization. The system was experimentally evaluated in an inhabited single-story house under tropical climate conditions. Operation tests indicate a successful performance of the protocols implemented for remote visualization and cloud storage. Moreover, an analysis of measured data allows the feasibility of identifying the occupants' energy consumption patterns and their relationship with the search for comfortable environmental conditions. Thus, the proposed friendly framework is a promising alternative to energy management. Its implementation in embedded Linux-based systems provides the flexibility to integrate control strategies based on artificial intelligence.

1. Introduction

The application of mechanisms and instruments to obtain satisfaction conditions into indoor environments has become a relevant issue for the energy and building sectors. On the one hand, thermal dissatisfaction seriously affects people's health, while on the other hand, artificial air conditioning for space cooling in buildings represents a significant challenge in terms of energy efficiency [1, 2]. This relationship between the thermal satisfaction levels and the energy demand to achieve it is more noticeable during extreme heat events. Specifically, in tropical or hot climate zones (characterized by present indoor temperatures above 26°C and exterior temperatures above 32°C), the use of air conditioning and ventilation mechanisms to cool and dehumidify interior spaces can represent more than 50% of buildings' energy

consumption [3]. Moreover, in these zones, due to population growth along with the temperature effects caused by climate change, electricity energy used in social housings is expected to have a growing influence on both local and global energy consumption rates [4, 5]. Considering this trend, monitoring strategies are necessary to minimize the overall energy demand while guaranteeing occupants' thermal satisfaction.

In recent years, the interest in implementing embedded systems linked to the concept of "Internet of Things" (IoT) has grown to represent a versatile tool with the capability of visualization and accessibility from practically any geographical location by using mobile devices. Besides, characteristics of IoT as compatibility with different devices regardless of the manufacturer, as well as its ability to acquire data from the environment where they are located, process it, and transmit it through the cloud, making this technology an ideal option for the development of low-cost monitoring systems [6]. Based on this, IoT has been successfully used to continuously monitor of diverse processes. This approach has shown broad application potential in the agricultural sector to report in real time to farmers the comfort level of comfort in crops (temperature, humidity, pH, etc.) [7, 8]. It has also been quickly adapted to the medical field for the storage and remote visualization of patients' vital signs (heart rate, oxygen saturation, blood glucose level, etc.) [9, 10]. In the energy sector, this approach has proven to be one of the main trends to reduce instrumentation costs and streamline the decision-making process to improve the performance of large photovoltaic installations [11, 12], the real-time monitoring of offshore wind farms [13, 14], and the automation of households to integrate them into smart grids [15, 16].

In the specific case of monitoring systems for buildings, Table 1 contains the studies developed in recent years based on IoT technology. According to the table, most of the studies carried out are based on monitoring indoor environmental parameters [17–28], the most interesting being the temperature and relative humidity inside the enclosure. However, it is essential to emphasize that many focus their IoT systems for air quality applications [17, 21, 26, 27]. This leads to the fact that not all consider thermal comfort conditions as a priority of the monitoring systems designed for interiors. In the cases of Karami et al. [17], Ferdoush and Li [21], and Ponce and Gutiérrez [27], the measurement of the ambient temperature represents an indicator to consider the levels of specific harmful agents in the air, which is why they have primarily focused on laboratory conditions. Another important factor in analyzing from the perspective of thermal comfort is the validation of IoT systems for measuring indoor conditions in inhabited buildings. Most of the reported systems are validated under laboratory conditions [17, 18, 22–24], with few studies reporting the interaction among users, postoccupied buildings, and sensor networks developed. Silva et al. [19] developed an Arduino-based IoT system to study the thermal satisfaction effects of churchgoers under a Mediterranean climate. For their part, Carre and Williamson [20] and Martín-Garín et al. [25] designed smart sensor networks (also based on Arduino)

to study the conditions of thermal satisfaction in previously occupied apartments, both for Mediterranean and Oceanic climatic conditions, respectively. In the same context, other studies have addressed scenarios such as malls [26], offices buildings [21], and universities buildings [27]. Finally, there is great interest in the evaluation under real climatic scenarios, intending to analyze the phenomena of the structure and materials of the building in interaction with the climate as well as the routine of the occupants.

Regarding the monitoring of electrical appliances and devices, there are few studies in this area. However, the most mature studies base their approach to energy consumption in buildings [29, 33]. Similar to the case of thermal comfort variables, laboratory conditions prevail for the analysis of developed energy monitoring IoT systems [29–33]. Furthermore, the state-of-the-art review reveals that although monitoring systems based on IoT have been designed for buildings, they have not addressed the interaction between energy demand and thermal satisfaction, playing these concepts in isolation. Finally, it has been found that none of the studies carried out focused on the implementation of these monitoring systems for homes under actual environmental conditions, which is relevant as one of the primary buildings for people, where a large amount of electricity is consumed as well as where the best comfort conditions are required.

An essential factor to highlight from Table 1 is the hardware devices, where Raspberry Pi and Arduino are the most used for IoT applications [17–25, 27–31, 33]. The foregoing is relevant given that although both are development boards with similar characteristics, such as operating in low power mode and having easy connection to analog and digital sensors, they have different architectures and technical parameters. On the one hand, the Arduino is a microcontroller that shows its best performances in hardware applications, while the Raspberry Pi is a Linux-based minicomputer that can be linked to the internet and run various types of software and firmware. Studies have shown that because the Arduino requires additional components such as SD cards and shields for storage and web connection, they do not make it viable for data management. However, its low cost and easy programming show that its best performance occurs when used as a sensor node [34]. For its part, the Raspberry Pi works better as a central control system; this is because its peripherals have an ethernet and Wi-Fi connection. These benefits allow the Raspberry Pi to be an important resource in the execution of multiple jobs that seek to link various areas in the IoT. For example, Mudaliar and Sivakumar [34] implemented an IoT system using a Raspberry Pi for monitoring various energy parameters (voltage, current, power, power factor, etc.) in a switchgear industry manufacturing company and analyzed energy consumption day by day. Nadafa et al. [35] used the Raspberry Pi to design a home security system. Through programming, the intrusion detection system allows the acquisition of visuals on an Android device through a Wi-Fi connection. Bora et al. [36] proposed a real-time health monitoring system that with the use of Arduino acquires the heart rate, body temperature, heart rate and electrical activity (ECG), and location. Said data is

TABLE 1: Monitoring systems developed in recent years for the measurement of indoor environmental variables and the consumption of electrical devices in buildings.

Publication	Summary	Indoor measured parameters	Hardware device	Building type	Climatic conditions
Indoor environmental parameters:					
Karami et al. [17]	Continuous measurement toolbox for indoor environmental quality	Temperature, humidity, air quality, and illuminance	Arduino	Indoor environment	Laboratory conditions
Lewis et al. [18]	Digital environmental monitoring	Temperature, pressure, and humidity	Raspberry Pi	Indoor environment	Laboratory conditions
Silva et al. [19]	Prototype to measure environmental parameters	Temperature, humidity, and ventilation	Arduino	Church	Mediterranean climate
Carre and Williamson [20]	Indoor environment data collection	Temperature, humidity, light, sound level, air velocity, and air quality	Arduino	Apartment	Mediterranean climate
Ferdoush and Li [21]	Enable wireless sensor network	Temperature and humidity	Arduino & Raspberry Pi	Office building	Subtropical climate
Meana-Llorián et al. [22]	Adjust the indoor temperature	Temperature	Raspberry Pi	Indoor environment	Laboratory conditions
Vujović and Maksimović [23]	Adjust the indoor temperature	Temperature	Raspberry Pi	Indoor environment	Laboratory conditions
Sung and Hsiao [24]	Analyze indoor environmental data	Temperature, humidity, CO_2, wind speed	Arduino	Indoor environment	Laboratory conditions
Martín-Garín et al. [25]	Building environmental monitoring	Temperature and humidity	Arduino	Apartment	Oceanic climate
Kalia and Ansari [26]	Portable device to measure indoor environmental parameters	Temperature, humidity, pressure, and dew point	ESP8266	Commercial center	Tropical savanna climate
Ponce and Gutiérrez [27]	Predicting climatic conditions inside an enclosure	Temperature and humidity	Raspberry Pi	University building	Laboratory conditions
Shinde et al. [28]	Environment monitoring system	Measure air pollutant concentrations, temperature, and humidity	Raspberry Pi	Outdoor variables	Laboratory conditions
Electrical energy parameters:					
Arumuga Perumal et al. [29]	Building monitoring system to understand energy consumption	Voltage and current	Raspberry pi	University building	Laboratory conditions
Abate et al. [30]	Smart electric meter	Voltage and current	STM32F2	Indoor environment	Laboratory conditions
Agyeman et al. [31]	Design and implementation on energy monitoring system in homes	Power consumption	Raspberry Pi & Arduino	Indoor environment	Laboratory conditions
Pereira et al. [12]	Measure PV and meteorological variables	Voltage, current, PV module temperature, solar irradiance, and relative humidity	Raspberry Pi	PV system	Tropical weather
Matsui et al. [32]	Sensing electricity consumption.	Electricity power	Sassor Inc.	Indoor environment	Mild weather
Shapsough et al. [33]	Implementation of PV system for the purpose of evaluating power loss due to dust	Measure in real time IV curves	Raspberry Pi	PV system	Laboratory conditions

processed to the Raspberry Pi, which stores the data through a Wi-Fi connection and sends it to various users. Samson et al. [37] proposed an intelligent power monitoring system through the Raspberry Pi. This system is divided into two important parts: The first, the Electricity Board (EB), inter-acts entirely through the website. For the second part, the user has access to an energy monitoring application. In this system, all communications are carried out through the Wi-Fi module of the Raspberry Pi. Shapsough et al. [33] presented the design and implementation of an IoT-based solar

monitoring system for large-scale distributed solar installations and in smart cities. Through the implementation of the Raspberry Pi, this design allows the sending of IV curves from modules at different points to evaluate the power loss due to dust. Finally, Hafid et al. [38] using the Raspberry Pi and two systems on Chip (Soc) solutions, implemented a 3-lead complete ECG recording and an impedance cardiograph. This is with the purpose of developing research with affordable and available components.

Based on those mentioned above, this work implements the IoT technology for both energy efficiency and thermal comfort applications in houses. The study has focused on developing a low-cost monitoring system designed to measure energy and indoor environmental parameters. A central control system composed of a Raspberry Pi device was used to interact with the sensors through various communication protocols, packaging, fragmentation, data transfer, and storage measured in the cloud. The work uses a case study of the Mexican buildings known as social housing, which cover 80% of the country's homes [39]. Specifically, the evaluation of the proposed IoT system is addressed under tropical climate conditions, since according to the National Institute of Statistics, Geography and Informatics (INEGI), social housings under this climate congregate 15% of the country's air conditioning and mechanical ventilation equipment [40]. Therefore, this represents an essential focus in terms of energy efficiency and thermal comfort.

The work points to two main objectives of great interest to improve energy performance in buildings that have not yet been addressed:

(i) It contemplates the development and validation of a real-time monitoring system applied to the case of a postoccupied building under a climate scenario not yet addressed (tropical climate conditions)

(ii) It implements the IoT approach as a tool to associate the occupants' routine with their habits of environmental comfort and energy consumption

The work's main contribution is the design of a friendly framework with minimal development time and easy implementation as a tool to monitor energy consumption online and variables that affect the thermal comfort of the occupants. The most outstanding characteristic of the framework is to be based on free software and hardware, making it an economical alternative to be implemented in buildings with reduced space and low budgets, such as social housing and apartments. Furthermore, as it is based on an embedded Linux-based system (Python), it presents the flexibility of extending its application by hybridizing IoT with artificial intelligence for future energy management strategies.

The content of the work is divided as follows: Section 2 provides a general description of the development, operation, coding, and visual interfacing of the IoT monitoring system. Section 3 presents the evaluation of the system in a house of social interest in tropical climatic conditions, involving the feasibility of implementation and the analysis of measurements. Section 4 engages in a discussion of further developments, and Section 5 presents the conclusions.

2. IoT Monitoring System

The low-cost IoT monitoring system proposed in this investigation is developed for the continuous and long-term measurement of electrical energy consumption and environmental data of social housing, residential apartments, and similar. It is designed to operate in a wireless network that allows the data communication of sensors located in different building areas and its subsequent collection, storage, and display by a cloud service. The IoT monitoring system comprises two main elements, the remote sensor nodes (RSN) and the server coordinator (SC), illustrated in Figure 1.

The RSN is the hardware element in charge of collecting, digitizing, and transmitting the values of the physical variables. The RSN is divided into two types: (a) the first is designed for measuring the voltage, current, and power consumed by home electrical devices; (b) the second is intended to measure the environmental variable of relative humidity and temperature inside and at the surrounding environment of the home. The inclusion of a wireless communication structure facilitates the incorporation or replacement of new RSNs without the need for modifications (or changes) in the system's current configuration. Wireless data transmission is enabled through Wi-Fi interconnection technology. Using the Wireless Local Area Network (WLAN) and the high-level communication protocol MQTT, the RSN measurements are sent to the SC.

The SC incorporates a set of high-level software programs (Application Programming Interfaces (APIs)) that execute data acquisition, monitoring, cloud storing, and visualization. For this purpose, the Raspberry Pi was used; it is a cheap and low-power reduced board computer that operates under a distribution of the open-source Linux operating system [41]. Into the Raspberry Pi board was implemented different APIs intended to perform specific operation processes on data. Mosquitto API [42] is used for assigning tags to RSN and enabling communication. Node-RED [43] is used for decoding, defragmenting, and retrieving the data. InfluxDB [44] creates a backup database, and ThingSpeak API [45] enables the IoT remote access by an HTTP protocol, providing end-users a real-time visualization of the last 60 measured data. Likewise, it generates a CSV file with all the captured data for subsequent analysis, if required.

2.1. Remote Sensor Nodes

2.1.1. Electrical Energy Monitoring Sensors. The electrical variables (voltage, current, and power) are measured using the commercial device POWR2, developed by ITEAD [46]. It is a Wi-Fi-based wireless switch that can connect to a wide range of appliances. Among its main applications are home automatization, energy monitoring, overload protection, and timing functions. The POWR2 was selected by its cost-effective and open hardware/software manufacturing
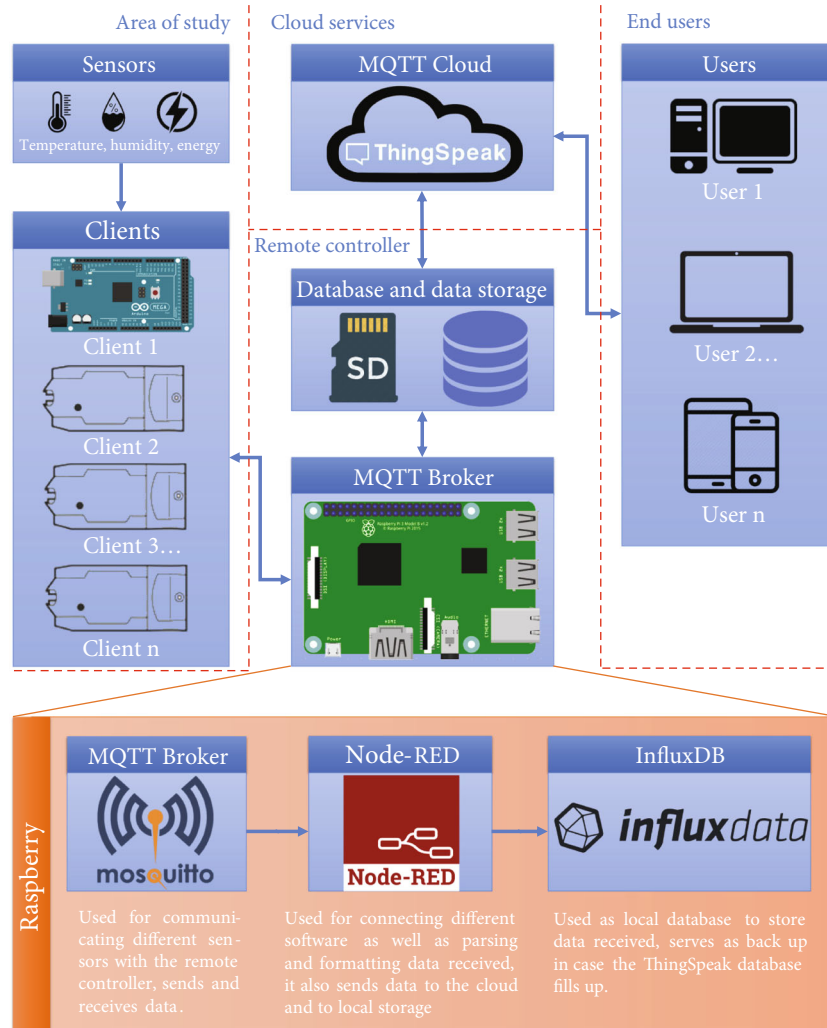
FIGURE 1: IoT monitoring system prototype operational diagram.

equipped with serial-TTL programming ports for burning external firmware (Figure 2(a)), enabling it to be used as a complement to wireless platforms and embedded systems [47]. Conventionally, the POWR2 operates transmitting data to a cloud platform via the smartphone application eWeLink [48]. However, the conventional mode restricts access to the device database. For this work, the control of POWR2 was carried out using the custom firmware ESPurna [49].

The POWR2 is designed to work in an operating range of 90~250 V AC to tolerate a maximum current of 16 A and a maximum power of 3.5 kW, suitable for the power measurement of most household appliances. To measure the electrical consumption variables, the device is equipped with the HLW8012 integrated circuit (IC), a single-phase energy monitor chip by HLW Technology [50]. The IC monitors the analog voltage and current, converting these into digital values of root mean square (RMS) voltage, current, and active power. The IC integrates a self-calibration process based on current and voltage peak-to-peak values, providing measurement accuracy of ±1.0%. The sensor has a response time of 2-3 s, enough for energy consumption

measurements in appliances that do not vary too much in their operating time. Additionally, each POWR2 incorporates an ESP8266 module [51] integrated directly into its printed circuit board (PCB) for remote communication via Wi-Fi (Figure 2). A more detailed description of the POWR2 configuration and electronic instrumentation is given in [47].

2.1.2. Indoor Environmental Monitoring Sensors. To monitor the indoor and surrounding environmental variables, a set of RSN were manufactured (http://www.arduino.cc). These are designed to act as dataloggers, recording sensed humidity and temperature values at building interest zones. The Arduino platform was selected because of its versatility, big library repository, and extensive online support community. Furthermore, it has proven to be reliable and easy to implement in monitoring networks [6, 10, 52].

The data logger system was manufactured using the Arduino Mega, a development board based on the microcontroller (MCU) ATmega2560 [53]. It is an 8-bit board MCU with 54 digital pins, 16 analog inputs, and 4 serial ports, characteristics more than enough for the monitoring

FRONT

LED
Switch
Serial-TTL
GND
TX
RX
VCC
Relay

BACK

Energy monitor IC
HLW8012
Flash memory
PN25F32
Wi-Fi module
ESP8266

Wi-Fi Antenna

Bridge rectifier
ABS8

N: Neutral line
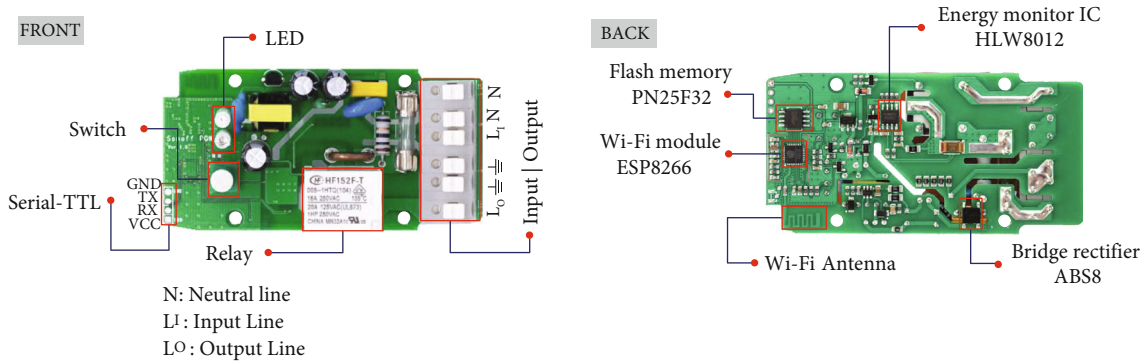L^I: Input Line
L^O: Output Line

FIGURE 2: POWR2 device used for remote measurement of electrical consumption of household appliances.

system. The Arduino Mega incorporates a USB type B port to establish communication with computers and a power jack powered by either AC-DC converters or batteries. Moreover, the board produces a regulated 5.0 V and 3.3 V outputs to provide the supply voltage for sensors and electronic devices.

To achieve monitoring and wireless transmission of environmental variables, various modules were integrated into the Arduino Mega board. The MCU is designed to measure every 5 minutes; however, the measurement time can be modified via software. Figure 3 illustrates the schematic diagram of the developed environmental wireless measurement device. A detailed description of the functions of each element incorporated in the Arduino MEGA Board is provided below:

(i) *Environment sensor DHT22* [54]: it is a low-cost digital sensor based on a capacitive element for the measurement of relative humidity and an NTC thermistor for temperature. The sensor contains a high-resolution ADC (16 bits) to simplify hardware connections and provide data using a digital signal based on a one-wire protocol. The humidity measuring range covers from 0 to 100% with an accuracy of 2-3% RH, while the temperature measuring range is from -40 to 125°C with ±0.5°C accuracy, both complying with the limits defined in the standards ISO 7726 [55], EN 16242 [56], and ANSI/ASHRAE 55 [57]. Its response time is 2 s in open air, enough time to observe the slow heat transfer phenomena that affect the variation of the temperature and relative humidity inside buildings [58].

(ii) *Backup storage data system*: it consists of a micro SD card breakout-board (by CATALEX [59]) and a Real-Time Clock (RTC) DS3231 [60]. It is designed to temporarily store the recorded information in a 2 GB micro SD memory and avoid data loss in case of faults in the WLAN. Measurements are stored with their respective date and time in a "txt" file; during data storage, these are separated by tabs to facilitate their possible use in spreadsheets.

(iii) *LCD check display*: its function is to display on a 16 × 2 LCD screen the connection or operation faults of the components that make up the RSN,

for immediate reparation. A switching element is connected to the LCD screen power supply to guarantee energy savings, turning it on only when faults are detected.

(iv) *Support cooling system*: since the RSN is intended to operate in tropical climate regions with temperatures above 34°C in summer, a piece of cooling equipment was incorporated to avoid circuit overheating during the operation. It is composed of a temperature sensor and two DC fans. The temperature sensor is placed on the peripheries of the Arduino board; when the sensor detects that the temperature exceeds 40°C, the fans are activated, generating forced convection cooling. The fans turn off when the temperature drops below 35°C. This reduces the temperature inside the case and prevents damage to the remote sensing device.

(v) *Emergency energy supply system*: this works as a backup source that comes into operation if the electrical grid does not supply energy to the RNS due to failures. It is made up of a 9 V battery connected to a relay. When the primary power source is cut off, the relay switches by connecting the battery to the Arduino board. This allows the RSN operation while the primary power source is reset.

(vi) *Wi-Fi module*: an ESP8666 module [51] mounted on a breakout board was used to establish wireless communication. This was implemented to incorporate the RSN into the WLAN and transmit the monitored data to the SCN.

2.2. Server Coordinator Node. The central control system with which all sensors communicate is a Raspberry Pi model B. It is a low energy consumption microcomputer running with the Raspbian operating system, a Debian-based Linux open-source distribution [41]. The operating system and data are stored on an SD card (with a minimum of 4 GB). The Raspberry Pi operates with a 5 V power supply (and just over 700 mA) provided by a micro USB connector. Additionally, the device has various peripheral elements such as USB ports, ethernet ports, general-purpose input, output pins, and HDMI, SCI, and DSI ports. The Raspberry Pi
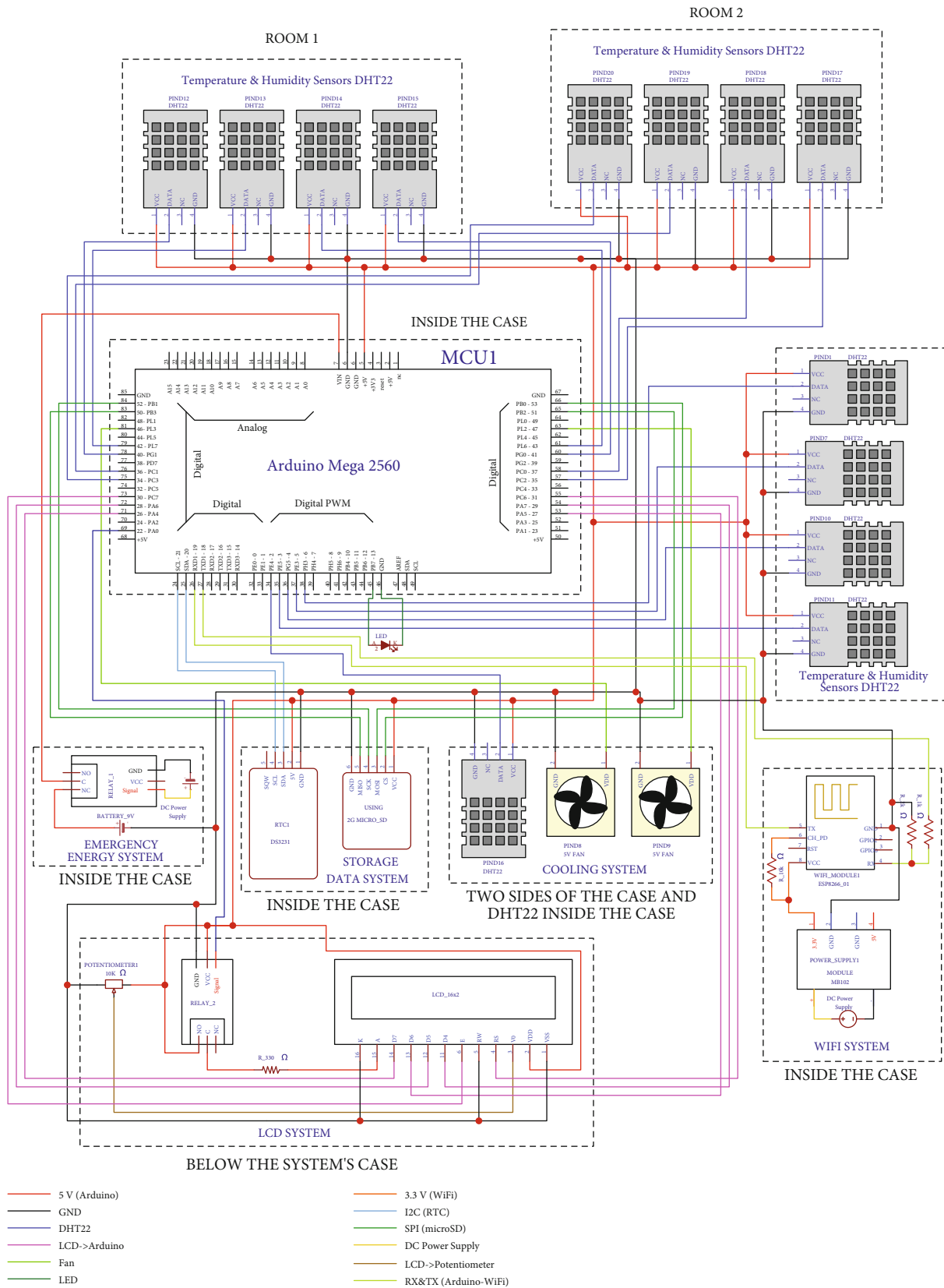
FIGURE 3: Schematic diagram of the remote sensor nodes based on the Arduino board used for the monitoring of environmental variables.

TABLE 2: Costs per device of the equipment that make up the IoT monitoring system.

Component	Company	Quantity	Cost per unit (USD)	Subtotal (USD)
Electrical energy monitoring system				
POWR2	ITEAD	1	$24.00	$24.00
Environmental monitoring system				
Arduino Mega	Arduino	1	$8.00	$8.00
DHT22	Adafruit	2*	$7.00	$14.00
DS3231	ElectroCrea	1	$1.50	$1.50
Micro SD card breakout	CATALEX	1	$1.30	$1.30
LCD HD44780	SparkFun	1	$3.00	$3.00
5 V DC fan	Steren	2	$5.00	$10.00
ESP8666	Haictronic	1	$5.00	$5.00
Relay RAS 0510	Sun Hold	2	$0.90	$1.80
9 V battery	—	1	$1.90	$1.90
Total				$46.50
Coordinator server node				
Raspberry Pi 2 B	Raspberry	1	$56.00	$56.00

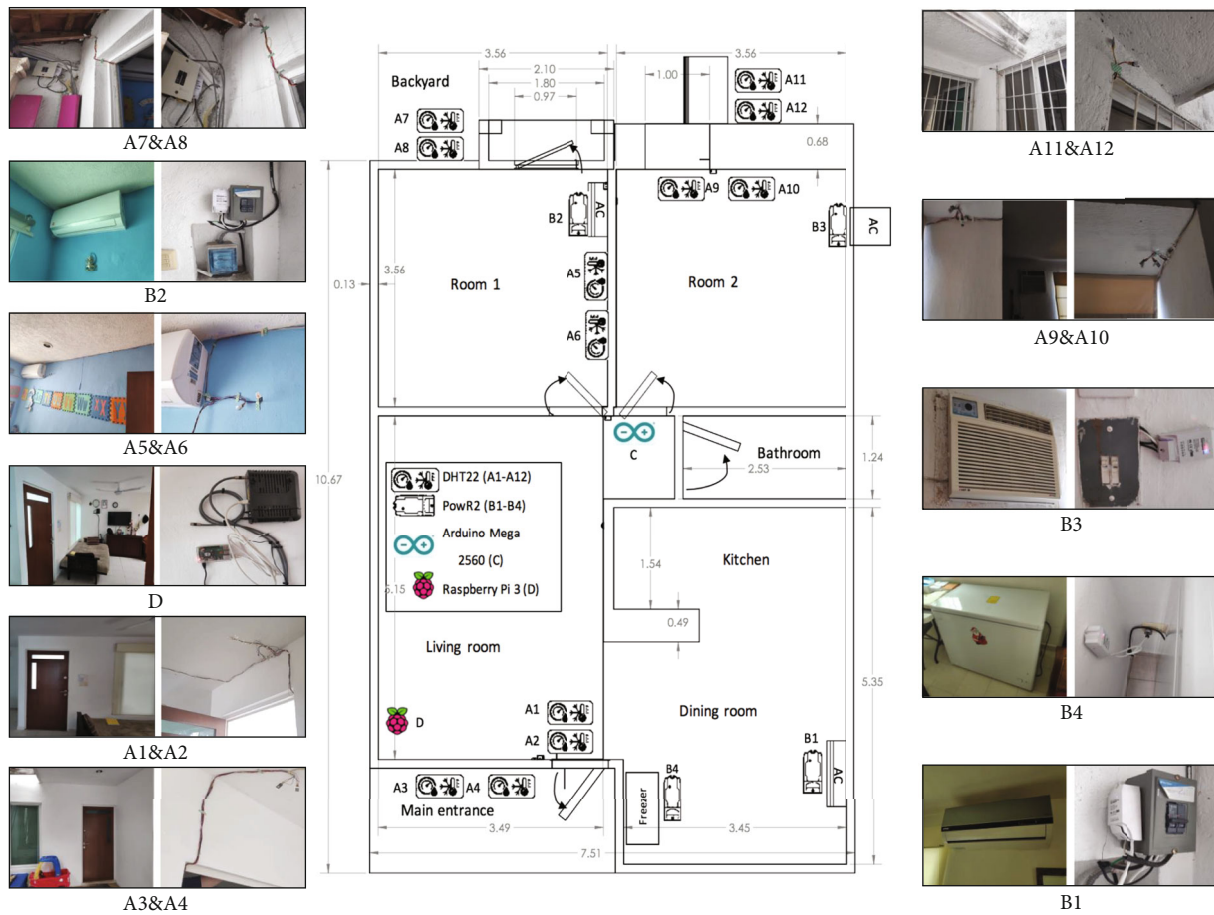*Each RSN is enabled to contain up to 14 sensors. The basic system is equipped with 2.

FIGURE 4: Schematic diagram of the house used for the case study with the detailed location of the devices that make up the monitoring system.

TABLE 3: Technical characteristics of the electrical devices monitored with the developed IoT system.

Device	Minisplit		Window A/C	Freezer
Model	EMPRC182-Y	RC2G-HW12C2	53WRC121A	DCF087A1WMX
Capacity	18,000 BTU/h	12,000 BTU/h	12,000 BTU/h	8.7 ft^3
Nominal voltage	220 V	220 V	220 V	115 V
Nominal current	7.56 A	5.78 A	11.6 A	1.4 A
Nominal power	1700 W	1250 W	1225 W	150 W
Location in house	Dining room	Room 1	Room 2	Dining room
Assigned RSN	B1	B2	B3	B4

was selected over conventional computers due to its low cost, high processing capacity, and open-source feasibility. Moreover, the Python programming language is installed by default, representing a significant advantage due to a large amount of documentation on the internet, making it easy to program and configure it and carry out the communication between different devices.

For this work, the Raspberry Pi is connected to the internet by the ethernet protocol. Because different platforms are used to capture the environmental and energy consumption variables, it is necessary to establish a reliable communication framework for the data capture, reception, storage, and actual-time display. The framework established in this work proposes the interaction of RSN with CSN through the MQTT communication protocol. The proposed framework was composed of the set of applications Mosquitto MQTT, Node-RED, InfluxDB, and ThingSpeak, whose function is described in detail below.

MQTT is an open network protocol developed as an extremely lightweight publish/subscribe messaging transport between devices. It is designed for connections with remote locations that require few computing resources or limited bandwidth [61]. This protocol was selected for being one of the primary standards for IoT applications due to its simplicity, lightness, low power consumption, low bandwidth, security, and robustness, ideal for devices that operate 24/7 and those powered by batteries. MQTT communication systems are made up of devices named clients that connect to a central server known as a broker. Each client can act as a publisher or subscriber and communicates with one or more brokers. Clients connect with the broker through "topic levels" made up of UTF-8 strings. Communication is carried out through specific topics, where the client-publisher sends the information to the broker, which is distributed to any client subscribed to said topic [62]. In the present work, the RSNs were assigned as clients while the broker was implemented in the Raspberry Pi board.
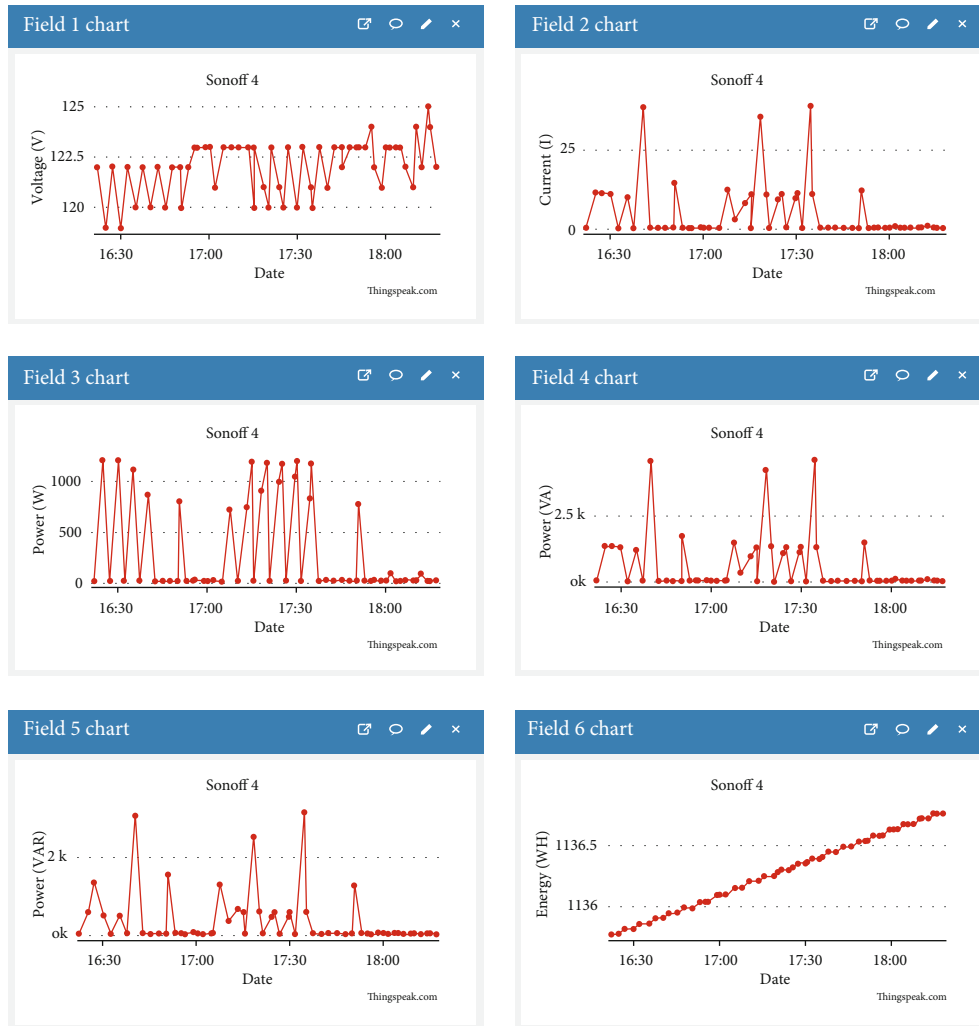
The Mosquitto API was used to enable the Raspberry Pi as a broker in the MQTT protocol. This software is commonly used to establish clients and/or brokers in MQTT communication systems [63]. It is characterized by its lightness and ability to handle various connections between different clients, being suitable for microprocessors with low computing power (compared to big servers). It is a free, open-source software implemented by hobbyists, academics, and commercial products with extensive documentation and

implementation examples on the web [42]. Within the context of the proposed monitoring system, Mosquitto API permits communication between the RSN and the RC, periodically sending the collected measurements to the broker (Raspberry Pi). The MQTT communication was established by assigning a specific topic to each RSN to identify the origin and type of the data sensed. During the system's operation, the broker sequentially requests the measurements of each RSN at specific intervals. On the other hand, it is configured to operate as a remote controller with the possibility of sending on/off commands to the RSN, although this function is only used in case the RSN needs to be restarted remotely.

Node-RED is a browser-based flow editor for wiring hardware devices, APIs, and online services [43]. It is an open-source tool focused on IoT applications, included by default in Raspbian operating systems. It is based on node.js, a JavaScript platform that uses an event handler to create data-intensive and lightweight real-time applications that can work seamlessly across many devices. Moreover, it is compatible with a large number of services and communication protocols, among which the MQTT stands out.

Node-RED was used as a bridge to link the Raspberry Pi to the cloud database. Monitored data sent by the MQTT protocol are entered into the software's process flow. The software defragments the UTF-8 code line (specific topic) and retrieves the relevant information from the measurements. The information is then recombined and assigned to the appropriate format for storage and/or sending to the cloud (InfluxDB and ThingSpeak use different storage formats). At the end of the process flow, the numerical measurement values are sent in separate data packets to their respective repositories for storage.

InfluxDB allows storage of the information received by the broker in a database; it is a free and open-source time-series database [44]. It was selected due to its multiple attributes, such as the ability to work under the Python language, compatibility with the IoT environment, and time-centered functions to query a data structure made up of measurements. InfluxDB is commonly used for storing large volumes of data-bound-to-time values, qualities that highlight it to be implemented in IoT systems over other databases such as MySQL, Oracle, or SQL Server designed mainly for inventory control [64]. Among its main applications are DevOps monitoring, metric sensor storage, and real-time measurement analysis.

Field 1 chart

Sonoff 4

Field 2 chart

Sonoff 4

Field 3 chart

Sonoff 4

Field 4 chart

Sonoff 4

Field 5 chart

Sonoff 4

Field 6 chart
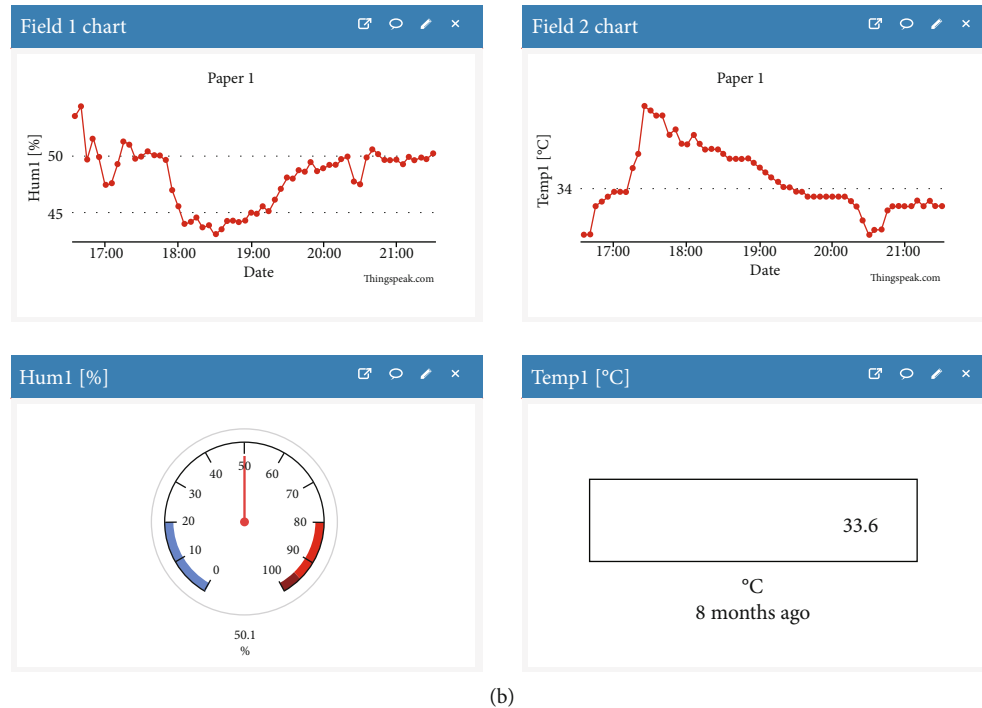
Sonoff 4

(a)

Figure 5: Continued.

FIGURE 5: Graphical user interface for web monitoring: (a) electrical energy monitoring; (b) indoor and outdoor surrounding environmental monitoring.

In the presented IoT system, this application is used to create a backup of the data uploaded to the cloud in case of saturated storage memory or not-available data transfer for some reason. The data packet coming from Note-RED is saved in a CSV file inside the SD memory incorporated in the Raspberry Pi board. Each stored character is equivalent to one byte; during each saving period, around 200 bytes of memory are used. Thus, in the case of continuous monitoring with measurements 12 times per hour (every 5 minutes), it is possible to store one year's worth of recorded data in less than 3 MB. Thus, a cheap 4 GB micro SD memory is more than enough to hold the Raspbian operating system and the database.

ThingSpeak is an IoT cloud service designed to enable users to load, store, and view sensor data from remote locations anywhere and/or at any time with a web browser [45]. It is a free-of-cost platform to perform the analysis and instant graphic representation of data shared by the linked measurement devices. It is also enabled to associate with the MATLAB application service and perform online analysis and processing.

ThingSpeak operates through the creation of a user account on its website (http://www.thingspeak.com). The user creates several input channels based on the type and amount of parameters monitored. Each channel is assigned with a unique API key to guarantee access to measurements of interest. API keys are linked to URLs to redirect the user to a data chart of the required information. Additionally, there is the possibility of configuring the channel as public (visible to anyone) or private (visible only to whoever has the user's access codes). This process was implemented to obtain remote access to the database stored in the cloud through mobile devices and computers (end users).

The data packet coming from Note-RED was uploaded to the cloud service. The monitored variables were classified with their respective collection location and assigned to a channel. The variables were visualized through a dynamic graph that displays the last 40 recorded data. Table 2 compiles the commercial information of devices that integrate the IoT monitoring system.

## 3. Case Study: Social Housing under Tropical Climate Conditions

3.1. Description of the Study Location and the Facility. The IoT monitoring system was set up in a single-story social housing located in the city of Merida, Yucatan, Mexico ($20°58'$N, $89°37'$W). This region is characterized by presenting a tropical climate with high temperatures throughout the year, reaching values above $36°$C in summer and relative humidity above 70% most of the time [65]. Combining these environmental characteristics necessitates the intensive use of indoor cooling equipment, making the city a suitable location for evaluating an energy/environmental monitoring system.

The house under study has an East-West orientation and a built area of 10.67 m $\times 7.51$ m $\times 2.60$ m (length, width, and height, respectively). The house is distributed into seven zones (living room, dining room, kitchen, bathroom, backyard, and two bedrooms) and is inhabited by six people (4 adults and 2 children). Further, for the interior air conditioning, the building has three types of equipment consisting
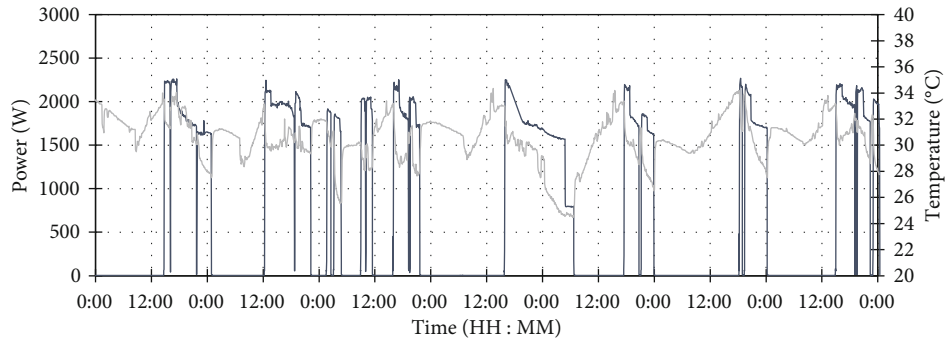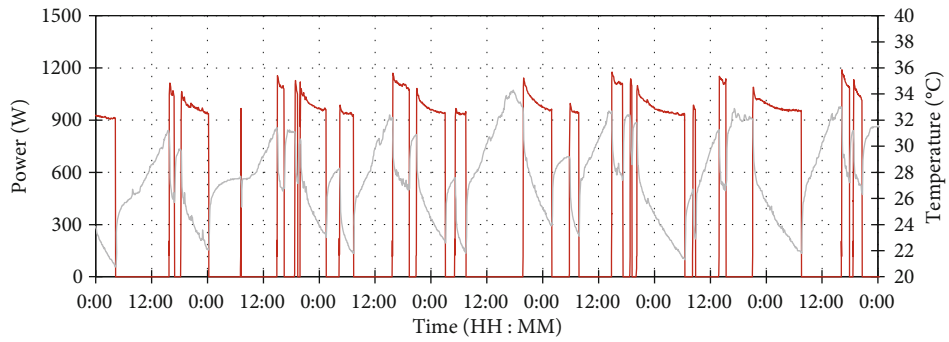
Power drawn by minisplit in living room

Temperature in living room

(a)
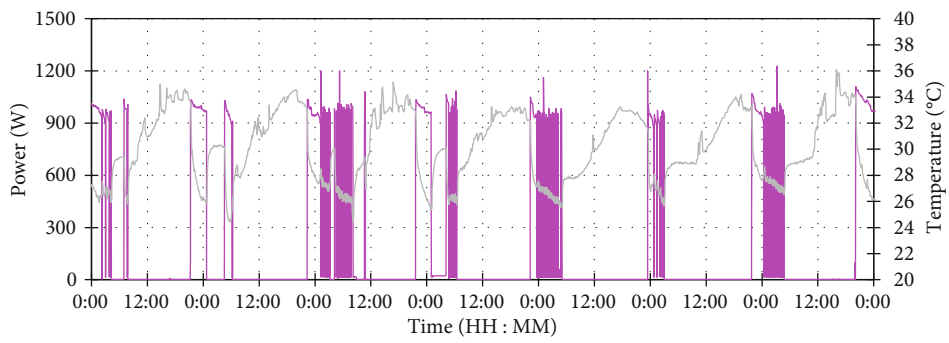
Power drawn by minisplit in room 1

Temperature in room 1

(b)

Power drawn by window AC in room 2
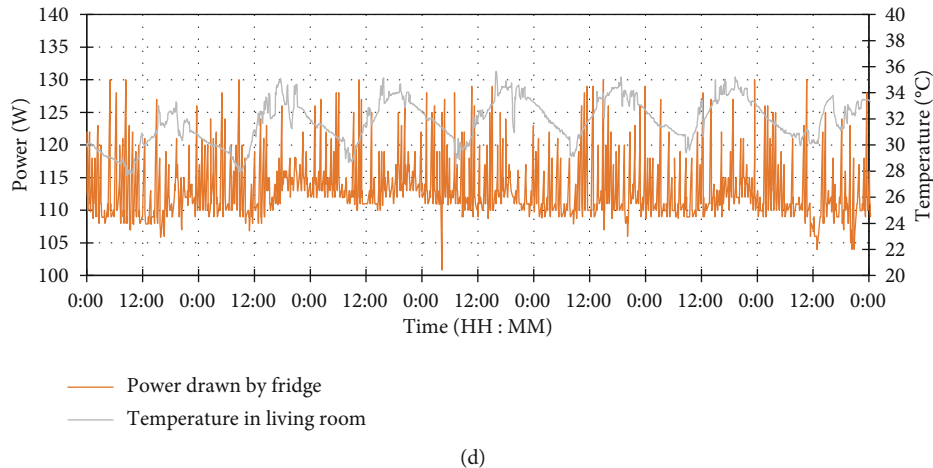
Temperature in room 2

(c)

FIGURE 6: Continued.

(d)

FIGURE 6: Seven-day measurement period corresponding to the energy consumption of the refrigeration equipment and the ambient temperature inside the enclosure: (a) living-dining room (minisplit); (b) room 1 (minisplit); (c) room 2 (window air conditioner); (d) dining room (freezer).

of two minisplit units and a window air conditioner, distributed in the areas with the highest occupancy in the house (dining room and bedrooms).

Figure 4 provides a sketch of the house with the mounting location of the measurement devices, classifying the environmental monitoring sensors as A1-A12 and the electrical energy monitoring sensors as B1-B4. The sensors for monitoring the air temperature and relative humidity were placed in pairs, covering specific measurement zones inside and around the house. The sensors are arranged as follows: A1 and A2 living room and dining room; A3 and A4 main entrance (exterior adjoining living room and dining room); A5 and A6 room 1; A7 and A8 backyard area adjoining room 1; A9 and A10 room 2; and A11 and A12 backyard area adjoining room 2. The placement of sensors in pairs was to validate the monitored information. It is important to emphasize that none of the sensors was directly exposed to irradiation.

Regarding the energy consumption monitors, these were installed in each of the space cooling equipment in the home; additionally, an electrical energy measurement device was incorporated in the refrigerator in the house. The measurement of the electrical demand of these devices is to study the patterns of energy consumption produced by changes in temperature and humidity. Table 3 presents the technical specifications of the monitored electrical devices and the assigned sensor code for each one.

3.2. GUI and Analysis. The results of implementing the IoT monitoring system are presented in two categories: (I) real-time visualization by GUI and (II) recording and analysis of measurements from the data stored into the Raspberry Pi's SD memory card by InfluxDB.

Figure 5 illustrates the graphical user interface, from ThingSpeak cloud service, for real-time visualization of the air temperature, relative humidity, and power monitored values. Figure 5 shows the web monitor for computer viewing. Through this interface, any authorized user can remotely observe the status of the sensed variables and each of the sensors installed. The data can be accessed in two ways; the first corresponds to a gauge (or label) that exhibits spot measurements (Figure 5), mainly used for displaying temperature and relative humidity values.

The graphs contained in Figures 6 and 7 depict a sample corresponding to a week of the interaction between electrical power consumed by the cooling devices and the indoor temperature (Figure 6) and relative humidity (Figure 7) at the house's interest zones. In the case of indoor temperature, Figures 6(a)–6(c) clearly show that air conditioners' turn on/turn off cycles operate under a thermal comfort threshold between 32°C and 24°C. Figure 8 also shows the performance of two different types of technology for space cooling. This can be seen in Figure 6(b) (minisplit) and Figure 6(c) (window air conditioner) in which, although they cool practically identical volumes, the cooling speed to reach the comfort temperature is slower in the case of the window air conditioner, finally impacting on electricity consumption and its associated cost. Thus, the information and knowledge gathered by the IoT system offer the possibility of establishing control strategies to reduce energy consumption based on the habits of occupants and the cooling devices installed in the home. The aforementioned is essential in social housings under these types of climate that apply an intensive use of these cooling systems.

Concerning indoor relative humidity, Figures 7(a)–7(c) show a reduction of this parameter during the operation of the air conditioners in the range of 50% to 30%. This is a desired characteristic in the tropical climate regions, compared to the dry climate, because excess humidity affects the heat index of the occupants. Nevertheless, the use of different cooling technologies impacts the form in that the humidity in the room is reduced. According to Figure 7(c), the use of window air conditioning stabilizes the relative humidity at 40%. Otherwise, with the minisplit (Figure 7 (b)), the relative humidity can fluctuate between 30% and
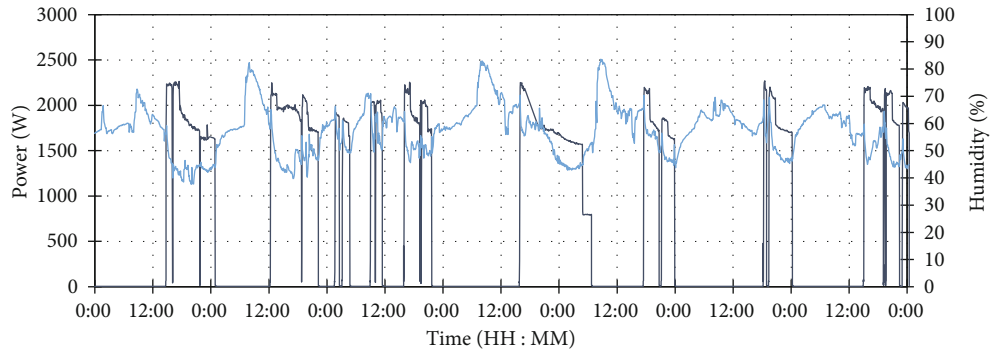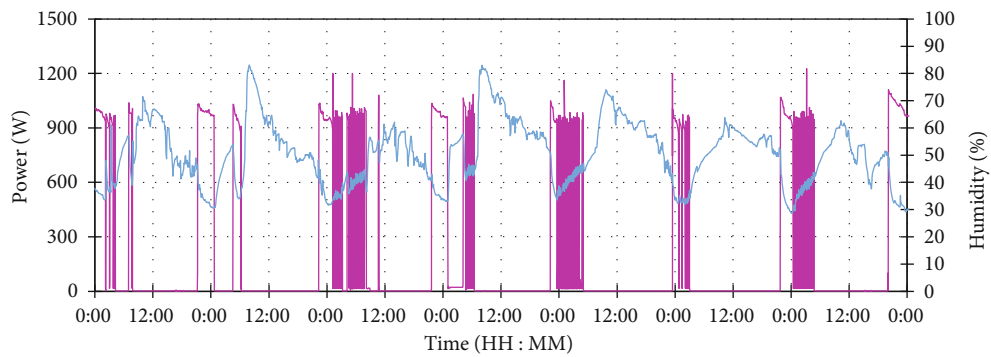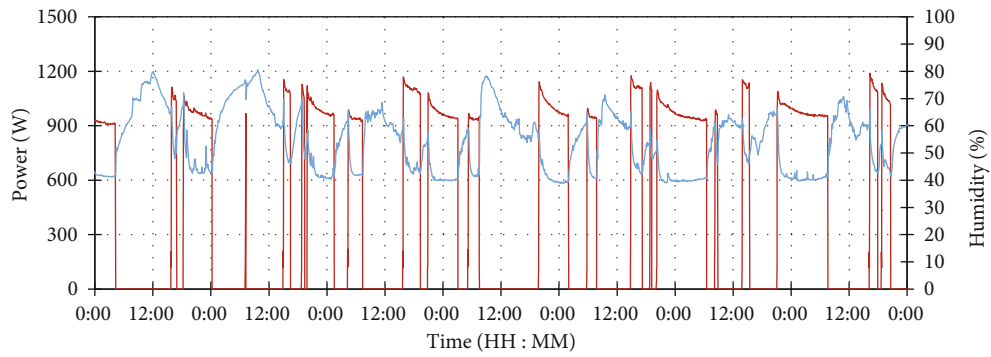
(a)
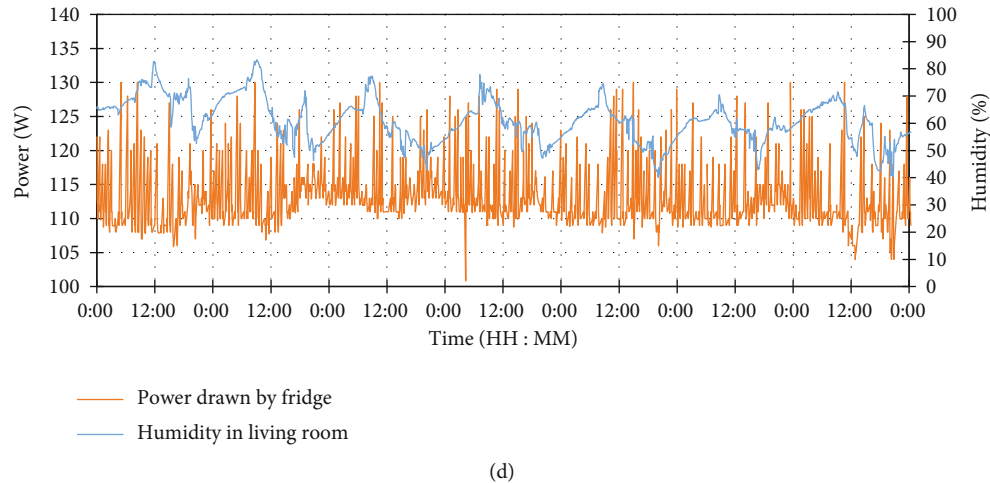

(b)

(c)

Figure 7: Continued.

(d)

FIGURE 7: Seven-day measurement period corresponding to the energy consumption of the refrigeration equipment and the relative humidity inside the enclosure: (a) living-dining room (minisplit); (b) room 1 (minisplit); (c) room 2 (window air conditioner); (d) dining room (freezer).

50%. As can be seen, these relative humidity levels are associated with electricity consumption since the window air conditioner, when operating continuously for much longer, manages to maintain this parameter. At the same time, the electrical power consumed by the minisplit encourages these variations.

Finally, in the case of the refrigerator, its energy consumption has a cyclical behavior during its operation. On the one hand, it can be seen that the compressor starting frequency can be related to the ambient temperature inside the room since it is relatively low when the room temperature begins to drop, while on the other hand, the relative humidity of the kitchen does not play a significant role in it as it is an isolated system.

Another benefit provided by the information storage scheme of the IoT system is the possibility of treating the collected samples. In the case of environmental variables and energy consumption of cooling devices, Figures 8 and 9 show the average behavior (after the data processing) of air temperature and relative humidity both inside and surrounding the monitored house. In this figure, the relative humidity levels reach their maximum in the morning (approximately between 9:00 am and 10:00 am) and begin to drop during the day due to solar radiation and wind speed factors. Observing the humidity in this period of the day, it is higher inside the house than outside. The above sets a guideline for integrating natural ventilation processes during specific periods of the day (for example, 10:00 am-6:00 pm) to take advantage of the low humidity outside.

Regarding the air temperature, the measurements reflect that during the nights, it is higher inside the house than outside, leading to an increase in the use of air conditioning to achieve comfort conditions. The above is a product of thermal storage phenomena since the houses of social interest in this region of Mexico are made of concrete, promoting heat retention. Implementing passive systems such as ventilated ceilings, facades, and thermal insulators would help reduce

temperatures and energy consumption within homes. Thus, the IoT system allows an analysis to implement new technologies and passive systems to improve thermal satisfaction and energy management in this type of housing.

Figure 9 illustrates the monthly average behavior throughout the day for both environmental variables. It is observed that during the sampling month, a trend is outlined throughout every day for the different hours, which can be summarized in an average behavior with specific standard deviations, both for relative humidity and air temperature. The preceding demonstrates how the data collected through the monitoring system can be disposed of and the feasibility of acquiring these variables in the climatic conditions considered.

Figure 8 shows the data processed for energy consumption by the monitored devices, where the power consumed is considered the parameter of most significant interest to analyze. This graph shows the average power consumed by each device throughout the day. According to the figure, it is possible to observe that effectively; the minisplit in the room is the electrical equipment that has the highest use and energy consumption throughout the day. Based on its performance curve, this device is mainly used after midday, when the ambient temperature rises and continues to operate until around midnight. From the quantitative perspective, the collected data, on average, the most significant energy requirement, is presented at the 19:50-hour power cycle, consuming power equivalent to 1,051.9 W. On the other hand, in this same graph, it is possible to see that the air conditioning equipment installed in rooms 1 and 2 are required during the night-morning hours (22:00 hrs-8:00 hrs), operating on average about 12 hours. The highest average consumption recorded by these devices was 606.2 W at 22:25 for the minisplit in room 1 and 542 W at 23:30 for the window AC in room 2. It is also possible to see that although both air conditioners cool space with practically the same volumes and use the same amount of time,
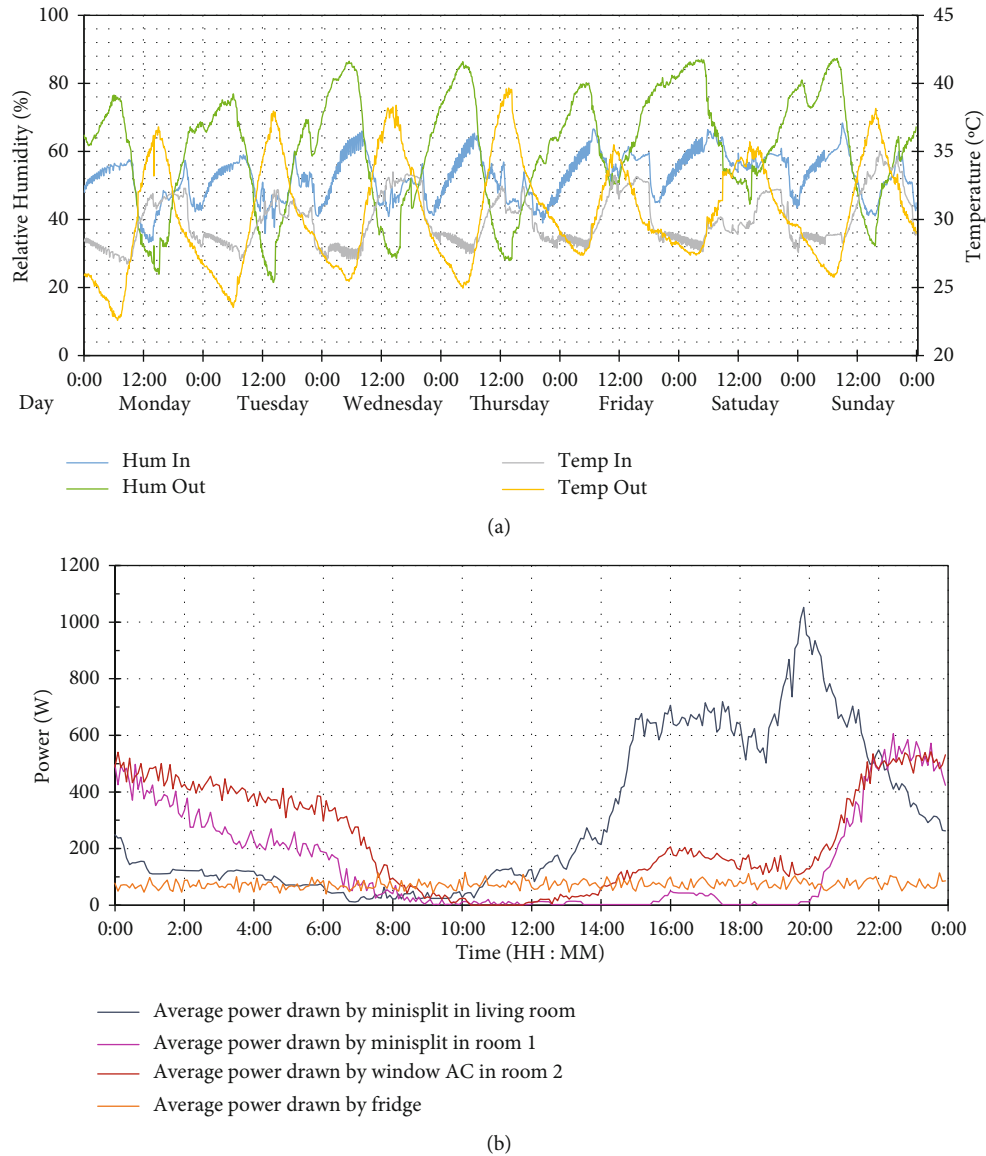
(a)

(b)

FIGURE 8: Average behavior curves of relative humidity and air temperature measured inside and outside the house under study.

the power consumed on average by the minisplit is less than that of the window AC (a difference of 81 W). This is because the AC unit in room 2 is an older window air conditioner, while the unit in room 1 is a modern element with more efficient technology. Finally, the consumption of the refrigerator was not affected by external environmental factors (temperature, humidity) because the volume to be cooled is an isolated system; there are only disturbances when the refrigerator is opened; however, these interactions are so short that the influence they have on consumption is minimal. Table 4 summarizes the most relevant information from electrical measurements. Thus, similar to environmental variables, the feasibility of data storage related to the characteristics of electrical equipment is viable. Finally, the contrast of the stored information shows that the system can be used as a tool for energy management based on the occupants' comfort.

4. Further Developments

From an environmental monitoring perspective, future planned developments involve incorporating new sensors (for example, proximity sensor, air quality sensor, wind speed sensor, light sensor, and binary for opening doors and windows) to improve quality in the measurement of comfort parameters. Concerning energy monitoring, work is being done to promote energy monitoring to an energy management process. This takes advantage of the benefits of using the Raspberry Pi microcomputer. Since the IoT monitoring system currently has a robust and effective software framework, we are working with the incorporation of artificial intelligence to establish management strategies based on the inhabitants' comfort habits.

Another important aspect is the economic one, with the knowledge that cost is the main barrier to integrating
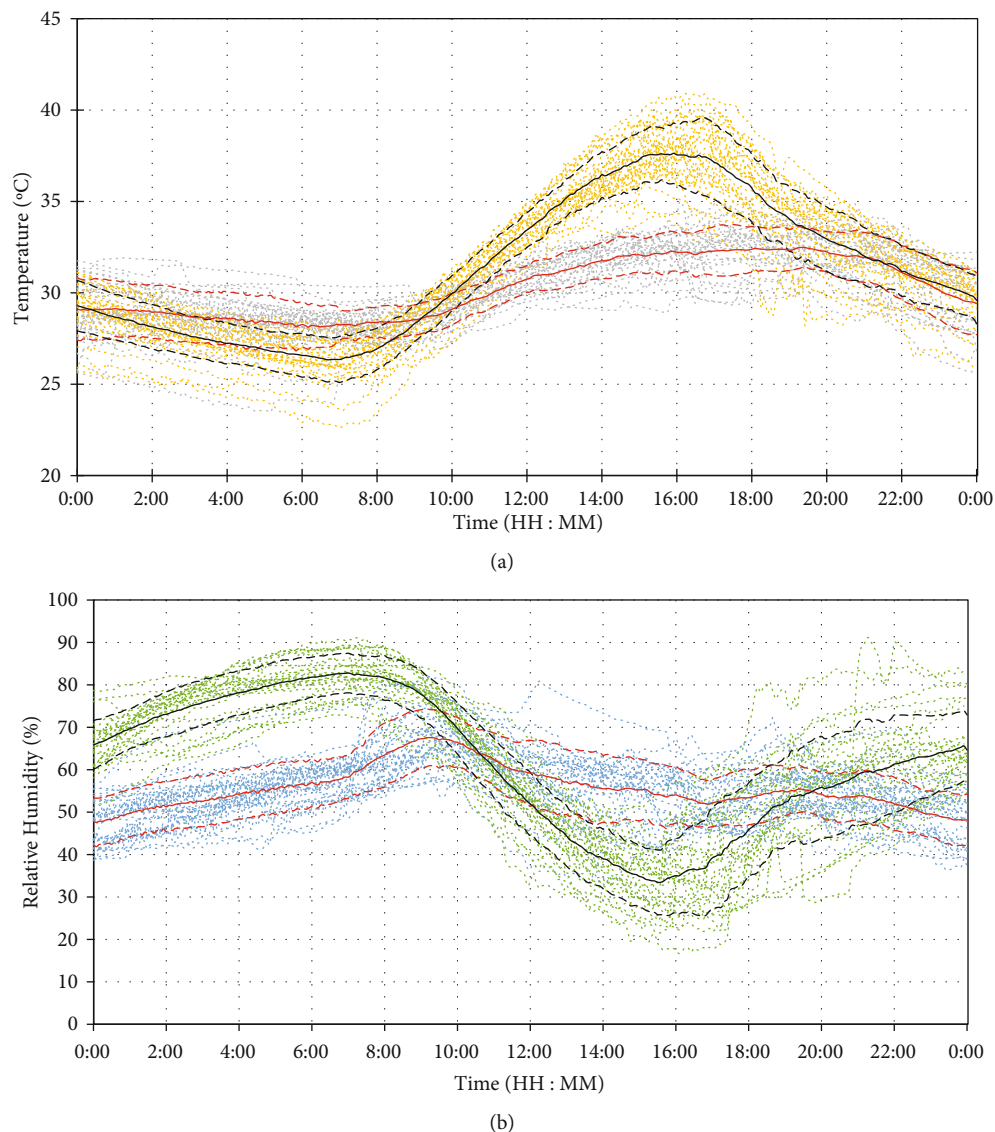
(a)

(b)

FIGURE 9: Description of the monthly behavior of the air temperature inside and outside the house under study.

TABLE 4: Quantitative measured values of the monitored cooling systems.

Device	Average power (W)	Room volume (m^3)	Average usage time (hrs)
Minisplit	308.5	108.17	10
Minisplit	183.0	38.02	12
Window AC	264.4	38.70	12
Freezer	73.4	0.25	24

monitoring and/or energy management equipment in social housings. Therefore, we continue working to reduce costs further to make the system more affordable.

Among the main strategies addressed for this purpose are replacing the current Arduino and Raspberry Pi cards by versions (with the same capacity) reduced in size and cost. This would favor a more compact and economical

development of environmental RSN as well as SCN with lower energy consumption.

5. Conclusions

A wireless IoT system for monitoring environmental and energy consumption parameters into homes was designed, developed, and evaluated. The system stores and provides real-time information regarding indoor air temperature and relative humidity and the electrical power consumed by cooling equipment in the residence. The system was built using affordable hardware platforms and diverse open license software. Communication between the measurement devices and the server that compose the IoT system was established using the MQTT protocol. The registered information is stored in databases and uploaded to the cloud to be available to users through web platforms.

The system was evaluated over in a postoccupancy single-story house under tropical climate conditions.

Operation tests indicate a successful performance of the protocols implemented for data sensing, communication, and cloud storage. The implementation of measurement devices with ESP8266 Wi-Fi modules and the use of the MQTT protocol were critical pieces for the wireless transmission of information safely and with low power consumption. The ease and viability of remote viewing from the graphical interfaces were demonstrated (different access points computers, smartphones, tablets, etc.), enabling as a feasible tool for monitoring at locations (with internet access) far from the measuring points. Moreover, the collected measurements show that the system can generate a solid database, indicating that it is suitable to operate for long sampling periods. On the other hand, the analysis of measurements demonstrates the feasibility of identifying the occupants' energy consumption patterns and their link with the search for comfortable environmental conditions, demonstrating the relevance of implementing these systems for energy management in regions with this type of climate.

The prototype presented can become an accessible alternative for monitoring and managing energy in homes based on comfort conditions. The design and low cost of the system make it suitable to be implemented in buildings with reduced space and low budgets, such as social housing and apartments. On the other hand, the communication, storage, and real-time visualization strategies implemented in this work have the ability to be adaptable and move to other scenarios where monitoring systems linked to the internet are required, such as applications in industrial processes, renewable energy, and transportation. Finally, the administration of information through low-cost microcomputers such as the Raspberry Pi offers the possibility of applying the system on a more advanced scale focused on developing domotic and inmotic systems that manage energy consumption through the knowledge of occupants' habits. So future works must be aimed at establishing control strategies and implementing artificial intelligence for learning the energy habits and comfort conditions and its implementation in real conditions for smart homes and buildings.

Data Availability

The measured data used to support the findings of this study are available from the corresponding authors upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

References

[1] O. May Tzuc, A. Livas-García, M. Jiménez Torres, E. Cruz May, L. M. López-Manrique, and A. Bassam, "Artificial intelligence techniques for modeling indoor building temperature under tropical climate using outdoor environmental monitoring," *Journal of Energy Engineering*, vol. 146, no. 2, article 04020004, 2020.

[2] IEA, *Energy Efficiency 2019*, International Energy Agency, Paris, France, 2020, https://www.iea.org/reports/energy-efficiency-2019.

[3] E. M. Saber, K. W. Tham, and H. Leibundgut, "A review of high temperature cooling systems in tropical buildings," *Building and Environment*, vol. 96, pp. 237–249, 2016.

[4] D. Rim, S. Schiavon, and W. W. Nazaroff, "Energy and cost associated with ventilating office buildings in a tropical climate," *PLoS One*, vol. 10, no. 3, pp. 1–14, 2015.

[5] IEA, *Annual energy outlook 2020*, International Energy Agency, 2020.

[6] A. H. Alavi, P. Jiao, W. G. Buttlar, and N. Lajnef, "Internet of Things-enabled smart cities: state-of-the-art and future trends," *Measurement*, vol. 129, pp. 589–606, 2018.

[7] M. S. Mekala and P. Viswanathan, "CLAY-MIST: IoT-cloud enabled CMM index for smart agriculture monitoring system," *Measurement*, vol. 134, pp. 236–244, 2019.

[8] Y. Shen, "Construction of a wireless sensing network system for leisure agriculture for cloud-based agricultural Internet of Things," *Journal of Sensors*, vol. 2021, Article ID 3021771, 11 pages, 2021.

[9] X. Wang, F. Chen, H. Ye et al., "Data transmission and access protection of community medical Internet of Things," *Journal of Sensors*, vol. 2017, Article ID 7862842, 14 pages, 2017.

[10] A. Ghosh, A. Raha, and A. Mukherjee, "Energy-efficient IoT-health monitoring system using approximate computing," *Internet of Things*, vol. 9, article 100166, 2020.

[11] A. R. Al-Ali, A. Al Nabulsi, S. Mukhopadhyay, M. S. Awal, S. Fernandes, and K. Ailabouni, "IoT-solar energy powered smart farm irrigation system," *Journal of Electronic Science and Technology*, vol. 17, no. 4, article 100017, 2019.

[12] R. I. S. Pereira, I. M. Dupont, P. C. M. Carvalho, and S. C. S. Jucá, "IoT embedded Linux system based on Raspberry Pi applied to real-time cloud monitoring of a decentralized photovoltaic plant," *Measurement*, vol. 114, pp. 286–297, 2018.

[13] B. Srbinovski, G. Conte, A. P. Morrison, P. Leahy, and E. Popovici, "ECO: an IoT platform for wireless data collection, energy control and optimization of a miniaturized wind turbine cluster: power analysis and battery life estimation of IoT platform," in *2017 IEEE International Conference on Industrial Technology (ICIT)*, pp. 412–417, Toronto, ON, Canada, 2017.

[14] N. P. G. Bhavani, P. Vaishnavi, and K. Sujatha, "Off-shore wind power as a pillar of energy transmission using IOT (OSWPETIOT)," in *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, pp. 2494–2499, Chennai, India, 2017.

[15] D. Vasicek, J. Jalowiczor, L. Sevcik, and M. Voznak *2018 26th Telecommunications Forum (TELFOR)*, pp. 1–4, Belgrade, Serbia, 2018.

[16] M. Collotta and G. Pau, "Bluetooth for Internet of Things: a fuzzy approach to improve power management in smart homes," *Computers and Electrical Engineering*, vol. 44, pp. 137–152, 2015.

[17] M. Karami, G. V. McMorrow, and L. Wang, "Continuous monitoring of indoor environmental quality using an Arduino-based data acquisition system," *Journal of Building Engineering*, vol. 19, pp. 412–419, 2018.

[18] A. J. Lewis, M. Campbell, and P. Stavroulakis, "Performance evaluation of a cheap, open source, digital environmental monitor based on the Raspberry Pi," *Measurement*, vol. 87, pp. 228–235, 2016.

[19] H. E. Silva, G. B. A. Coelho, and F. M. A. Henriques, "Climate monitoring in World Heritage List buildings with low-cost data loggers: the case of the Jeronimos Monastery in Lisbon (Portugal)," *Journal of Building Engineering*, vol. 28, article 101029, 2020.

[20] A. Carre and T. Williamson, "Design and validation of a low cost indoor environment quality data logger," *Energy and Buildings*, vol. 158, pp. 1751–1761, 2018.

[21] S. Ferdoush and X. Li, "Wireless sensor network system design using Raspberry Pi and Arduino for environmental monitoring applications," *Procedia Computer Science*, vol. 34, pp. 103–110, 2014.

[22] D. Meana-Llorián, C. González García, B. C. Pelayo G-Bustelo, J. M. Cueva Lovelle, and N. Garcia-Fernandez, "IoF-Clime: the fuzzy logic and the Internet of Things to control indoor temperature regarding the outdoor ambient conditions," *Future Generation Computer Systems*, vol. 76, pp. 275–284, 2017.

[23] V. Vujović and M. Maksimović, "Raspberry Pi as a sensor web node for home automation," *Computers and Electrical Engineering*, vol. 44, pp. 153–171, 2015.

[24] W. T. Sung and S. J. Hsiao, "The application of thermal comfort control based on smart house system of IoT," *Measurement*, vol. 149, article 106997, 2020.

[25] A. Martín-Garín, J. A. Millán-García, A. Baïri, J. Millán-Medel, and J. M. Sala-Lizarraga, "Environmental monitoring system based on an open source platform and the Internet of Things for a building energy retrofit," *Automation in Construction*, vol. 87, pp. 201–214, 2018.

[26] P. Kalia and M. A. Ansari, "IOT based air quality and particulate matter concentration monitoring system," *Materials Today: Proceedings*, vol. 32, pp. 468–475, 2020.

[27] H. Ponce and S. Gutiérrez, "An indoor predicting climate conditions approach using Internet-of-Things and artificial hydrocarbon networks," *Measurement*, vol. 135, pp. 170–179, 2019.

[28] V. R. Shinde, P. P. Tasgaonkar, and R. D. Garg, "Environment monitoring system through Internet of Things (IOT)," in *2018 International Conference on Information , Communication, Engineering and Technology (ICICET)*, pp. 1–4, Pune, India, 2018.

[29] V. S. Arumuga Perumal, K. Baskaran, and S. K. Rai, "Implementation of effective and low-cost building monitoring system(BMS) using Raspberry PI," *Energy Procedia*, vol. 143, pp. 179–185, 2017.

[30] F. Abate, M. Carratù, C. Liguori, and V. Paciello, "A low cost smart power meter for IoT," *Measurement*, vol. 136, pp. 59–66, 2019.

[31] M. O. Agyeman, Z. Al-Waisi, and I. Hoxha, "Design and implementation of an IoT-based energy monitoring system for managing smart homes," in *2019 Fourth International Conference on Fog and Mobile Edge Computing (FMEC)*, pp. 253–258, Rome, Italy, 2019.

[32] K. Matsui, Y. Yamagata, and S. Kawakubo, "Real-time sensing in residential area using IoT technology for finding usage patterns to suggest action plan to conserve energy," *Energy Procedia*, vol. 158, pp. 6438–6445, 2019.

[33] S. Shapsough, M. Takrouri, R. Dhaouadi, and I. Zualkernan, "An IoT-based remote IV tracing system for analysis of city-wide solar power facilities," *Sustainable Cities and Society*, vol. 57, article 102041, 2020.

[34] M. D. Mudaliar and N. Sivakumar, "IoT based real time energy monitoring system using Raspberry Pi," *Internet of Things*, vol. 12, article 100292, 2020.

[35] R. A. Nadafa, S. M. Hatturea, V. M. Bonala, and S. P. Naikb, "Home security against human intrusion using Raspberry Pi," *Procedia Computer Science*, vol. 167, pp. 1811–1820, 2020.

[36] P. Bora, P. Kanakaraja, B. Chiranjeevi, M. J. S. Sai, and A. Jeswanth, "Smart real time health monitoring system using Arduino and Raspberry Pi," *Materials Today: Proceedings*, vol. 46, pp. 3855–3859, 2021.

[37] J. B. Samson, K. Alwin Fredrick, M. N. Sathiya, R. Catherine Joy, W. J. Wesley, and S. Stanley Samuel, "Smart energy monitoring using Raspberry Pi," in *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 845–849, Erode, India, 2019.

[38] A. Hafid, S. Benouar, M. Kedir-Talha, F. Abtahi, M. Attari, and F. Seoane, "Full impedance cardiography measurement device using Raspberry PI3 and system-on-chip biomedical instrumentation solutions," *IEEE Journal of Biomedical and Health Informatics*, vol. 22, no. 6, pp. 1883–1894, 2018.

[39] INFONAVIT, *Acerca de nosotros|Infonavit*, Instituto del Fondo Nacional de la Vivienda para los Trabajadores, 2020, https://portalmx.infonavit.org.mx/wps/portal/infonavit.web/el-instituto/el-infonavit/acerca-de-nosotros/!ut/p/z0/04_Sj9CPykssy0xPLMnMz0vMAfIjo8zizdwNDDycTQz9L-Hy8TQ0CDQK83Q28DAyDPQz1C7IdFQE0QU1x/.

[40] INEGI, *Primera encuesta nacional sobre consumo de energéticos en viviendas particulares (ENCEVI)*, Instituto Nacional de Estadística y Geografía, 2018.

[41] J. Newmarch, "Raspberry Pi," in *Linux Sound Program*, pp. 537–545, Apress, Berkeley, CA, 2017.

[42] R. A. Light, "Mosquitto: server and client implementation of the MQTT protocol," *Journal of Open Source Software*, vol. 2, no. 13, p. 265, 2017.

[43] JS Foundation, "Documentation: Node-RED," April 2020, https://nodered.org/docs/.

[44] InfluxData, "InfluxDB 1.8 documentation |InfluxData documentation," April 2020, http://influxdata.com/influxdb/v1.8/.

[45] S. Pasha, "ThingSpeak based sensing and monitoring system for IoT with Matlab analysis," *International Journal of New Technology and Research (IJNTR)*, vol. 2, pp. 19–23, 2016.

[46] SONOFF Official, "SONOFF POWR2-WiFi smart light switch to monitor home energy usage," April 2020, https://sonoff.tech/product/wifi-diy-smart-switches/powr2.

[47] ITEAD Wiki, "Sonoff Pow - ITEAD Wiki," 2019, https://www.itead.cc/wiki/Sonoff_Pow.

[48] "eWeLink, eWeLink-your smart home center," April 2020, https://www.ewelink.cc/en/.

[49] X. Pérez, "GitHub - xoseperez/espurna: home automation firmware for ESP8266-based devices," April 2020, https://github.com/xoseperez/espurna.

[50] D. Minchev and A. Dimitrov, "Home automation system based on ESP8266," in *2018 20th International Symposium on Electrical Apparatus and Technologies (SIELA)*, pp. 1–4, Bourgas, Bulgaria, 2018.

[51] Haitronic, "HR0128 NodeMCU board ESP8266 WiFi module CP2102 ESP-12E Lua WiFi," May 2020, http://www.haitronic.cn/index.php?route=product/product&product_id=214.

[52] P. P. Ray and N. Thapa, "A systematic review on real-time automated measurement of IV fluid level: status and challenges," *Measurement*, vol. 129, pp. 343–348, 2018.

[53] Arduino, "Arduino Mega 2560 Rev3 | Arduino Official Store," May 2020, https://store.arduino.cc/usa/mega-2560-r3.

[54] T. Liu, *Digital humidity and temperature sensor*, Adfruit, 2016, https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf.

[55] ISO 7726, *Ergonomics of the thermal environment — instruments for measuring physical quantities*, International Organization for Standardization, Switzerland, 2001.

[56] EN 15758, "Conservation of cultural property. Procedures and instruments for measuring temperatures of the air and the surfaces of objects," 2010, https://shop.bsigroup.com/ProductDetail?pid=000000000030173521.

[57] ANSI/ASHRAE, *ANSI/ASHRAE standard 55-2017: thermal environmental conditions for human occupancy*, American Society of Heating, Refrigerating and Air-Conditioning Engineers, 2017.

[58] V. H. Hernández Gómez and D. M. Gálvez, "Analytical model for double skin roofs," *Applied Thermal Engineering*, vol. 60, no. 1-2, pp. 218–224, 2013.

[59] M. A. Vishnu, "Interfacing Catalex microSD card with Arduino - Vishnu M Aiea," 2017, https://www.vishnumaiea.in/projects/hardware/interfacing-catalex-micro-sd-card-module.

[60] Electrocrea, "DS3231 módulo RTC + batería— ElectroCrea," May 2020, https://electrocrea.com/products/modulo-rtc-ds3231.

[61] MQTT community, *FAQ - frequently asked questions*, MQTT, 2020, April 2020, http://Mqtt.org.

[62] MQTT Community, "Home MQTT GitHub," 2019, https://github.com/mqtt/mqtt.github.io/wiki.

[63] Eclipse Foundation, "Eclipse Mosquitto," April 2020, https://mosquitto.org/.

[64] D. Simmons, "What are some interesting uses of InfluxDB? - Quora," 2018, https://www.quora.com/What-are-some-interesting-uses-of-InfluxDB.

[65] Gobierno de México, "Centro Hidrometeorológico Yucatán - Mérida," May 2020, https://smn.conagua.gob.mx/es/centro-hidrometeorologico-yucatan-merida.