



Artículo

Diseño de un sistema de monitoreo de temperatura y humedad con IoT, y alertas en WhatsApp® mediante Node-RED

Norma-Alonso ¹, Yuridia Hernández² Tonatiuh Meneses ³

¹ Afiliación 1; Tecnológico Nacional de México Campus Huichapan, Ingeniería Industrial, México

² Afiliación 2; Tecnológico Nacional de México Campus Huichapan, Ingeniería Industrial, México

³ Afiliación 3; Tecnológico Nacional de México Campus Huichapan, Ingeniería Industrial, México

* Correspondencia: nlalonso@iteshu.edu.mx

Resumen: Se presenta un sistema de IoT económico para supervisar continuamente la temperatura y la humedad del entorno, así como su diseño y funcionamiento. El sistema emplea una tarjeta ESP32 conectada a un sensor DHT11, la plataforma Node-Red para procesar los datos y emitir alertas por medio de WhatsApp mediante la API CallMeBot, además del protocolo MQTT con bróker público HiveMQ. Cuando las condiciones de humedad y temperatura predefinidas se exceden, se reciben inmediatamente en un teléfono móvil las notificaciones, y cada 60 segundos se envían los datos. El sistema se ha diseñado pensando en aplicaciones educativas o de pequeñas empresas, además de contextos de la Industria 4.0, y tiene capacidad de escalar.

Palabras clave: WhatsApp®, MQTT, Node-RED, IoT, monitoreo ambiental e Industria 4.01.

Introducción

En años recientes, el Internet de las Cosas (IoT) ha transformado la manera de recolectar, procesar y transmitir información sobre lo que nos rodea. El fundamento de este modelo tecnológico es la conexión de dispositivos físicos que son capaces de recopilar datos y enviarlos por medio de redes digitales, lo cual posibilita su monitoreo y seguimiento en tiempo real (Atzori, Iera & Morabito, 2017). Su implementación se ha ampliado a diversos sectores, desde la gestión del uso adecuado de la energía y la defensa del medio ambiente hasta la automatización industrial. Esto es fundamental para tomar decisiones fundamentadas en datos (Al-Fuqaha et al., 2015). En este escenario, es crucial monitorear las condiciones ambientales, como la temperatura y la humedad, para garantizar que los procesos de producción permanezcan estables, los materiales no se deterioren y el confort en las zonas habitadas. Hoy en día, estas aplicaciones pueden usarse para asuntos académicos y pequeñas empresas debido a que existen soluciones apropiadas que emplean sensores como el DHT11 y tarjetas programables como la ESP32 (Ooi et al., 2021).

Asimismo, las plataformas de integración visual son más confiables y fáciles, como Node-RED®, pues simplifican el desarrollo de sistemas IoT al permitir la conexión modular entre nodos de procesamiento, entrada y salida. Esta herramienta reduce la complejidad de la programación tradicional al facilitar la comunicación entre bases de datos, servicios de mensajería en la nube y sensores. El propósito de este proyecto es utilizar Node-RED® para enlazar la aplicación de mensajería WhatsApp® con el microcontrolador ESP32, lo que permitirá recibir alertas automáticas en caso de que los niveles de humedad o temperatura excedan los límites previamente fijados. El objetivo es desarrollar un sistema de monitoreo remoto que sea confiable y adaptativo para las condiciones medioambientales en tiempo real. Por lo tanto, con el fin de que las pequeñas empresas e instituciones educativas tengan la posibilidad de utilizarlas, se pretende brindar soluciones inteligentes y sostenibles a través de tecnologías IoT debido a su bajo costo (García &

Citar este trabajo: Alonso, N.; Hernández, Y.; Meneses, T. *Diseño de un sistema de monitoreo de temperatura y humedad con IoT y alertas en WhatsApp® mediante Node-RED. RELITEC'S 2025, 8va, edición*

Recibido: 30/10/2025

Aceptado: 06/11/2025

Publicado: 20/11/2025

López, 2022). El objetivo principal de esta investigación es crear un sistema IoT que pueda monitorear la temperatura y la humedad en tiempo real, utilizando una tarjeta ESP32 y Node-Red para enviar notificaciones a un teléfono celular mediante WhatsApp.

2. Materiales y Métodos

Un sistema de dispositivos inteligentes conectados, común en IoT, opera por medio de la recopilación e intercambio de datos en tiempo real. Estos sistemas están formados por dispositivos inteligentes que tienen la capacidad de recolectar datos y transmitirlos a aplicaciones de IoT mediante Internet. Estas aplicaciones fusionan software y servicios que hacen más sencillo el análisis, la administración y el procesamiento de datos que provienen de diversos dispositivos conectados. Por último, gracias a una aplicación, los dispositivos pueden ser controlados y supervisados, lo cual hace más sencillo su monitoreo y gestión remota (Amazon Web Services, 2021).

2.1. Arquitectura recomendada

El modelo experimental, que consiste en un sensor DHT11 montado sobre un microcontrolador ESP32, permite la recolección de variables, específicamente la temperatura y la humedad. Esto puede verse en la figura 1 de la arquitectura sugerida. La programación y el procesamiento del sistema se realizaron con el apoyo de Thonny®, lo que facilitó la implementación y administración del código (Schmidt, s.f.).



Figura 1. Arquitectura propuesta. Fuente: Elaboración propia.

2.2 Descripción de los dispositivos inteligentes

Sensor DHT11

Un dispositivo de temperatura y humedad, el DHT11, que se muestra en la Figura 2. Es un dispositivo periférico económico que tiene la habilidad de calcular el porcentaje de humedad en el aire y la temperatura, dentro del intervalo del 20 % al 90 % HR, con un margen de error de ± 5 % HR. La temperatura, con un rango que va desde 0 hasta 50 °C y una precisión de ± 2 °C, se vuelve entonces una alternativa económica para aplicaciones de monitoreo básico (Gay, 2018).



Figura 2. Sensor DHT11 utilizado en el sistema de monitoreo. Fotografía original del autor (2025), basada en la descripción técnica de Gay (2018) y Aosong Electronics Co., Ltd. (2023).

ESP32

El ESP32 se muestra en la Figura 3 es un microcontrolador de bajo costo y bajo consumo energético, equipado con conectividad Wi-Fi y Bluetooth integrada. Esta electrónica especializada no solo permite la comunicación mundial, sino también el envío de informes de estado, la ejecución de tareas y la recepción de órdenes desde o hacia cualquier parte del mundo (Cameron, 2023; Gopal et al., 2020).



Figura 3. Microcontrolador ESP32 empleado para la transmisión de datos en el sistema IoT. Fotografía original del autor (2025), basada en la descripción técnica de Cameron (2023).

2.3. Descripción de la aplicación IoT Conexión física

La Figura 4 muestra la conexión de VCC a 3V3, Data a GPIO 15 y GND a GND para la obtención de los datos de humedad y temperatura (Schmidt, s.f.).

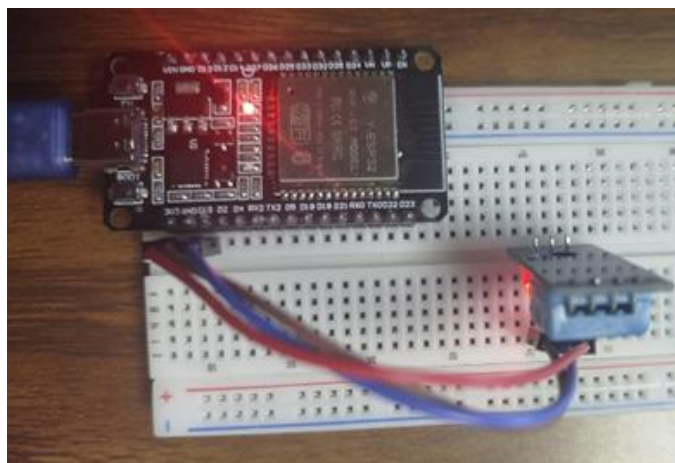


Figura 4. Conexión física. Fotografía original del autor (2025)

2.3 Software Thonny®

Utilizando el software Thonny, se creó un código para la lectura de los datos de temperatura y humedad a través de GPIO 15 del microcontrolador ESP32 en un intervalo de 60 segundos, como se aprecia en la Figura 5 (Schmidt, s.f.).

```

dht11_esp32_articulo.py
1 import network
2 import time
3 from umqtt.simple import MQTTClient
4 from machine import Pin, reset
5 import dht
6
7 WIFI_SSID = "prueba"
8 WIFI_PASSWORD = "iteshu2025"
9 WIFI_MAX_RETRY = 10
10
11
12 MQTT_BROKER = "broker.hivemq.com"
13 MQTT_PORT = 1883
14 MQTT_TOPIC_TEMP = b"temperatura"
15 MQTT_TOPIC_HUM = b"humedad"
16 CLIENT_ID = b"ESP32_DHT11_v1"
17 MQTT_KEEP_ALIVE = 60
18
19 try:
20     sensor = dht.DHT11(Pin(15))
21 except ValueError:
22     print("Error: Pin 15 no válido. Revisa tu hardware.")
23
24

```

Console

```

11:23 | % Temp: 23°C | Hum: 40% |
11:24 | % Temp: 23°C | Hum: 40% |
11:25 | % Temp: 23°C | Hum: 39% |
11:26 | % Temp: 23°C | Hum: 39% |
11:27 | % Temp: 23°C | Hum: 39% |
11:28 | % Temp: 23°C | Hum: 39% |
11:29 | % Temp: 24°C | Hum: 66% |
11:30 | % Temp: 23°C | Hum: 45% |
11:31 | % Temp: 23°C | Hum: 43% |
11:32 | % Temp: 23°C | Hum: 43% |
11:33 | % Temp: 23°C | Hum: 41% |
11:34 | % Temp: 23°C | Hum: 41% |
11:35 | % Temp: 23°C | Hum: 40% |
11:36 | % Temp: 23°C | Hum: 39% |
11:37 | % Temp: 23°C | Hum: 41% |
11:38 | % Temp: 23°C | Hum: 39% |
11:39 | % Temp: 23°C | Hum: 39% |

```

MicroPython (ESP32) • CP2102 USB to UART Bridge Controller @ COM2

Figura 5. Código y monitoreo en Thonny. Fuente: Elaboración propia.(2025)

Una vez obtenidos los datos en el software Thonny, es necesario enviarlos a la plataforma Node-RED, mediante los tópicos “temperatura” y “humedad”, utilizando el protocolo de comunicación MQTT a través del broker público.” *hivemq*” en el puerto 1883(Node-RED, 2025).

```

dht11_esp32_articulo.py
1 import network
2 import time
3 from umqtt.simple import MQTTClient
4 from machine import Pin, reset
5 import dht
6
7 WIFI_SSID = "prueba"
8 WIFI_PASSWORD = "iteshu2025"
9 WIFI_MAX_RETRY = 10
10
11
12 MQTT_BROKER = "broker.hivemq.com"
13 MQTT_PORT = 1883
14 MQTT_TOPIC_TEMP = b"temperatura"
15 MQTT_TOPIC_HUM = b"humedad"
16 CLIENT_ID = b"ESP32_DHT11_v1"
17 MQTT_KEEP_ALIVE = 60
18
19 try:
20     sensor = dht.DHT11(Pin(15))
21 except ValueError:
22     print("Error: Pin 15 no válido. Revisa tu hardware.")
23
24

```

Console

```

11:25 | % Temp: 23°C | Hum: 39% |
11:26 | % Temp: 23°C | Hum: 39% |
11:27 | % Temp: 23°C | Hum: 39% |
11:28 | % Temp: 23°C | Hum: 39% |
11:29 | % Temp: 24°C | Hum: 66% |
11:30 | % Temp: 23°C | Hum: 45% |
11:31 | % Temp: 23°C | Hum: 43% |
11:32 | % Temp: 23°C | Hum: 43% |
11:33 | % Temp: 23°C | Hum: 41% |
11:34 | % Temp: 23°C | Hum: 41% |
11:35 | % Temp: 23°C | Hum: 40% |
11:36 | % Temp: 23°C | Hum: 39% |
11:37 | % Temp: 23°C | Hum: 41% |
11:38 | % Temp: 23°C | Hum: 39% |
11:39 | % Temp: 23°C | Hum: 39% |
11:40 | % Temp: 23°C | Hum: 39% |

```

MicroPython (ESP32) • CP2102 USB to UART Bridge Controller @ COM2

Node-RED

Flow 1

```

graph LR
    Temperatura --> debug4[debug 4]
    Humedad --> debug5[debug 5]

```

debug

```

29/10/2025, 11:36:43 node debug 4
temperatura : msg.payload : number
23
29/10/2025, 11:36:44 node debug 5
humedad : msg.payload : number
39
29/10/2025, 11:37:43 node debug 4
temperatura : msg.payload : number
23
29/10/2025, 11:37:44 node debug 5
humedad : msg.payload : number
41
29/10/2025, 11:38:44 node debug 5
humedad : msg.payload : number
39
29/10/2025, 11:38:44 node debug 4
temperatura : msg.payload : number
23
29/10/2025, 11:39:44 node debug 5
humedad : msg.payload : number
39
29/10/2025, 11:40:43 node debug 4
temperatura : msg.payload : number
23
29/10/2025, 11:40:43 node debug 5
humedad : msg.payload : number
39

```

Figura 6. Recepción de datos en Node-RED. Fuente: Elaboración propia (2025).

3. Resultados

El monitoreo permanente de la humedad y la temperatura es viable mediante el uso de WhatsApp, el sensor DHT11, el protocolo MQTT, la plataforma Node-RED y el microcontrolador ESP32. Las lecturas se transmiten cada minuto, lo que posibilita la detección de modificaciones relevantes y su actualización directa en el teléfono móvil del usuario. Esto posibilita la identificación temprana de cambios drásticos en las condiciones del medio

ambiente y el monitoreo desde sitios lejanos (Gopal et al., 2020; Amazon Web Services, 2021) (véase la figura 6).

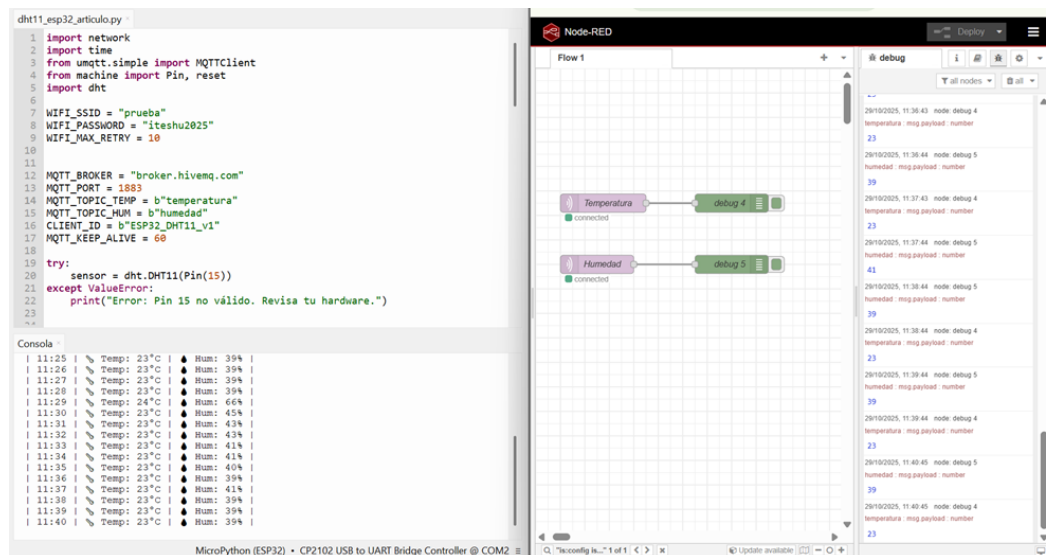


Figura 6. Recepción de datos en Node-RED. Fuente: Elaboración propia. (2025)

La Figura 7 ilustra la creación de un flujo en Node-RED que emplea el servicio de CallMe-Bot API. El nodo de función se utilizó para manejar, guardar y combinar los datos con el número de teléfono y la clave API. Para transmitir la información a la aplicación WhatsApp de forma automática, se empleó un nodo de solicitud HTTP (Node-RED, 2025).

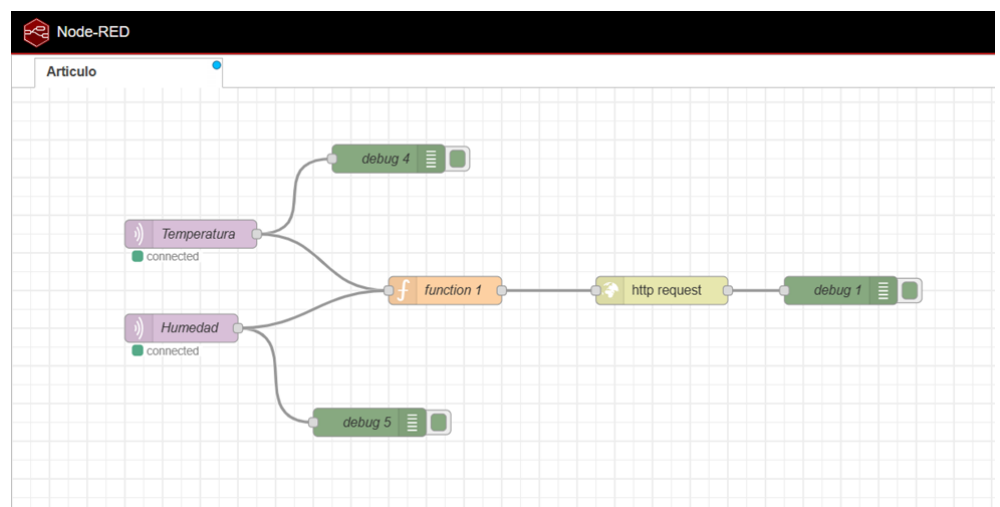


Figura 7. Flujo de integración en Node-RED. Fuente: Elaboración propia. (2025)

Los datos sobre la temperatura y la humedad se obtienen automáticamente por medio de mensajes de texto en la aplicación WhatsApp (véase la figura 8). Este procedimiento permite la visualización de los datos en el equipo móvil del usuario (Node-RED)

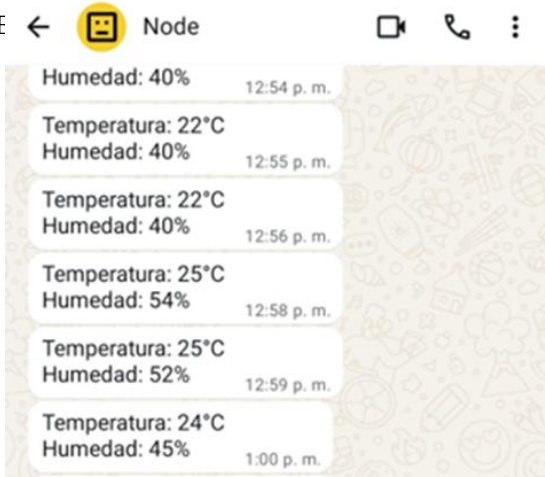


Figura 8. Datos mostrados en WhatsApp. Fuente: Elaboración propia. (2025)

En la figura 9 se muestra la monitorización de un dato específico que se ve replicado tanto en la consola del IDE Arduino-ES32, en Node-Red y en la Pantalla del celular y la aplicación de WhatsApp

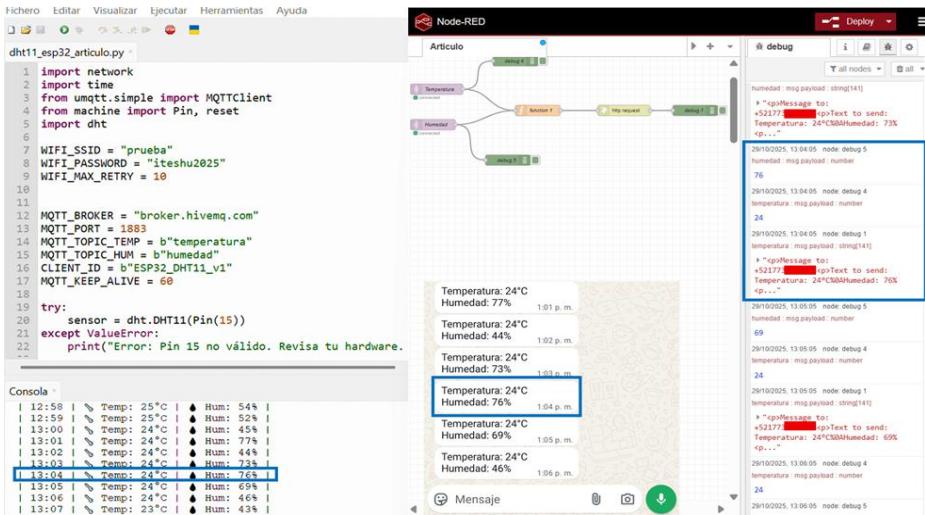


Figura 9. Monitorización de un dato específico. Fuente: Elaboración propia. (2025)

Tabla 1 Resultados representativos del monitoreo del sistema. Fuente elaboración propia (2025)

HORA	TEMPERATURA °C	HUMEDAD %
12:58	25	54
12:59	25	52
13:00	24	45
13:01	24	77
13:02	24	44
13:03	24	73
13:04	24	76

En los resultados también se pudo comprobar que el tiempo de entrega del mensaje a WhatsApp fue inferior a 5 segundos en todas las pruebas.

4. Discusión

El sistema propuesto demuestra que es posible, desde un punto de vista técnico y funcional, implementar tecnologías IoT para la supervisión ininterrumpida de la humedad y la temperatura. Para ello se utiliza Node-RED como plataforma de procesamiento de datos, así como el microcontrolador ESP32 y el sensor DHT11. La comunicación continua y estable, obtenida mediante el protocolo MQTT, está en línea con lo que Gopal et al. (2020) sostuvieron, enfatizando que la arquitectura basada en este protocolo es especialmente eficaz en sistemas de vigilancia medioambiental por su alta fiabilidad de transmisión de datos y su escaso uso de recursos.

De acuerdo con investigaciones anteriores sobre los sensores DHT11 (Gay, 2018), la información adquirida se guarda en el sistema y puede detectar cambios de temperatura y humedad con una precisión adecuada para usos simples de control ambiental. Aunque tiene limitaciones si se lo compara con sensores más completos, su fácil instalación y su costo bajo lo convierten en una alternativa práctica y educativa para desarrollar proyectos de automatización e ingeniería aplicada.

En cuanto a la visualización y procesamiento de datos, se comprobó que la plataforma Node-RED es una alternativa muy efectiva para gestionar flujos de información y crear avisos automatizados. De acuerdo con Node-RED (2025), su estructura nodal posibilita la comunicación entre diversos sistemas y la integración de servicios externos, por ejemplo, el envío de notificaciones mediante chatbots o aplicaciones de mensajería. Esta visión es coherente con los fundamentos de la Industria 4.0, en la que la automatización y la conectividad son elementos fundamentales para tomar decisiones rápidas y adecuadas (UNIR, 2022).

Las restricciones del sistema comprenden: la latencia que puede aparecer al mandar mensajes por WhatsApp en redes saturadas, la necesidad de una conexión estable a través de Wi-Fi y la precisión promedio del sensor DHT11 ($\pm 2^\circ\text{C}$ y $\pm 5\%$ HR).

5. Conclusiones

La arquitectura desarrollada muestra que un sistema IoT puede realizar un seguimiento permanente de la temperatura y la humedad mediante el uso del microcontrolador ESP32 y el periférico DHT11, con la plataforma Node-RED como medio para administrar los datos. El sistema no solamente almacena datos, sino que adicionalmente detecta valores fuera de rango, garantizando así una supervisión fiable. La aplicación WhatsApp® les da seguimiento a los datos, mejorando la capacidad de respuesta ante situaciones críticas. Esta solución económica y escalable se puede implementar en cualquier situación que necesite una monitorización en tiempo real de variables ambientales (UNIR, 2022).

Contribución: "Conceptualización: Norma Alonso y Yuridia. González; metodología: Tonatiuh Meneses; software: Norma Alonso; validación: Tonatiuh Meneses.

Financiamiento: "Esta investigación no recibió financiamiento externo"

Agradecimientos: Al ITESHU® por el uso de las instalaciones y laboratorios de Ingeniería industrial

Conflicto de interés: Declare conflictos de interés o establezca "Los autores declaran no tener conflicto de intereses"." Los financiadores no tuvieron ningún rol en el diseño del estudio; en la recopilación, análisis o interpretación de datos; en la redacción del manuscrito, o en la decisión de publicar los resultados.

Referencias

1. Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of Things: A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys & Tutorials*, 17(4), 2347–2376. <https://doi.org/10.1109/COMST.2015.2444095>
2. Amazon Web Services. (2021). ¿Qué es IoT? — Explicación del Internet de las cosas. Recuperado de <https://aws.amazon.com/es/what-is/iot/>
3. Aosong Electronics Co., Ltd. (2023). DHT11 Temperature & Humidity Sensor Datasheet.
4. Atzori, L., Iera, A., & Morabito, G. (2017). Understanding the Internet of Things: Definition, potentials, and societal role of a fast evolving paradigm. *Ad Hoc Networks*, 56, 122–140. <https://doi.org/10.1016/j.adhoc.2016.12.004>
5. Cameron, N. (2023). Microcontrolador ESP32. En *Formatos y comunicación ESP32 (Serie Maker Innovations)*. Apress. https://doi.org/10.1007/978-1-4842-9376-8_1
6. García, P., & López, M. (2022). Integración de plataformas IoT y sistemas de mensajería instantánea para monitoreo remoto. *Revista Iberoamericana de Tecnología e Innovación*, 8(2), 45–58.
7. Gay, W. (2018). Sensor DHT11. En *Advanced Raspberry Pi*. Apress. https://doi.org/10.1007/978-1-4842-3948-3_22
8. Gopal, M., Prakash, T. C., Ramakrishna, N. V., & Yadav, B. P. (2020, diciembre). IoT based solar power monitoring system. *IOP Conference Series: Materials Science and Engineering*, 981(3), 032037. IOP Publishing. <https://doi.org/10.1088/1757-899X/981/3/032037>
9. IBM. (2022). Node-RED: Flow-based programming for the Internet of Things. IBM Developer.
10. Node-RED. (2025). node-red-contrib-whatsapp-cmb. Recuperado de <https://flows.node-red.org/node/node-red-contrib-whatsapp-cmb>
11. Ooi, C. P., Htike, Z. Z., & Lim, C. P. (2021). An overview of Node-RED and its applications in the Internet of Things. *Journal of Emerging Trends in Computing and Information Sciences*, 12(3), 101–110.
12. Schmidt, D. (s.f.). Todo Arduino: Programación fácil. Recuperado de https://www.google.com.mx/books/edition/Todo_Arduino/oT0EEQAAQBAJ
13. UNIR. (2022, 22 de agosto). ¿Qué es el IoT Industrial y qué importancia tiene su uso en la Industria 4.0? *Revista UNIR*. Recuperado de <https://www.unir.net/revista/ingenieria/iot-industrial/>