Internet of Things Remote Laboratory for MQTT remote experimentation

Jesús Anhelo[1], Antonio Robles and Sergio Martin[1]

[1] Universidad Nacional de Educación a Distancia, Madrid, Spain
smartin@ieec.uned.es

Abstract. Remote laboratories have matured substantially and have seen widespread adoption across universities globally. This paper delineates the design and implementation of a remote laboratory for Industry 4.0, specifically for Internet of Things. It employs Raspberry Pi and ESP8266 microcontrollers, to bolster online Internet of Things (IoT) learning and experimentation platforms. Such platforms hold significant value in delivering high-quality online education programs centered on IoT. Students have access to a web interface where they can write Arduino code to program the behavior of each one of the nodes of an Internet of Things scenario. This setup allows them to remotely program three NodeMCU boards in a manner akin to the usage of the Arduino IDE connected to an Arduino board locally. The system offers the ability to compile and upload code, complete with error notifications. Additionally, it furnishes several functionalities such as the ability to load new local code, save the authored code to one's personal computer, load predefined examples, access a serial monitor, and avail the Node Red platform. This amalgamation of features promises to offer a comprehensive and interactive remote learning experience for students engaging with IoT technologies.

Keywords: Remote Laboratories; On-line labs; Internet of Things; Raspberry Pi; NodeMCU; ESP8266; Node-Red; Arduino; HTLM; Javascript; PHP; MQTT; Mosquitto Broker.

1 Introduction

As the technological landscape advances at an unprecedented pace, the fourth industrial revolution, often referred to as Industry 4.0, has emerged as a transformative force driving digital innovation across global industries. Central to this movement is the Internet of Things (IoT), which interconnects a plethora of devices, creating intelligent networks and systems that are capable of perceiving, learning, and acting autonomously.

The seamless integration of physical and digital spaces offered by IoT has substantial implications for sectors ranging from manufacturing to healthcare, logistics, and beyond. Through enhanced connectivity, data sharing, and real-time analytics, IoT provides an impetus for innovative approaches to process optimization, predictive maintenance, and decision-making strategies that are set to reshape business models and market dynamics. Therefore, the understanding and application of IoT in the context of Industry 4.0 is crucial in the current era of technological advancement.

However, the advent of such complex and interwoven systems demands a well-prepared workforce that can navigate the intricacies of these technological transformations. High-quality education is essential in preparing both future professionals and current practitioners to face these emerging challenges and capitalize on the opportunities they present. Particularly in light of the ongoing pandemic, online educational programs have become an imperative tool for disseminating knowledge and skills to large, geographically dispersed populations.

In this context, online laboratories play an integral role in providing hands-on, practical experience in the realm of IoT within Industry 4.0. These virtual learning environments replicate real-world systems and allow for interactive, experiential learning that goes beyond traditional, lecture-based pedagogy. They facilitate the exploration and understanding of IoT systems' design, deployment, and management, fostering critical competencies required in today's digitized industrial landscapes.

This paper delves into the intersection of Industry 4.0, IoT, and the vital role of online laboratories in providing quality education on these subjects. By illuminating the pivotal function of such platforms in driving successful digital transformation, we aim to contribute to the literature on Industry 4.0 and IoT education and underscore the necessity for robust, immersive online learning experiences in cultivating the workforce of the future.

2 Remote laboratories

To address these educational challenges, the concept of remote laboratories was introduced. These are physical laboratories that can be controlled remotely, in contrast to virtual labs that merely simulate the intended environment. Remote laboratories offer several benefits [1]:

- Accessibility: Remote labs are available 24/7 from anywhere with an internet connection, offering students the flexibility to access the laboratory according to their convenience.
- Collaboration: These labs can be easily shared among different universities, with numerous projects and consortiums formed specifically for this purpose.

Despite these advantages, remote labs present a significant challenge - supervision. In the absence of physical staff to aid and guide students, the reliance on automated support systems is high. While some progress has been made in this domain, there is still a substantial need for improvement. Ideal remote laboratories should possess the following characteristics [1]:

- High Availability: The lab should be accessible without any time restrictions.
- Concurrency: The lab should have the capability to support multiple users simultaneously.

- Cost-effectiveness: The construction and operational costs of the lab should be as minimal as possible.
- Resilience: The lab must be designed to handle misuse effectively.

One prevalent approach to creating such remote labs involves the use of embedded web-based remote monitoring systems, which have several advantages over traditional PC servers:

- Cost Efficiency: This solution is cost-effective as it utilizes inexpensive hardware and open-source software.
- Low Power Consumption: The power requirements for these systems are minimal.
- Compact Size: The equipment involved is small in size.
- Security: As the web server operates using HTML and PHP on port 80, the firewall remains secure.
- Scalability: This type of remote lab can support multiple users.
- Remote Programmability: The lab can be programmed remotely, adding to its convenience.

3 Development of IoT-MQTT remote laboratory

In this chapter it is described the design and development of an IoT remote lab for MQTT experimentation to allow students to program and interact with the real devices over the Internet. It is based on a previous version of the authors [3] and the conclusions obtained in related developments [4, 5] with the addition of new hardware, software functionalities and learning outcomes.

The architecture of this new IoT lab is shown in fig. 1. It is based on a Raspberry Pi with Apache2 software as HTTP server connected to three nodeMCU boards based on the microcontroller ESP8266. They can be monitored through a web cam.

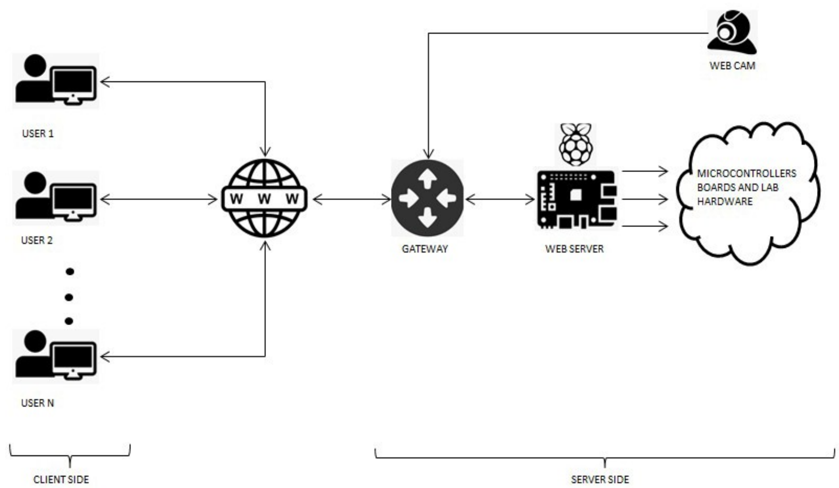

Fig. 1. Client-Server Architecture of the Lab.

The remote laboratory provides a website with an Arduino code editor, where the student can write the code to be executed in each one of the 3 boards (Fig. 2); and a webcam that shows the result of such execution on the visual peripherals (LEDs, display).

Fig. 2. Screenshot of the web interface of the remote lab, including code editor (left) and webcam video stream (right).

The software supporting the remote lab is UNED Arduino Remote Lab, based on PHP, already described by authors in [3]. This paper presents a different experiment (focused on MQTT experimentation) with different boards (NodeMCU instead of Arduino) and electronic components. Thus, different IoT scenarios are designed. Also, for the development of this experiment, the serial monitor functionality has been implemented, to allow users communicate with the boards through serial. This is extremely useful for debugging purposes.

Another improvement of this remote lab is the installation of Eclipse Mosquitto MQTT Broker. It is an open-source message broker that implements the MQTT protocol versions 5.0, 3.1.1 and 3.1. It is lightweight and is suitable for use on all devices from low power single board computers to full servers. Thanks to this MQTT broker, different MQTT practices can be carried out remotely.

The lab consists of a Raspberry Pi connected, via USB, to three breadboards with NodeMCU boards, LEDs, humidity and temperature sensors, light intensity sensors, a buzzer, and a display.

The Raspberry Pi will communicate with the NodeMCU boards through USB connections and the NodeMCUs will communicate with each other through Wi-Fi connection. As Fig. 3 shows, each one of the boards have connected several electronic components. The breadboards will be named as follow: Display Board; Sensors board 1; and Sensors Board 2.

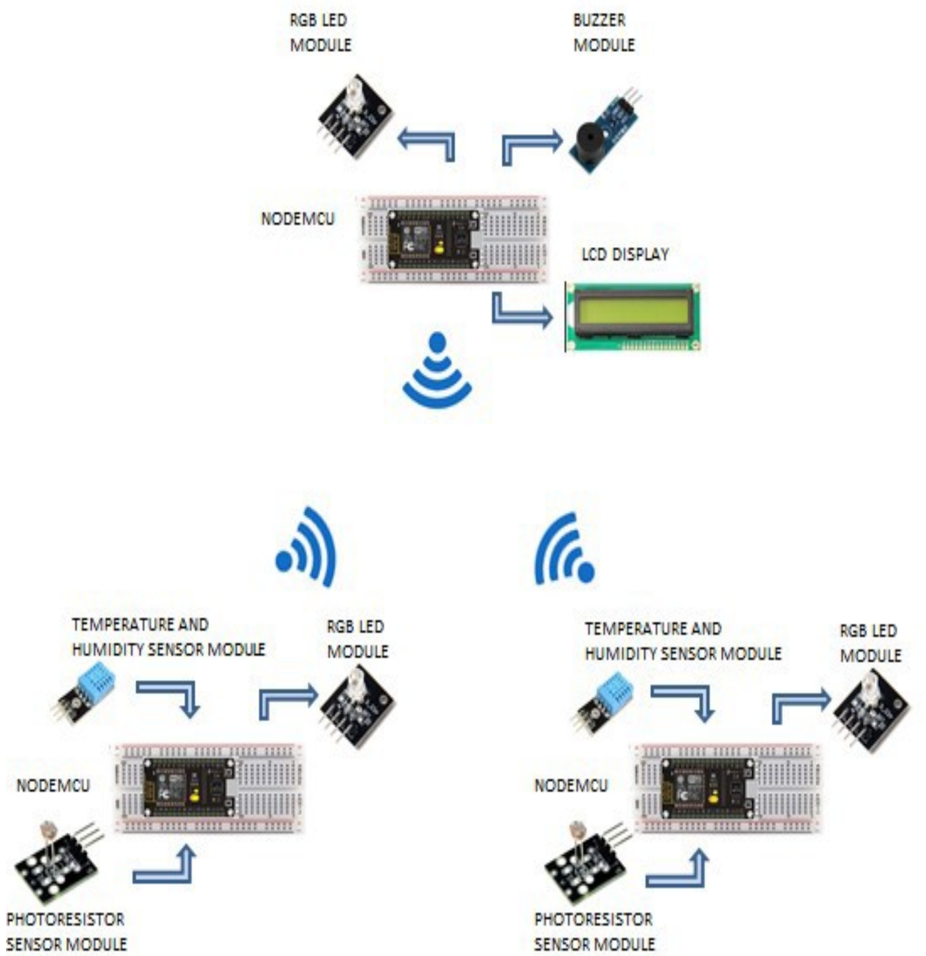

Fig. 3. IoT-MQTT remote laboratory components diagram.

4 Practices designed for MQTT remote experimentation

In this section some exercises are proposed to the students so they can learn how to program an IoT environment using the MQTT protocol in a progressive way. The practices are organized in 6 different scenarios:

1. Scenario 1. Lear how to program the different electronic components of the boards with no Wi-Fi connection. Several practices are designed in this scenario:
   a. Using the LED. The aim of this first practice is to practice with PWM programming through changing a LED intensity progressively.

b. Using buzzer. The goal of this practice is the use of the buzzer KY-006. The intensity of the module KY-016 is increased until a certain level is reached when an alarm is activated, and the buzzer sounds, and the LED is set in red color.
c. Using LCD with I2C. In this practice a LCD1602A display with I2C bus will be used. The intensity in the led will increase until a limit in which the led will display in red and an audible alarm will sound. The display will show the instant intensity value until the alarm value from which the display will show the text "alarm".
d. Using DHT11. The goal of this practice is the use of the humidity and temperature sensor DHT11. The output of this module is a digital signal. The led will be on in green color indicating the system is running. The data of temperature and humidity will be printed in the serial monitor.

2. Scenario 2. Master node creates Wi-Fi network. In the second scenario the ESP8266 will be used in Access Point mode (AP) to generate its own WIFI network. Several practices are designed in this scenario:
   a. One board will be the Access Point to generate de network and the rest of devices will be in Station mode and can connect to this network to communicate data. This is an access point enabled by software in a device that was not originally created to be a router and it is called "SoftAP". The access point device acts as the hub of all communications. It is the TCP/IP router to the rest of the network. Packets bound for devices within the WLAN need to go to the correct destination. The SSID keeps the packets within the correct WLAN, even when overlapping WLANs are present. However, there are usually multiple access points within each WLAN, and there must be a way to identify those access points and their associated clients. This identifier is called a basic service set identifier (BSSID) and is included in all wireless packets.
   b. Generate a Wi-Fi network with a NodeMCU in Access Point mode and two NodeMCUs in Station mode with static IP. This exercise is like the previous one except that a static IP is stablished for each board. A static IP is useful to access the device without the necessity of connect a computer to the device for knowing the IP.
3. Client-Server Communication between 2 nodes connecting to the Server's network with basic actions. In this scenario two boards will communicate through a client-server connection over HTTP in the Wi-Fi network created by the Access Point. The client will send a request to the server to the server carry out an action. The client will request through the GET method the root URI / and the server will send the "Hello" message to the client which we can see in the serial monitor and in the display.
4. Server-Client Communication between 2 nodes connecting to the Client's network with advanced actions. In this scenario two boards will communicate through a client-server connection over HTTP. A board, as server, will collect temperature, humidity and light intensity data from the environment and will send these data to the other board, as client. The client node will display the information in a LCD and will

send a command to the server to turn the led on in green color in the server board during each lecture. The LED in the Client board will be turn on as well.

5. Server-Client Communication between 3 nodes connecting to the server board's network with advanced actions. In this scenario the board with the LCD will be the Server and Access Point. The other two boards will be Clients and Stations in the Wi-Fi network collecting temperature, humidity and light intensity values and sending them to the Server (Fig. 3).

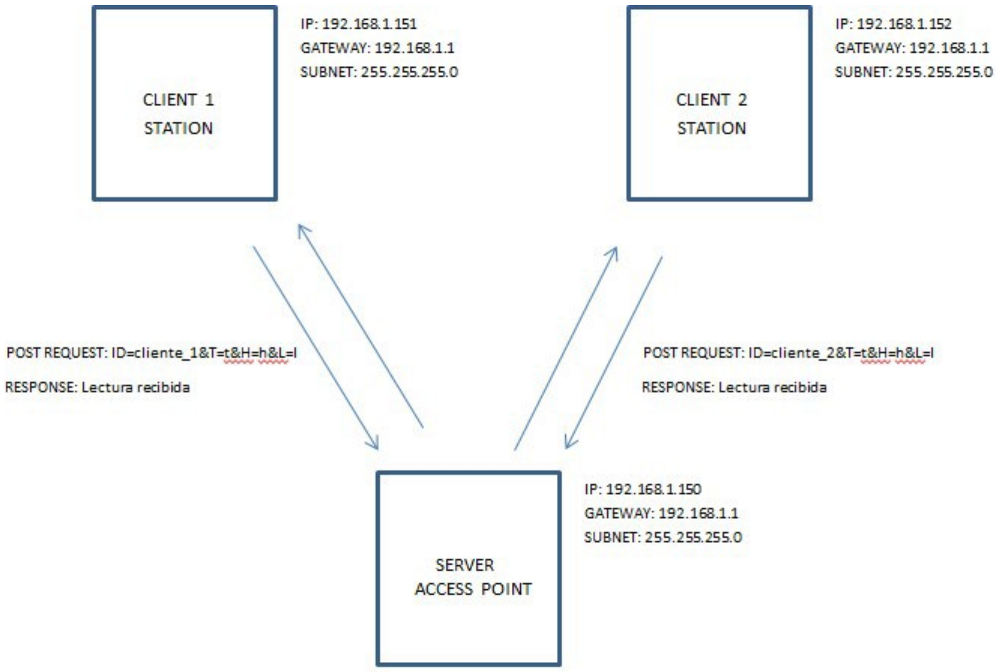

Fig. 3. Boards Configuration.

6. MQTT Communication with one NodeMCU board as broker. The task related to this scenario is the task 11. In this scenario the measurements of the sensors will be communicate via MQTT protocol with a nodeMCU board acting as broker. The Broker routes messages between clients. Each client device which connects to an MQTT broker is asked to provide a unique client identifier. The broker uses this unique client identifier to track clients and push messages to them. The main concept to understand in MQTT is "topics" – these are channels of information which clients can publish to or subscribe to messages from. Topics in MQTT are organized in a hierarchy separated by forward slashes, e.g.: iot_widgets/lightbulbs/light_88234323. An MQTT message simply consists of a topic (string) and a payload (byte array). It is up to the clients to decide what conventions to follow for payloads. Once a client device is connected to an MQTT broker, it can publish a message at any time by specifying the topic and payload. Once a client device is connected to an MQTT broker, it can also subscribe to a topic by specifying the topic and a callback function

which will be run every time someone publishes a message onto the topic (Fig. 4 and 5).

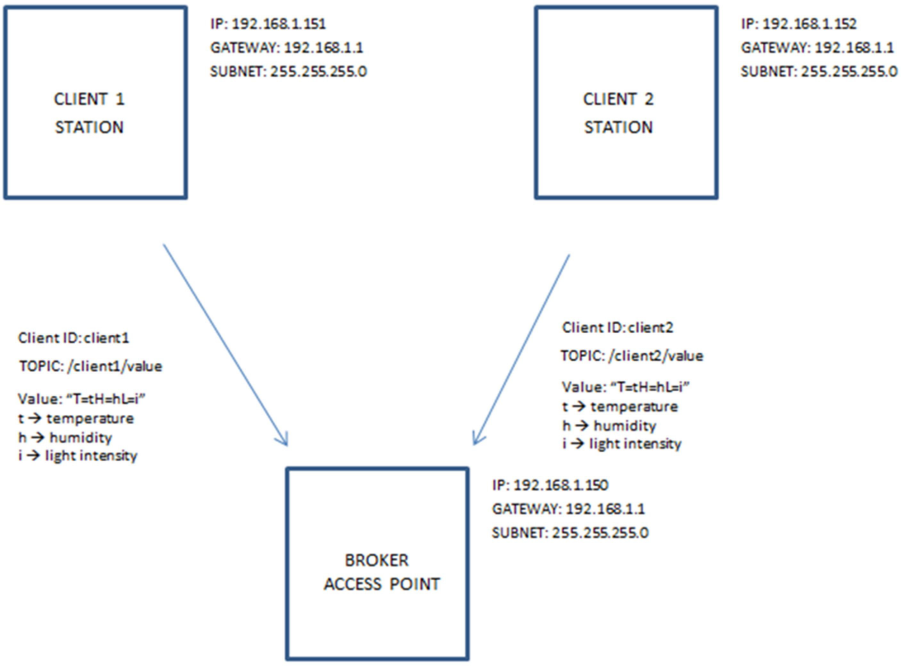

Fig. 4. Task 11. Boards Configuration.

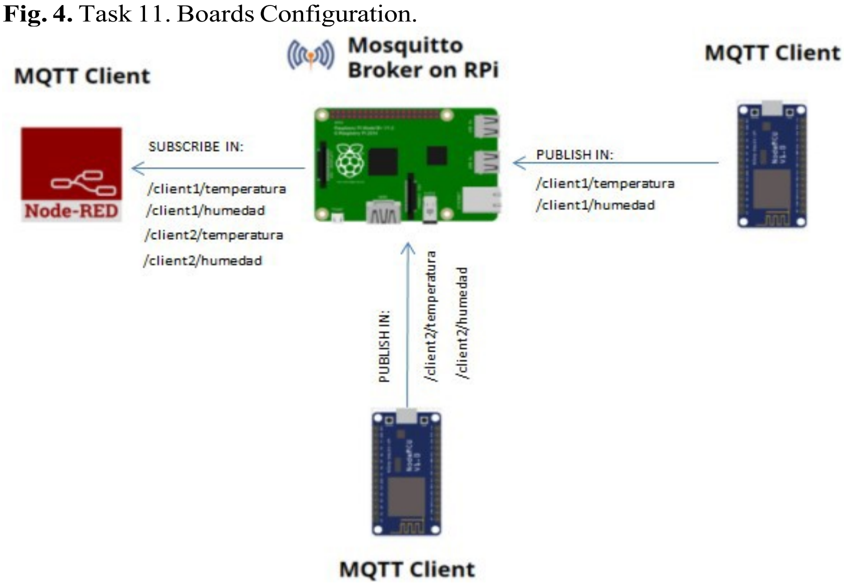

Fig 5. Boards Configuration.

5 Conclusions

The main goal of this paper was the development of a remote IoT Laboratory based on the Arduino language, a Raspberry PI as server and boards with ESP8266 micro-controllers with the aim of supporting MQTT remote programming of real devices.

Although the objectives have been met, some limitations have been identified that will lead to new developments. For example, the need of an easier maintenance and deployment system, maybe based on virtualization techniques, such as Docker. Also, a booking system is needed if the system is expected to be used in massive environments, such as MOOCs [6]. Currently, the system solves the concurrency through a queue (first to arrive, first to get the service). However, in advance scheduling is needed for its generalization and use in massive environments.

6 Acknowledgments

This publication is part of the In4Labs project with reference TED2021-131535B-I00 funded by MCIN/AEI/10.13039/501100011033 and by European Union “NextGenerationEU”/PRTR.

References

1. Sancristobal, E., et al M. Virtual and Remote Industrial Lab Integration in Learning Management Systems. IEEE Ind. Electronics Mag. Vol. 8, issue 4, 2014, pp. 45-58 (12).
2. Blazquez-Merino, M., et al Use of VISIR Remote Lab in Secondary School: Didactic Experience and Outcomes. Int. Conf. Remote Eng. and Virtual Instrum., pp. 69-79, 2018. DOI: 10.1007/978-3-319-95678-7_8.
3. Fernández-Pacheco, A., Martin, S. and Castro, M. Implementation of an Arduino remote laboratory with raspberry Pi. IEEE Global Eng. Educ. Conf., 2019, pp. 1415–1418. doi: 10.1109/EDUCON.2019.8725030.
4. Martin, S., Gomez, A., Castro, M.. (2023). Industry 4.0 remote lab based on PIC microcontrollers. DYNA, 98(4). 334. DOI: https://doi.org/10.6036/10925
5. Martin, S., Gomez, A., Castro, M.. (2023). Development of an iot remote laboratory based on PIC microcontrollers. DYNA New Technologies, 10(1). [10P.]. DOI: https://doi.org/10.6036/NT10828
6. Martin, S. et al., "Assessment and recognition in technical massive open on-line courses with and without on-line laboratories," 2023 IEEE Global Engineering Education Conference (EDUCON), Kuwait, Kuwait, 2023, pp. 1-4, doi: 10.1109/EDUCON54358.2023.10125261.