Remote monitoring of an object using a wireless sensor network based on NODEMCU ESP8266

Hicham Ouldzira[1], Ahmed Mouhsen[2], Hajar Lagraini[3], Mostafa Chhiba[4], Abdelmoumen Tabyaoui[5], Said Amrane[6]

[1,2]Laboratory of Mechanical, Engineering, Industrial Management and Innovation,
The Faculty of Sciences and Technology, Hassan 1st University, Morocco
[3,4,5]Radiation Materials and Instrumentations Laboratory, The Faculty of Sciences and Technology,
Hassan 1st University, Morocco
[6]National Institute of Posts and Telecommunication, Morocco

Article Info	ABSTRACT
	In recent years, wireless sensors networks (WSNs) have been imposed as an effective means of interconnection with simultaneous communication and information processing. They allow operating with sensors at low cost and low power consumption in various application areas such as ecosystem monitoring, detection and monitoring of objects and smart cities, etc.This paper describes the development of a system to detect the presence of an object and monitor it. This prototype is based on four NodeMcu modules (a static access point that provides the WIFI network, a server, a client and a mobile access point attached to the remote surveillance object) programmed under Arduino IDE and communicating between them via the HTTP protocol. The remote monitoring of the object for a linear disposition of the nodes used is based on the existence of the mobile access point in the HTTP client field.

Corresponding Author:

Hicham Ouldzira,
Laboratory of Mechanical, Engineering, Industrial Management and Innovation,
The Faculty of Sciences and Technology, Hassan 1st University,
PO Box 577, Settat, Morocco.
Email: odzira@yahoo.fr

1. INTRODUCTION

Today, due to advances in technology and the Internet of Things, step-by-step tasks such as mailing, accessing bank records, monitoring the degree of other tasks [1], eHealth [2], smart home [3-4], smart city [5], intelligent transport [6], wireless sensor networks [7], smart grid [8], objects of remote monitoring systems [9]. However for any IoT application[10], it is important to consider some issues related to this technology such as security and energy efficiency especially in WSN [11,12], that is an ad hoc network consists of several nodes can receive and emit data autonomously [13] . These nodes are organized in the network according to different topologies [14]. Various research have been used to solve the problems detection and security of objects such as ambient assisted living [15] and k-nearest neighbor [16].

In this paper, we present a realization WSN that allows remotely monitor an object in anywhere in the world. The wireless sensors constituting our network are NodeMcu (ESP8266 [17,18]), that plays the main role in this system for the exchange of data. The creating a WiFi network is through the interconnection between the client and the server via an access point. The realized prototype consists of four NodeMcu (ESP8266) which are low-cost and low-power modules [19]. The object is monitored by the communication

between the mobile access point (NodeMcu 4) and the client (NodeMcu 3) which in turn communicates with the server (NodeMcu 2) via the static access point (NodeMcu 1). Note that the communication between the different nodes is done by the HTTP protocol of the library ESP8266.h and the integrated development environment Arduino_IDE.

The rest of the paper is organized as follows. In section 2 related works were defined. In section 3 proposed system were explained. In section 4 voltage adaptation between the NodeMcu and GSM Module. In section 5 Hardware realizations were shown. In section 6 System implementation and the conclusion and future work was discussed in section 7.

2. RELATED WORK

Among the researches that have been carried out on remote surveillance of objects is the RSSI method [20–22], which is based on the measurement of the distance between two nodes as a function of the power of the signal transmitted between them. then there are three distance estimation techniques such as Curve Fitting Technique (CFT), Estimated Signal Strength (ESS) and Friis Transmission Equation (FTE) [23]. Despite the advantages of CFT over ESS and FTE, it has a problem in response time for real-time systems. To overcome this inconvenient we will exploit the presence or absence of the remote-controlled object within the area covered by the ESP8266 module.

2.1. NodeMcu (ESP8266)

NodeMCU is also called ESP8266 Wi-Fi chip. This card can act in station mode or access point mode in Figure 1. In station mode, it plays the role of wireless sensor, it connects to available Wi-Fi networks. In Access Point mode, it can act as an access point so that other devices can connect to this module. Wi-Fi communication works under the standard IEEE 802.11 b / g / n protocol. The microcontroller used on this board is ESP8266EX [24], which is characterized by:
  a) The operating voltage of the MCU is 3.3V.
  b) It has 11 digital input output pins.

This node plays a vital role in wireless sensor networks.

In our case, we used four NodeMcu to create two access points, an HTTP client and an Http server as shown in Figure 3.

Figure 1. The image of the NodeMcu used

2.2. Client-Server HTTP Protocol

The Hypertext Transfer Protocol (HTTP) is an application-level protocol for distributed, collaborative, hypermedia information systems [25]. It can transfer plain text, hypertext, audio, images, and Internet accessible information. The client–Server environment means a communication across a network between multiple programs. The server (NodeMcu 2) waits for requests sent by the client (NodeMcu 3) and responds to them. The HTTP request is a rowset sent to the server consisting of a query line, the query header fields, and the query body as shown in Figure 2.
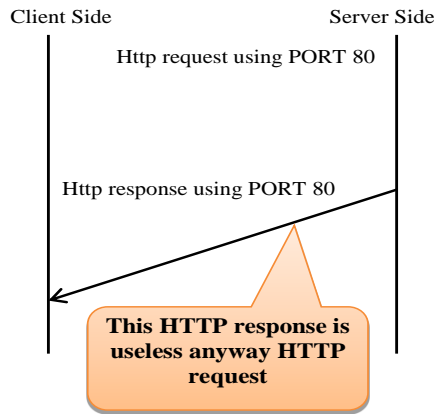
Client Side   Server Side

Http request using PORT 80

Http response using PORT 80

This HTTP response is useless anyway HTTP request

Figure 2. Communication between the HTTP client and the HTTP server

2.3. Access Point

The wireless access point (AP) is a network device that allows Wi-Fi devices to communicate with each other. It plays the role of transmitter and central receiver of wireless radio signals. Our prototype contains two access points one static (NodeMcu 1) and the other mobile (NodeMcu 4).
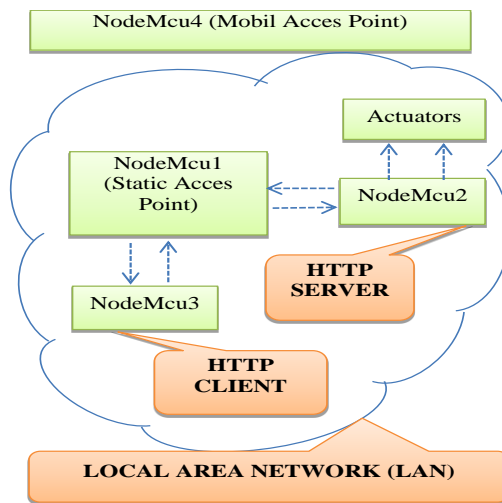
NodeMcu4 (Mobil Acces Point)

Actuators

NodeMcu1 (Static Acces Point)

NodeMcu2

HTTP SERVER

NodeMcu3

HTTP CLIENT

LOCAL AREA NETWORK (LAN)

Figure 3. the location of access points

2.4. Actuators

The server is connect to BUZZER module and GSM module as shown in Figure 4.

NodeMcu2 (HTTP Server)
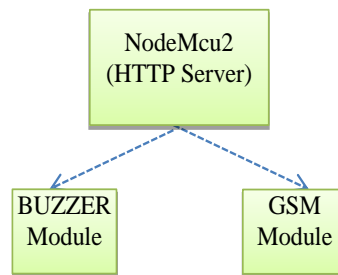
BUZZER Module

GSM Module

Figure 4. Communication between HTTP Server and Actuators

2.4.1 BUZZER Module

A buzzer module is an audio signaling device, which may be mechanical, electromechanical, or piezoelectric. Typical uses of buzzers and beepers include alarm devices, timers and confirmation of user input such as a mouse click or keystroke. It is an integrated structure of electronic transducers, widely used in computers, printers, copiers and other electronic products for sound devices.

2.4.2 GSM Module

GSM (Global System for Mobile Communication) is a communication module which was developed at bell laboratories in 1970. It is a digital cellular technology which was used for providing data and voice oriented services by using time division multiple access technique for communication purpose. It has a capacity to carry 64kbps to 120Mbps of data rates in Figure 5. GSM modem is a device which can be either a mobile phone or any modem which is used for communication between two different devices over any network. It works with a frequency of 1800 KHz and operating voltage is 5v. After initializing the GSM modem some attenuation commands are used to set the modem into text mode and for storing the messages. It will allow communication from anywhere, anytime and with anyone.
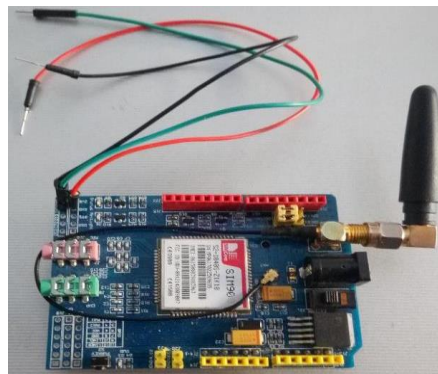

Figure 5. GSM Module

AT commands are used to initiate the GSM modem to perform any job. The list of some important AT commands used in this design is given in Table 1 below:

Table 1. Description of AT Commands for SMS Text Mode

Command	Description
AT+CSMS	Select message service
AT+CPMS	Preferred message storage
AT+CMGF	Message format
AT+CSMP	Set text mode parameters
AT+CSDH	Show text mode parameters
AT+CMGL	List messages
AT+CMGR	Read message
AT+CMGS	Send message
AT+CMSS	Send message from storage
AT+CMGW	Write message to memory
AT+CMGD	Delete message

3. PROPOSED SYSTEM

The NodeMcu1 static access point is used to create a LAN between NodeMcu3 that will function as an HTTP client and the NodeMcu2 being an HTTP server. The exchange of data between these nodes is done via port 80 (using HTTP standard). The relative access point to the NodeMcu4 is attached to the monitored object as shown in Figure 6.

Figure 7 depicts the customer layout and the separate static access point of a distance $(\xi-\chi)$ in such a manner that when the mobile access point is remote from the static access point by a distance superior to χ, The system must activate the GSM module and the BUZZER module as shown in Figure 8.

If the distance between the mobile access point and the static access point is less than or equal to χ, this means that the remote monitoring object is included in the monitoring field and the GSM and BUZZER modules remain in a state of rest however, as shown in Figure 8.

The remote monitoring of the object is done by the detection of the mobile access point by the HTTP client on the network relative to the static access point and it also scans the access points of his entourage with the relative point of access. This latter detects an HTTP request on the port 80, which is sent to the server HTTP indicating the presence or the absence.
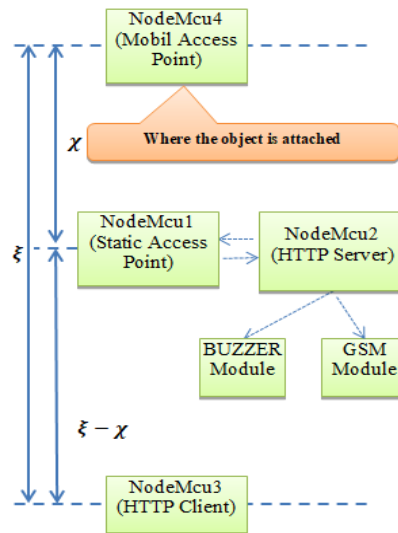
Figure 6. Communication between HTTP Server and Actuators

In this Figure 6, the object is attached to NodeMcu 4 which is the mobile access point. We propose that the object is remote at the static access point by the distance x, while keeping this distance using the HTTP client, hence the HTTP server is off.

In the Figure 7, the monitored object is still in the HTTP client's monitoring field, so the HTTP server is stopped.

In this Figure 8 the client HTTP Send an http request to the server via port 80 indicating that the object has left the field, and subsequently the HTTP server activates the BUZZER module and sends an SMS to the owner using GSM module.
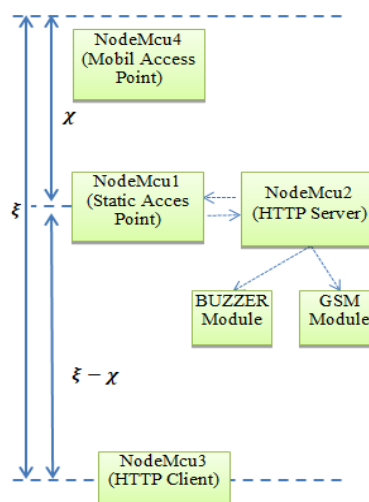
Figure 7. The object is in the surveillance field

4. VOLTAGE ADAPTATION NODEMCU & GSM

The NodeMcu operates on 3.3 volts and the GSM module operates with a 5-volt power supply which poses a problem of voltage adaptation between them. Since both modules operate at different voltage levels, the commands and messages in each module must first be converted to receiver lifetime before reaching them. The TX NodeMcu pin sends signals at the TTL level of 3.3 v. The TTL 3.3v signal is converted to TTL 5v signal before reaching the GSM module. The first transistor converts the TX signal to 5 volts output but the output is inverted. The inverted output is input to the second transistor. This converts the inverted output to an original signal with a TTL level of up to 5v. The output TTL signal is received by the RX pin of the GSM module. The Tx of the latter is connected to the NodeMcu Rx with a single level shift circuit between the levels. to solve this problem, we establish the circuit diagram shown in the Figure 9.
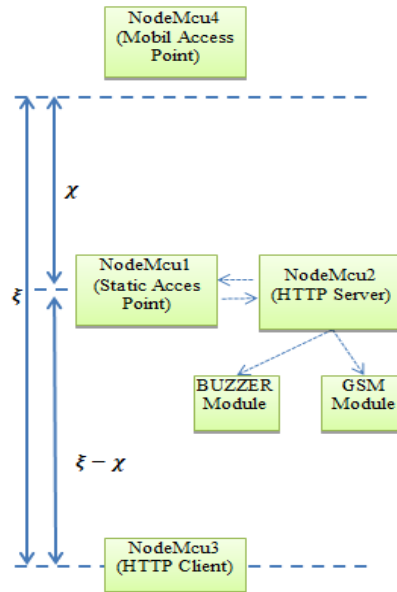

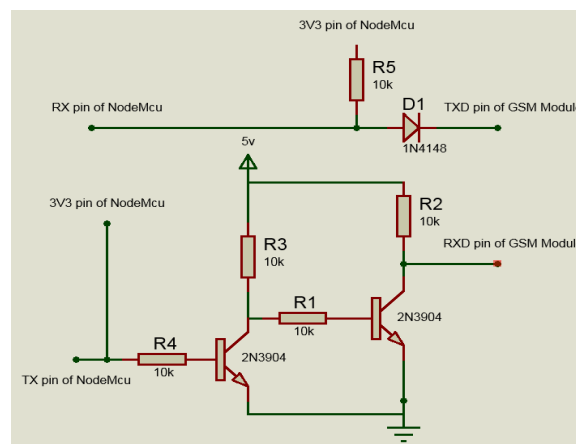Figure 8. The object is out of the surveillance field

Figure 9. Adaptation of the voltage between NodeMcu 2 and GSM module

5. HARDWARE REALIZATION

It is The Figure 10 shows the components of an "object detection and monitoring" system that includes: 1) a mobile access point (NodeMcu4 that does not exist in the figure), 2) an access point (NodeMcu1), 3) an HTTP client (NodeMcu3), 4) an HTTP server (NodeMcu2), 5) a BUZZER module, 6) a GSM module and 7) Accumulator battery.
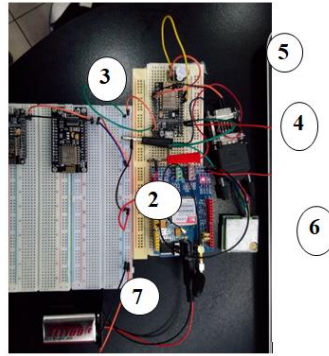
Figure 10. Components of the remote monitoring system of an object in real time

6.   SYSTEM IMPLEMENTATION

The flowchart in Figure 11 describes the Sending an http request to the server via port 80 indicating that the object inside the field (Using GET method), but in case the object has left the fields Figure 12, the HTTP server Send a command to the buzzer, making a SMS to the owner of the object.
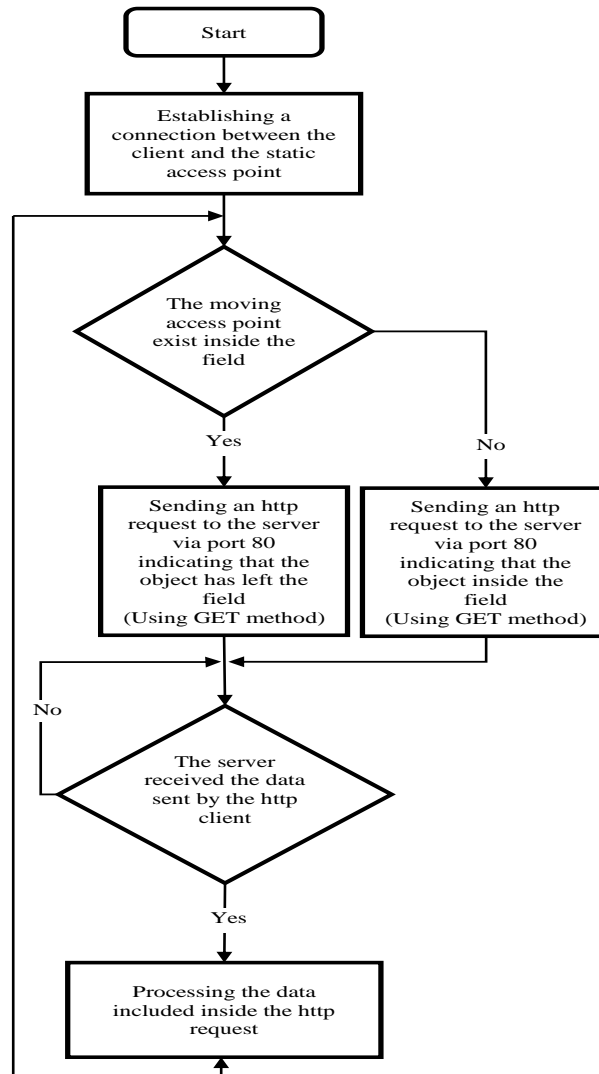

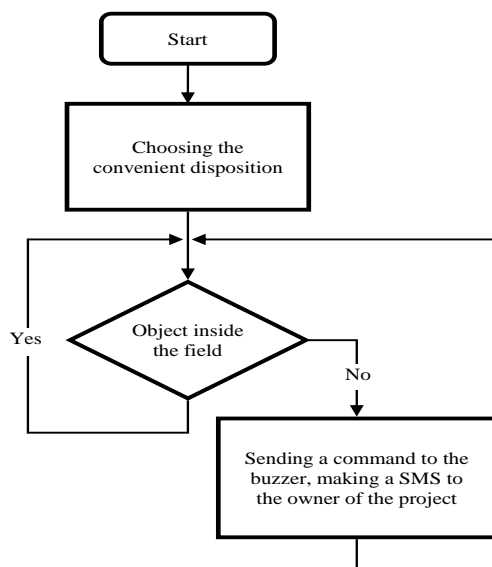Figure 11. The functional flowchart of the system

Figure 12. Flowchart remote monitoring of the object

7.   CONCLUSION

In this paper, we proposed a system that overcomes the real-time system response time inconvenience based on the RSSI method. This prototype has remotely monitored an object in any location based on the interconnection of four NodeMcu via port 80 of the HTTP protocol. When the object is outside the field controlled by the static access point, the HTTP client sends an HTTP request to the server that activates the GSM module and the BUZZER module. The GSM module initiates a phone call and sends a message to the owner using the AT commands. It makes it possible to remotely monitor an object for a linear disposition of the nodes used, in perspective a study is envisaged for some provisions in order to control the object whatever its displacement on the ground to be watched and also possible to locate it even in outside the monitored field using a GPS module.

REFERENCES

[1]   P. Gupta, D. D. Shah, et K. V. V. Satyanarayana, « An IoT Framework for Addressing Parents Concerns about Safety of School Going Children », *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 6, nº 6, p. 3052, déc. 2016.
[2]   N. A. A. Rahman et A. B. Jambek, « Biomedical health monitoring system design and analysis », *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 13, nº 3, p. 1056‑1064, mars 2019.
[3]   B. Li et J. Yu, « Research and Application on the Smart Home Based on Component Technologies and Internet of Things », *Procedia Engineering*, vol. 15, p. 2087‑2092, 2011.
[4]   S. Kumar, « Ubiquitous Smart Home System Using Android Application », *International journal of Computer Networks & Communications*, vol. 6, nº 1, p. 33‑43, janv. 2014.
[5]   V. A. Memos, K. E. Psannis, Y. Ishibashi, B.-G. Kim, et B. B. Gupta, « An Efficient Algorithm for Media-based Surveillance System (EAMSuS) in IoT Smart City Framework », *Future Generation Computer Systems*, vol. 83, p. 619‑628, juin 2018.
[6]   P. S. Saarika, K. Sandhya, et T. Sudha, « Smart transportation system using IoT », in *2017 International Conference On Smart Technologies For Smart Nation (SmartTechCon)*, Bangalore, 2017, p. 1104‑1107.
[7]   V. Suryani, S. Sulistyo, et W. Widyawan, « ConTrust: A Trust Model to Enhance the Privacy in Internet of Things », *International Journal of Intelligent Engineering and Systems*, vol. 10, nº 3, p. 30‑37, juin 2017.
[8]   R. Habachi, A. Touil, A. Boulal, A. Charkaoui, et A. Echchatbi, « Resolution of economic dispatch problem of the morocco network using crow search algorithm », *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 13, nº 1, p. 347, janv. 2019.

[9]   K. Radharamana, C. Mouli, et U. Kumar, « Web Architecture for Monitoring Field using Representational State Transfer Methods », *International Journal of Intelligent Engineering and Systems*, vol. 12, nᵒ 1, p. 84‑93, 2019.

[10] P. Asghari, A. M. Rahmani, et H. H. S. Javadi, « Internet of Things applications: A systematic review », *Computer Networks*, vol. 148, p. 241‑261, janv. 2019.

[11] J. Govindasamy et S. Punniakodi, « Energy Efficient Intrusion Detection System for ZigBee based Wireless Sensor Networks », *International Journal of Intelligent Engineering and Systems*, vol. 10, nᵒ 3, p. 155‑165, juin 2017.

[12] J. Arora, Gagandeep, et R. Kumar, « IoT-Based Smart Home Systems », in *Innovations in Computer Science and Engineering*, vol. 32, Singapore: Springer Singapore, 2019, p. 531‑538.

[13] Y. Khamayseh, M. B. Yassein, et M. Abu-Jazoh, « Intelligent black hole detection in mobile AdHoc networks », *IJECE*, vol. 9, nᵒ 3, p. 1968, juin 2019.

[14] R. R. Selmic, V. V. Phoha, et A. Serwadda, « Topology, Routing, and Modeling Tools », in *Wireless Sensor Networks*, Cham: Springer International Publishing, 2016, p. 23‑36.

[15] H. Malekmohamadi, A. Moemeni, A. Orun, et J. K. Purohit, « Low-Cost Automatic Ambient Assisted Living System », in *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, Athens, 2018, p. 693‑697.

[16] Xiaohui Yu, K. Q. Pu, et N. Koudas, « Monitoring k-Nearest Neighbor Queries over Moving Objects », in *21st International Conference on Data Engineering (ICDE'05)*, Tokyo, Japan, 2005, p. 631‑642.

[17] J. Julham, H. A. Adam, A. R. Lubis, et M. Lubis, « Development of soil moisture measurement with wireless sensor web-based concept », *IJEECS*, vol. 13, nᵒ 2, p. 514, févr. 2019.

[18] C.-P. Ooi, W.-H. Tan, S.-N. Cheong, Y.-L. Lee, V. M. Baskaran, et Y.-L. Low, « FPGA-based embedded architecture for IoT home automation application », *IJEECS*, vol. 14, nᵒ 2, p. 646, mai 2019.

[19] L. Shkurti, X. Bajrami, E. Canhasi, B. Limani, S. Krrabaj, et A. Hulaj, « 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)*, Bar, Montenegro, 2017, p. 1‑5.

[20] W. Xue, W. Qiu, X. Hua, et K. Yu, « Improved Wi-Fi RSSI Measurement for Indoor Localization », *IEEE Sensors J.*, vol. 17, nᵒ 7, p. 2224‑2230, avr. 2017.

[21] J. Janicka et J. Rapinski, « Application of RSSI Based Navigation in Indoor Positioning », in *2016 Baltic Geodetic Congress (BGC Geomatics)*, Gdansk, Poland, 2016, p. 45‑50.

[22] S. Li, X. Yang, R. Zhao, Y. Liu, X. Zhou, et L. Zhang, « An Indoor Positioning Method Based on RSSI Probability Distribution », *IOP Conf. Ser.: Mater. Sci. Eng.*, vol. 490, p. 042054, avr. 2019.

[23] S. Barai, D. Biswas, et B. Sau, « Estimate distance measurement using NodeMCU ESP8266 based on RSSI technique », in *2017 IEEE Conference on Antenna Measurements & Applications (CAMA)*, Tsukuba, 2017, p. 170‑173.

[24] Mediterranean Conference on Embedded Computing *et al.*, *2017 6th Mediterranean Conference on Embedded Computing (MECO) including ECYPS'2017: proceedings : research monograph : Bar, Montenegro, June 11th-15th, 2017*. 2017.

[25] R. Fielding *et al.*, « Hypertext Transfer Protocol -- HTTP/1.1 », RFC Editor, RFC2616, juin 1999.