

An efficient approach to monitoring environmental conditions using a wireless sensor network and NodeMCU

X. Bajrami, I. Murturi

In recent years, wireless sensor networks (WSNs) have become established as an effective way of interconnecting sensor devices with simultaneous communication and data processing. Since it allows low-cost and low-power sensor devices, the use of WSNs has increased in different application areas, such as environmental monitoring, smart cities and crowdsourcing. This paper presents the functional design and implementation of a system for monitoring environmental conditions through using a WSN and NodeMCU. The system is developed for web- and mobile-based applications where preprocessed data are subsequently sent to the cloud via a web API, where comprehensive data analytics can be performed to gain better insight into the overall environmental conditions and help with improvements. The user can monitor measured data anywhere via the Internet by using the web and mobile applications. If the data measured by a sensor node exceeds the configured value range, the web application sends a notification to the user to improve the environmental conditions.

Keywords: WSN; sensor nodes; monitoring environment; IoT

Ein effizienter Ansatz zur Überwachung von Umweltbedingungen mit einem drahtlosen Sensornetz und NodeMCU.

In den letzten Jahren haben sich drahtlose Sensornetze (WSNs) als wirkungsvoll erwiesen, wenn es darum geht, Sensoreinrichtungen an eine parallele Datenübertragung zu koppeln. Der Einsatz von WSNs hat dabei in unterschiedlichsten Bereichen stark zugenommen, da diese sehr gut mit den günstigen und energiesparenden Sensoreinrichtungen zusammenspielen. Beispiele sind etwa die Überwachung von Umgebungsbedingungen, der Einsatz in Smart Cities oder beim Crowdsourcing. Der vorliegende Beitrag erläutert Funktion und Umsetzung eines Systems zur Überwachung von Umgebungsbedingungen unter Einsatz eines drahtlosen Sensornetzes und auf Basis von NodeMCU. Das System ist für Web- und mobile Anwendungen konzipiert, bei denen die zu übertragenden Daten direkt via Web API an eine Cloud gesendet werden. Dort können in weiterer Folge umfassende Datenanalysen durchgeführt werden, um einen besseren Einblick in die generellen Umgebungsbedingungen zu erhalten bzw. um diese in weiterer Folge zu beeinflussen. Der Anwender kann von überall aus via Internet – mittels Web- oder mobiler Anwendung – auf die Daten zugreifen. So sendet das System auch Benachrichtigungen, wenn die vom Sensor erfassten Daten bestimmte vorkonfigurierte Werte übersteigen, damit der Anwender entsprechend eingreifen kann.

Schlüsselwörter: drahtloses Sensornetz; Sensorknoten; Überwachungsumgebung; Internet der Dinge

Received June 30, 2017, accepted January 10, 2018, published online April 25, 2018
© The Author(s) 2018



1. Introduction

Research on wireless sensor networks (WSNs) began sometime in the 1980s, but only since 2001 have WSNs generated increased interest from industrial and research perspectives. This is owing to the current availability of inexpensive miniature components with low power requirements, such as processors, radios, and sensors, that are often integrated into a single chip (SoC: system on one chip) [1].

Today WSN technology has become an integral part of any developing country as it is being used nowadays as the primary monitoring system in various applications. WSNs eliminate the hazards associated with wiring systems and make data measurement and monitoring processes much easier and more cost-effective [2].

Today our homes are becoming more and more digitised, and while away from their homes people require more monitoring and control of electronic household items or of the surrounding environment. Using WSN technology for data transmission, we can automatically measure different environments and access the measurements remotely from the Internet to obtain real-time data from sen-

sor nodes; thus, for a low cost, we can access stored data that are measured and saved on different environments.

In Kosovo, not much emphasis is placed on environmental monitoring because there are no laws about conservation or continuous monitoring of temperature and air humidity for state institutions, such as the inspectorate. Environmental monitoring has a direct impact on the quality of life and health of people, especially children and those of old age where the temperature should be 20–25 °C and the air humidity has to be around 50% regardless of the time of year. In industries and factories, for the machines and technological equipment to function properly the temperature and air humidity has to be appropriate.

In this paper, we monitored the temperature and humidity in the analytical laboratory at QKMF (Center of Family Medicine) Prizren.

Bajrami, Xhevahir, Department of Mechatronics, University of Prishtina, Pristina, Kosovo, Albania; **Murturi, Ilir**, Distributed Systems Group, Vienna University of Technology, Argentinierstraße 8/194-02, 1040 Vienna, Austria
(E-mail: imurturi@dsg.tuwien.ac.at)

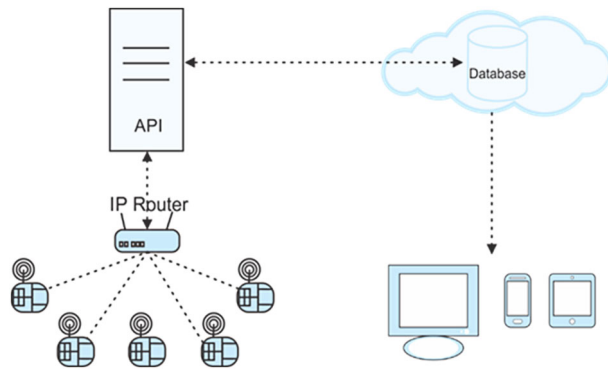


Fig. 1. The wireless sensor network system developed in this paper

The equipment that is used for various tests in the QKMF laboratory works if the temperature is in the range of 15–25 °C and air humidity 40–70%; if the temperature and the humidity are outside these boundaries it influences the results of the analysis. Monitoring of the temperature and humidity of the air was done also in the server room of Telcomm Company in Prizren. For the life expectancy of the servers, the temperature has to be in the range of 18–27 °C. Unfortunately, today the systems for environment monitoring are very expensive and have high energy consumption. In this paper, we used a NodeMCU device as the basis, which has the module ESP8266 integrated to communicate with the Internet while provides little power supplied as processors.

The preprocessed data transmitted by WSN sensor nodes are subsequently sent to the cloud via a web API where comprehensive real-time data analytics can be performed to gain better insight into the environmental overall conditions. Finally, the WSN sensor nodes perform only basic, lightweight analytics while the main data processing is carried out in the cloud, which means that the power consumption is significantly lowered.




2. Methodology

A WSN can be described as a network of sensor nodes that feel (understand) and control the environment, cooperating jointly with people or computers to control the surrounding environment [3]. The sensor node is one of the main parts of a WSN. Sensor nodes in the network are transmitters as well as receivers; the hardware of the sensor nodes in this paper includes the following parts: a power module with polymer batteries, a microcontroller NodeMCU including Wi-Fi module ESP8266 and the sensors DHT11, DS18B20 and YL-69. After the creation of the sensor nodes and the uploading of the code on the development board, the sensor nodes send data via a linking bridge (Wi-Fi router) in the cloud, where the database is located, through a developed web API.

The web API sends a warning email about the improvement of environmental conditions if the data is out of the configuration range.

The web application is suitable for mobile devices to monitor the areas from anywhere via the web while having the possibility of monitoring data weekly or daily, as presented in Fig. 1. The data transferred to the cloud can be used for more complex analysis that will predict and prioritise more precisely the environmental conditions, optimise the time needed to reach the mentioned locations and provide timely information about the environmental conditions.

Table 1. Measured parameters and sensors used

Measured parameters	Sensors	Figures
Temperature and air humidity	DHT11	
Soil temperature	DS18B20	
Soil humidity	YL-69	

3. Wireless sensor network node design

3.1 Sensors

During this study, in order to test the system, sensors are connected to the NodeMCU board to measure the temperature and air humidity, temperature and soil moisture of different objects. The sensors used for the measurement of these parameters are presented in Table 1.

3.1.1 Temperature and air humidity sensor (DHT11)

For the measurement of air temperature and humidity, a DHT11 sensor is used, which is produced by AOSONG. These sensors are small, economical, user-friendly and have low energy consumption. Inside they have an NTC component (negative temperature coefficient) for temperature measurement and a capacitive sensor to measure air humidity. They have four pins, one of which is enabled for data transmission is at a distance of up to 100 m. The DHT11 sensor is good for 20–80% humidity readings with 5% accuracy and for 0–50 °C temperature readings ± 2 °C accuracy. A sampling rate of up to 1 Hz (once every second) is possible.

3.1.2 Temperature sensor (DS18B20)

To measure the temperature of soil and water a DS18B20 sensor is used, which is produced by DALLAS. The usable temperature range is between -55 and 125 °C with 3.0–5.5 V power/data. It can convert the temperature reading into 12-bit digital data in 750 ms (max). The sensor is water resistant and it has a stainless steel head, which makes it suitable for any environment, be it wet or rough. It is accurate from -10 to $+85$ °C.

3.1.3 Humidity sensor (YL-69)

To measure the humidity of the soil a YL-69 sensor is used. The sensor is divided into two parts, the electronic part on the left and the probe consisting of two electrodes for moisture sensitivity on the right. The sensor has a potentiometer for adjusting the sensitivity so that if the measured value falls below the calibrated value, the digital output sends a signal. The sensor has four pins where two enable data transmission, one in digital and one in analogue.

3.2 NodeMCU

The NodeMCU is a microcontroller with integrated Wi-Fi, which means that there is no need for an additional Wi-Fi chipset. The design of the SoC allows communication through the GPIOs by connecting to the Internet and transmitting data over the Internet. This is a perfect connection for the Internet of Things (IoT). It has a price of about \$2.50, with a physical size of $49 \times 24.5 \times 13$ mm and consumes 0.00026–0.56 W of power. This is the best hardware around in terms of cost and this chip is the future of the IoT [4].

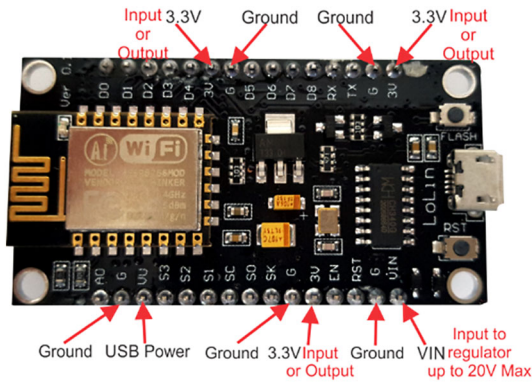


Fig. 2. Illustration of the pins for the input and output of the power supply

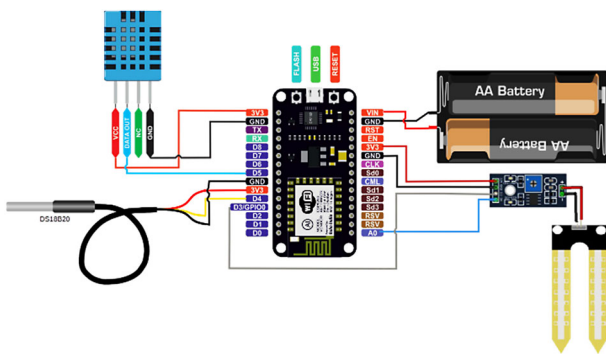


Fig. 3. Connection of the NodeMCU board with the DHT11, DS18B20 and YL-69 sensor and the batteries

Today, many retailers offer ESP8266 Breakout boards to facilitate our work. The NodeMCU is an easily usable board and it has a variety of pins. It has a USB connection port to connect to the computer.

3.2.1 NodeMCU power supply

To supply the NodeMCU with electricity, 3.7 V lithium polymer batteries with a capacity of 2500 mAh are used, which are linked to the NodeMCU through the VIN and ground pins. The positive pole of the battery (+) connects to the pin of the VIN and the negative pole of the battery (-) to the ground pin. The NodeMCU has a power output of 3.3 V to supply other devices with power; in this case, we supplied the sensors with 3.3 V of power (Fig. 2).

3.2.2 Connection of NodeMCU with the sensors used in this paper

Sensors enable the transmission of data in digital or analogue form, but some sensors have pins for both analogue and digital transmission. As an example, a DHT11 sensor is connected to the NodeMCU. The VCC and GND sensor pins are connected to the 3V3 and GND pins, respectively, whereas the data out pin is connected to the D5 pin. In this case, the measured sensor data is sent via the data out pin to the D5 pin on the NodeMCU board in digital form. The same connection applies to other sensors used in this project; however, the sensors with an analogue output connect to the NodeMCU via an A0 pin. The connection of the sensors used in this project with the NodeMCU board and batteries is presented in Fig. 3.

The setup is not constantly in the ON state; data measurements are taken every 15 min, after which it goes into sleep mode, which means that the power consumption is significantly lowered.



Fig. 4. The built sensor node

3.3 Coding of NodeMCU through Arduino IDE

After building the sensor nodes, the code, which enables the storage of the data measured by the sensors in the database, was written and uploaded to of NodeMCU board using Arduino IDE. Since 2015, Arduino IDE has allowed programming of NodeMCU boards. The Arduino language is merely a set of C/C++ functions that can be called from our code. The developed program and the changes made in the program can be easily uploaded into the NodeMCU board through a USB cable.

3.4 Creating the sensor nodes

The battery and sensors together with the NodeMCU board are placed in a box, as presented in Fig. 4. The code from Arduino IDE is uploaded to the sensor nodes to access the wireless network and to send the measured data to the web API program, which is developed in PHP language and saves the data to the database. The user can access the data that is stored in the database from anywhere via the developed web and mobile applications.

4. Web and mobile applications

4.1 "WSN Monitoring" web application

So the users can monitor the environments via the Internet, a web application called "WSN Monitoring" has been developed. WSN Monitoring is developed on PHP technology, which is dedicated to desktop and mobile browsers. WSN Monitoring has two type of users: system administrators and monitoring users. System administrators are responsible for the registration of new users, editing, deleting, managing, generating reports etc. In other words, system administrators have permission to read and write actions into the system. WSN Monitoring stores data in the cloud-based database. WSN Monitoring is a very convenient application that can be used to monitor different areas regardless of the user's location. Monitoring users are device users that will monitor data from the sensors installed at various locations. The interface developed for monitoring users also supports mobile and tablet browsers. Once the monitoring user is in a session gathering data from the location, it will remain open until the user logs out. In the case of any undesirable situation, such as increased temperature, WSN Monitoring will inform the user in real-time through browser notifications. The development of the interfaces for administrators and monitoring users was accomplished with the use of jQuery UI plugin and Bootstrap cascade style sheets. The application is suitable for smartphones or any smart device, and it is developed to be easy to use so that if the user instructs it, a module would know to use all the modules.

The application must first be configured: an email address has to be entered, as well as the minimum and maximum range for each

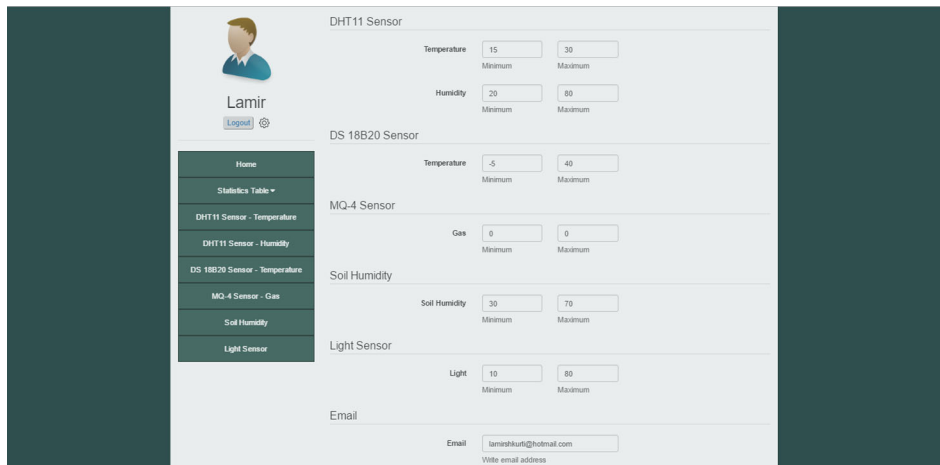


Fig. 5. Application configuration

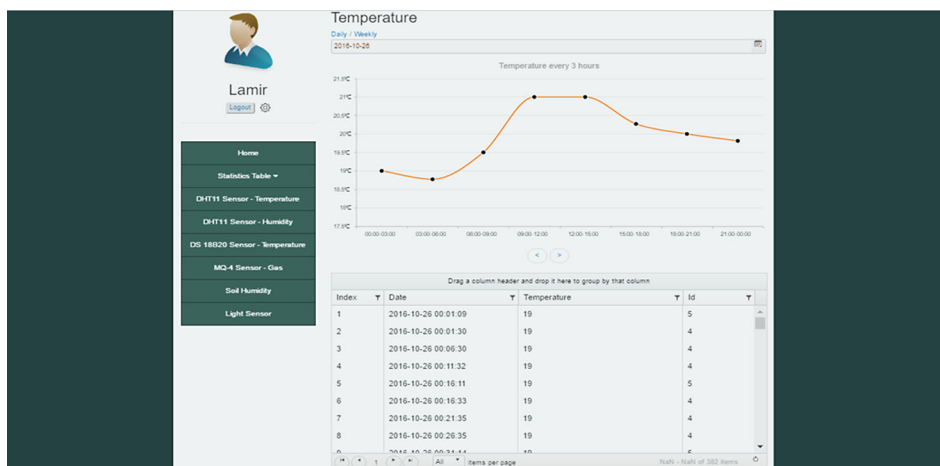


Fig. 6. Temperature graph and data table in the WSN Monitoring application

sensor so that the application can send a warning email to improve the environmental conditions if the sensor sends data that is out of the required range. The application configuration is presented in Fig. 5 [5].

On the left side there are menus and within each menu there are the names of the sensors used in this paper.

The web application has the possibility of monitoring the data weekly, daily, 3-hourly or half-hourly in either table or graphical form. An example temperature graph and data table from the WSN Monitoring application are presented in Fig. 6 [6].

4.2 Web API

In the context of this paper, a web API is also developed that records data received from the sensors and sends a warning email to improve the environmental conditions if the data received from the sensor exceeds the permitted range that the user has set in the WSN Monitoring web application mentioned above. The web API also receives data from the database and returns to the JSON if sensors or the web application make such a request.

In the configuration of the web application, the permissible temperature range is set to 15–25 °C. Therefore, if the temperature falls below the configured value range, the web API sends an email containing a message as presented in Fig. 7.

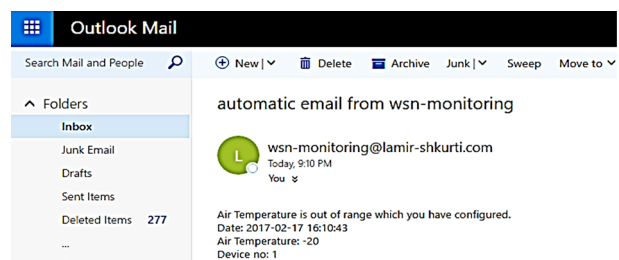


Fig. 7. Content of the email sent by the web API

4.3 The database

The database for the WSN Monitoring application is created in MySQL and features the following tables: users, configuration, temperature, humidity, temperature and soil. In the “users” table, the users who can access the web application are registered. In the “configuration” table, the sensor settings are stored along with the minimum and maximum range of each sensor. In the “temperature” table, the temperature recorded by the DHT11 sensor is stored, whereas the humidity measured by the humidity sensor is stored in the “humidity” table. In the “temp” table, the measured

temperature by the DS18B20 sensor is stored. In the “soil” table, the humidity of the YL-69 sensor is stored.

4.4 Cloud-based mobile development

In today's digital world, one of the fastest growing categories of cloud-based applications is data warehousing and retrieval of real-time data. Most of our interactions, exchanging data between applications and servers, occur in real-time. As mentioned at the beginning of the paper, WSN sensor nodes transmit data to the cloud-based database via web API request. Furthermore, a cloud-based mobile application is also developed for Android users. As cloud computing introduced the concept of Everything as a Service (XaaS) [7], the authors in the work [8] discussed the advantages of developing cloud-based applications. Based on their reported results, developing our model in a cloud-centric architecture can improve the computation capability and energy efficiency of mobile devices.

The WSN Monitoring application is developed using a unified mobile development platform from Google called Firebase. Firebase serves as a back-end as a service that provides real-time data synchronisation with all the clients registered to it, at a given instant. Firebase provides a real-time database for storing and syncing applications data. The data is stored in JSON format and synchronised to every connected client. Furthermore, it automatically backs up user data every day without any performance or bandwidth impact on the application [9].

The system administrator is responsible for configuring the right values for air temperature and the humidity. In this sense, there are two situations that can occur during the monitoring time, which are categorised into two groups:

- Non-critical event
- Critical event

The developed application provides more than one location that can be monitored. After the location is selected and the basic required configuration is given, one of the situations will be triggered. A non-critical event is cases where there is no need to send a notification message to the users. These cases can be considered as non-harming events, which do not cause any system failure owing to the parameters of temperature or humidity. For example, each time the user opens the application it will request the latest data from the database.

A critical event is a case when the responsible users must be informed through the notification messages. A similar approach was also presented in [10], where authors categorised the real-time questions based on their importance in order to increase the quality and transparency of election processes. The developed application and the triggered notification panel are presented in Fig. 8.

The advantage of retrieving real-time data through notifications is the ability to take immediate action when responding to an event in the shortest possible time. This makes it possible to update the database and send an immediate response to the user to provide the latest information from the sensors. Another advantage is that even if the application is closed (running in the background), in critical cases users are automatically informed of an event through the notification.

However, there are also different approaches with lower response times, such as in a server-less platform [11]. The proposed approach enables combining the benefits of the edge computing (lower response time and heterogeneous data management) with the computational and storage capabilities of the cloud. For example, time-sensitive data, such as life-critical vital signs, can be analysed at the edge, close to where data are generated, instead of being transported to the cloud for processing.

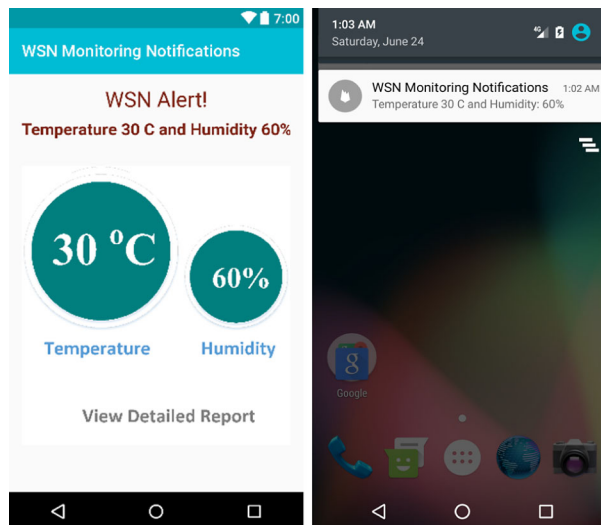


Fig. 8. The main application interface (left) and notification message (right)

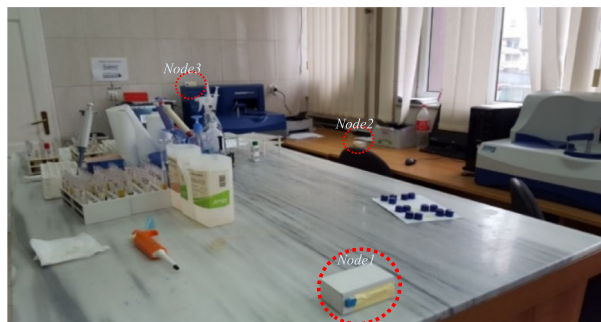


Fig. 9. Placement of sensor nodes in the QKMF laboratory

5. Results

After designing the sensor nodes and developing the web application, the system was tested in the laboratory at the Main Center of Family Medicine (QKMF) in Prizren as well as in the server room of Prizren's Telcomm company, which provides Internet and cable services. Telcomm and QKMF have an extensive network infrastructure throughout the facility and the implementation of the sensor nodes in these buildings was not difficult. Based on the results obtained, the data measured by sensors in both locations does not differ too much from the actual values of temperature and humidity in the mentioned locations so the error range is low. Since we use low-cost sensors, the error values of sensors are acceptable and within the tolerance boundaries. The results for both locations are presented and discussed in the following sub-sections.

5.1 Application in QKMF

The QKMF laboratory was monitored 24 h for 5 days. During working hours from 07:00 to 15:00 the laboratory workers also followed temperature and humidity in the laboratory. The testing equipment used in the QKMF laboratory works only if the temperature is in the range of 15–25 °C. To monitor the laboratory environment at QKMF, four sensor nodes were placed as presented in Fig. 9 and the web application was configured with temperature limits of 15–25 °C and air humidity limits of 40–70%.

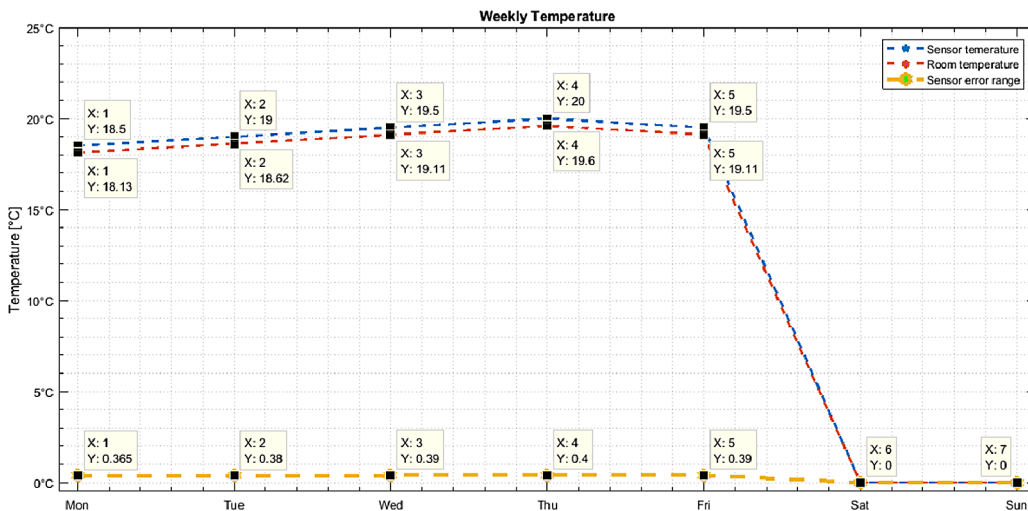


Fig. 10. Graph of the temperature measured over 5 days in the QKMF laboratory

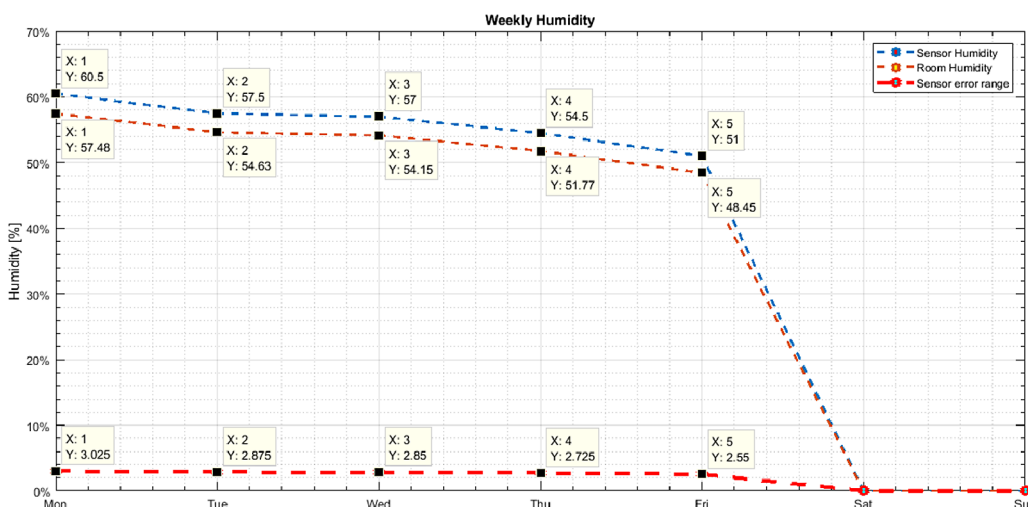


Fig. 11. Graph of the humidity measured over 5 days in the QKMF laboratory

Over the course of 5 days, testing data was stored in the database every 5 minutes. The results were within the range set since the laboratory was continuously monitoring the temperature and it had an air conditioner, which is mainly used in summer when the temperatures are high, and central heating for when temperatures are low.

The data collected from the QKMF laboratory are displayed in Figs. 10 and 11.

5.2 Application in Telcomm

The sensor equipment was also tested in a server room in Telcomm in Prizren for temperature monitoring. For the longevity of the servers, the temperature must be in the range of 18–27 °C. The data were collected for 5 days and stored in the database every 15 min.

In the server room, there is an air conditioning system that is used to ensure the longevity of the equipment by maintaining the optimum temperature. Results derived from the Telcomm server room are presented in Figs. 12 and 13.

Using the application developed in a similar way, the results can also be drawn for the other days on which the sensors have gathered data.

6. Conclusions

Before testing in the laboratory at QKMF and in the server Telcomm room, the system was tested for a few weeks and some shortcomings were found in terms of battery sustainability and the sensor nodes' box. The sustainability with 1.5 V Varta AA alkaline batteries was a maximum of 5 days, but with the application of 3.7 V lithium polymer batteries as described above the battery life of the NodeMCU device was extended up to 2 weeks. The results of the applications at QKMF and Telcomm show that this system is reliable and can be applied for monitoring the temperature and humidity of the air, sending warning emails about the improvement of various environmental conditions and analysing the collected data.

If the data measured by the sensor node exceeds the configured value range, the web application sends a warning email to the users to improve the environmental conditions. Meanwhile, in the future work, in addition to email, a GSM module could be used through which an SMS could be sent or a warning call could be made to the user regarding the changes to the environmental conditions. Regarding the types of sensors, only temperature and humidity sensors were used in the QKMF and Telcomm applications. In addition, in fu-

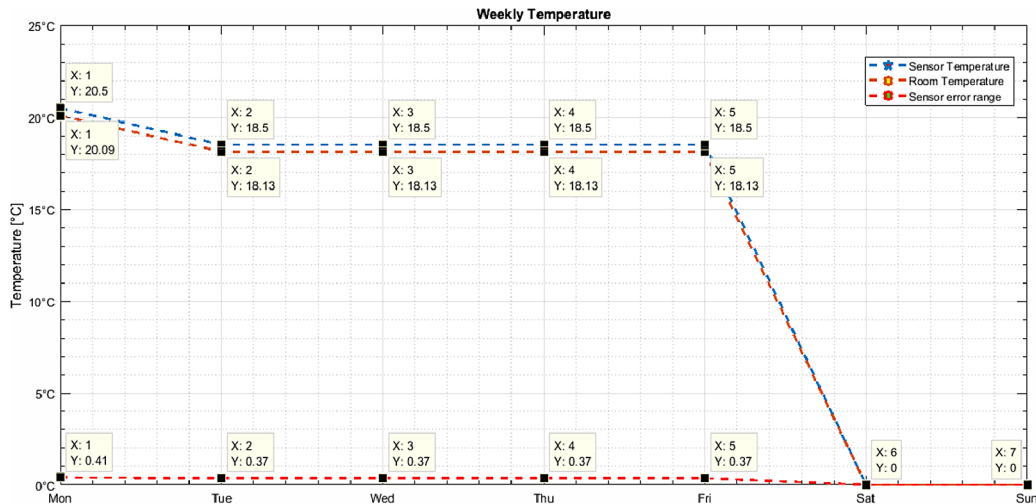


Fig. 12. Graph of the temperature measured for 5 days in the Telecomm server room

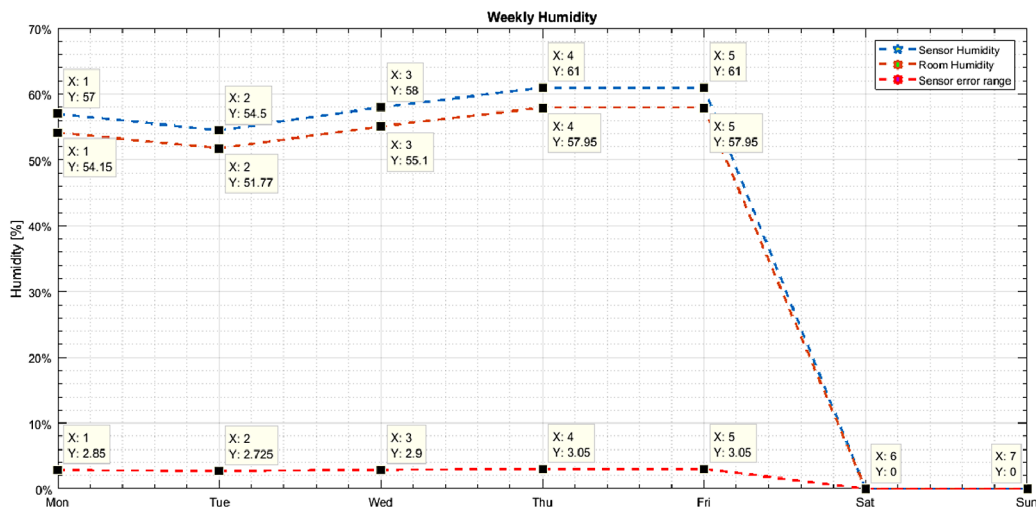


Fig. 13. Graph of the humidity measured for 5 days in the server room in Telcomm

ture work light sensors, PIR sensors for detecting movement, sensors for measuring the level of liquids, sensors for gas density measurements, gyrosopic sensors, optical sensors, MEMS sensors that can also be used as biomedical sensors, and any other sensors that can be connected with UART will be used. The developed system still needs improvement: it needs an easier method to connect to Wi-Fi, self-organising sensor nodes and a box with more sophisticated durability against water, dust and sunlight, for example, constructed using ABS of IP65 or IP67 standards. In addition, in the future, implementation of communication with Bluetooth and commercialisation of the system using an ESP-32S module from the same manufacturer with a combined Wi-Fi and Bluetooth module can be anticipated.

Acknowledgements

Open access funding provided by TU Wien (TUW).

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

- International Electrotechnical Commission (2014): Internet of things: wireless sensor networks. Geneva: International Electrotechnical Commission.
- Hemlata Tomer, K. M. (2015): Study and development of temperature & humidity monitoring system through wireless sensor network (WSN) using Zigbee module. Hemlata Tomer Int. J. Eng. Res. Appl., 5(7), 115–120.
- Mishra, R. Suman, R. Pooja P. (2016): Applicability and realization of security in LEACH protocol. IISTE, 6(3), 2225–2603.
- Jaffe, S. (2016): Design of inexpensive and easy to use day internet of things platform. San Luis: Faculty of California Polytechnic State University.
- Shkurti, L. (2017): Design and implementation of sensors in wireless sensor networks for smart home applications. Master thesis, Prizren.
- Shkurti, L., Bajrami, X., et al. (2017): Development of ambient environmental monitoring system through wireless sensor network (WSN) using NodeMCU and "WSN monitoring". In 2017 6th Mediterranean conference on embedded computing, MECO, New York: IEEE Conference Publications.
- Moreno-Vozmediano, R., Montero, R. S., Llorent, I. M. (2013): Key challenges in cloud computing: enabling the future internet of services. IEEE Internet Comput., 17(24), 18–25.
- Murturi, A., Kantarci, B., Oktug, S. (2015): A reference model for crowdsourcing as a service. In IEEE 4th international conference on cloud networking, CloudNet (pp. 64–66).
- Firestore Google. Online available at: <https://firebase.google.com/>.

10. Rexha, B., Murturi, I., et al. (2016): Improving quality of election process using crowd-sourcing techniques. *Int. J. Appl. Math. Electron. Comput.*, 4(4), 107–112.

11. Nastic, S., Rausch, T., Scekcic, O., Dustdar, S., et al. (2017): A serverless real-time data analytics platform for edge computing. *IEEE Internet Comput.*, 21(4), 64–71.

Authors



Xhevahir Bajrami

studied design and robotics at the University of Prishtina, Kosovo, Albania, where he received the M.Sc. in 2010. He finished the Doctoral studies at the Institute for Handling Devices and Robotics, at Vienna University of Technology (TU Wien), in 2013. He developed kinematics and dynamics for the Biped robot “Archie”. His research interest includes robotics, mechatronics, neural networks, genetic algorithms and fuzzy logic. Actually, he holds a position as assistant professor in several public universities where he teaches courses on sensors, microcontrollers, image processing, mechatron-

ics, sensor networks, artificial intelligence, machine learning, 3D modelling and animation.



Ilir Murturi

is a Ph.D. student and university assistant at the DSG, TU Wien, Austria. His research interests include IoT, edge computing, crowd-sourcing, privacy and smart cities. Murturi has an M.Sc. in computer engineering from the Faculty of Electrical and Computer Engineering, University of Prishtina.