

	GESTIÓN DE SERVICIOS ACADÉMICOS Y BIBLIOTECARIOS		CÓDIGO	FO-GS-15	
			VERSIÓN	02	
	ESQUEMA HOJA DE RESUMEN			FECHA	03/04/2017
				PÁGINA	1 de 1
ELABORÓ		REVISÓ	APROBÓ		
Jefe División de Biblioteca		Equipo Operativo de Calidad	Líder de Calidad		

RESUMEN TRABAJO DE GRADO

AUTOR(ES): NOMBRES Y APELLIDOS COMPLETOS

NOMBRE(S): JULIAN DAVID APELLIDOS: MOLINA VARGAS

FACULTAD: INGENIERÍA

PLAN DE ESTUDIOS: INGENIERÍA ELECTRÓNICA

DIRECTOR:

NOMBRE(S): ELMER ALEJANDRO APELLIDOS: PARADA PRIETO

NOMBRE(S): JHAIR MAURICIO APELLIDOS: ARENAS DOMÍNGUEZ

TÍTULO DEL TRABAJO (TESIS): DESARROLLO DE UN SISTEMA DE VISUALIZACIÓN Y MONITOREO DE ENTORNOS DE REFRIGERACIÓN BASADO EN TECNOLOGÍAS IOT PARA LA EMPRESA VHM INGENIERÍA.

RESUMEN:

El presente proyecto se basó en un sistema de visualización y monitoreo para las variables de temperatura y humedad en entornos de refrigeración basado en tecnologías IoT. Para ello, se implementó un tipo de investigación aplicada, dado que, al realizar el sistema de monitoreo y visualización se implementa lo investigado previamente. Para la construcción del sistema primero se recopiló información sobre las tecnologías de IoT, después se realizó el diseño del sistema, el cual está compuesto por hardware y software, seguidamente se desarrolló la programación y se implementó las conexiones desde los sensores de temperatura y humedad a la tarjeta ESP8266, el cual recibe los datos de temperatura, humedad, fecha y hora, finalmente, se realizó el aplicativo web, donde se observan los datos adquiridos en tiempo real y en forma gráfica, adicional se hicieron pruebas de uso en diferentes entornos.

PALABRAS CLAVES: IOT, MONITOREO, TEMPERATURA, ESP8266, ENTORNOS

CARACTERÍSTICAS:

PÁGINAS: 68

PLANOS: 0

ILUSTRACIONES: 41

CD ROOM: 0

DESARROLLO DE UN SISTEMA DE VISUALIZACIÓN Y MONITOREO DE
ENTORNOS DE REFRIGERACIÓN BASADO EN TECNOLOGÍAS IOT PARA LA
EMPRESA VHM INGENIERÍA

JULIÁN DAVID MOLINA VARGAS

UNIVERSIDAD FRANCISCO DE PAULA SANTANDER

FACULTAD DE INGENIERÍA

PLAN DE ESTUDIOS DE INGENIERÍA ELECTRÓNICA

SAN JOSÉ DE CÚCUTA

2022

DESARROLLO DE UN SISTEMA DE VISUALIZACIÓN Y MONITOREO DE
ENTORNOS DE REFRIGERACIÓN BASADO EN TECNOLOGÍAS IOT PARA LA
EMPRESA VHM INGENIERÍA

JULIÁN DAVID MOLINA VARGAS

DIRECTOR:

MSc. IE. ELMER ALEJANDRO PARADA P.

CODIRECTOR

ING. JHAIR MAURICIO ARENAS DOMÍNGUEZ

UNIVERSIDAD FRANCISCO DE PAULA SANTANDER

FACULTAD DE INGENIERÍA

PLAN DE ESTUDIOS DE INGENIERÍA ELECTRÓNICA

SAN JOSÉ DE CÚCUTA

2022

ACTA DE SUSTENTACIÓN DE UN TRABAJO DE GRADO

Fecha: CÚCUTA, 23 DE JUNIO DE 2022

Hora: 16:00

Lugar: SALON SC302

Plan de Estudios: INGENIERÍA ELECTRÓNICA

Título de la Tesis: "DESARROLLO DE UN SISTEMA DE VISUALIZACIÓN Y MONITOREO DE ENTORNOS DE REFRIGERACIÓN BASADO EN TECNOLOGÍAS IOT PARA LA EMPRESA VHM INGENIERÍA"

Jurados: JULIAN ORLANDO TARAZONA ANTELIZ IE.MSc
LUIS FERNANDO BUSTOS MARQUEZ IE.ESp

Director: ELMER ALEJANDRO PARADA, IE MSc
Codirector: JHAIR MAURICIO ARENAS DOMÍNGUEZ, IE

Nombre del Estudiante:	Código:	Calificación:
JULIÁN DAVID MOLINA VARGAS	1161481	4,1 (Cuatro, Uno)

APROBADA


JULIAN ORLANDO TARAZONA ANTELIZ


LUIS FERNANDO BUSTOS MARQUEZ


ANGELO JOSEPH SOTO VERGEL
Coordinador Comité Curricular
Ingeniería Electrónica

Tabla de contenido

	Pág.
Introducción	10
1. Descripción del problema	12
2. Justificación	14
2.1 Beneficios tecnológicos	14
2.2 Beneficios económicos	14
2.3 Beneficios sociales	15
3. Alcance	16
3.1 Tipo de proyecto	16
3.2 Resultados esperados	16
4. Limitaciones y delimitaciones	17
4.1 Limitaciones	17
4.2 Delimitaciones	17
4.2.1 Espacial	17
4.2.2 Temporal	17
4.2.3 Elementos	17
5. Objetivos	18
5.1. General	18
5.2. Específicos	18
6. Marco referencial	19
6.1. Antecedentes	19
6.2. Marco Teórico	21

6.3. Marco legal	28
7. Diseño metodológico	30
8. Resultados	34
8.1. Análisis de requerimientos	34
8.2. Implementación del bróker MQTT	36
8.3. Conexión placa NodeMCU(ESP8266)	40
8.4. Conexión Base de datos	43
8.5. Diseño de la interfaz web	45
8.6. Realización del diseño en PCB y Modelo 3D	48
8.7. Implementación del sistema con pruebas	52
8.8. Sistema final para producción	58
9. Conclusiones	60
Referencias	61
Anexos	64

Lista de figuras

	Pág.
Figura 1. Diagrama de comunicación MQTT.	22
Figura 2 Sensor DS18B20.	23
Figura 3. Sensor Am2315.	24
Figura 4. Modulo Wifi Nodemcu Amica Esp8266.	25
Figura 5. Lector De Memoria Micro SD.	26
Figura 6. Logo de Node js.	27
Figura 7. Logo de Vue js.	27
Figura 8. Representación gráfica Topología del sistema IoT.	30
Figura 9. Diagrama de flujo del sistema de monitoreo y visualización.	34
Figura 10. Diagrama del diseño del sistema de visualización y monitoreo.	35
Figura 11. Broker RabbitMQ.	37
Figura 12 Resumen del total de datos leídos por RabbitMQ.	38
Figura 13. RabbitMQ datos enviados por la tarjeta NodeMCU.	39
Figura 14. AWS servicio IoT core MQTT.	40
Figura 15. AUTODESK EAGLE.	40
Figura 16. Diagrama circuito esquemático en EAGLE.	41
Figura 17. Diagrama circuito PCB en EAGLE.	42
Figura 18. Diagrama circuito PCB en Fritzing.	43
Figura 19. Base de Datos MongoDB compass con dos dispositivos.	44

Figura 20. Base de Datos MongoDB compass overview.	45
Figura 21. Login VHM Ingeniería.	45
Figura 22. Flujo de la aplicación web con Node-RED.	46
Figura 23. Flujo del Login con Node-RED.	46
Figura 24 . Visualización y monitoreo del sensor de Cúcuta.	47
Figura 25. Visualización y monitoreo del sensor de Manizales.	48
Figura 26. Diseño PCB en Fusión 360.	49
Figura 27. Diagrama de archivos Gerber.	50
Figura 28. PCB en 3D con sus componentes.	50
Figura 29. Modelo 3D con la PCB expuesta.	51
Figura 30. Modelo 3D con tapa cerrada.	51
Figura 31. PCB PROTOTYPE MACHINE.	52
Figura 32. Configuración de los archivos Gerber.	52
Figura 33. Diagrama con cálculos en PCB PROTOTYPE.	53
Figura 34. Máquina EP-2006H realizando el PCB.	54
Figura 35. Máquina EP-2006H con el PCB.	55
Figura 36. Primera versión del PCB con sus componentes.	56
Figura 37. Caja contenedora de la PCB.	57
Figura 38. Segunda versión del PCB con sus componentes	57
Figura 39. Dispositivo físico del sistema sin armar.	58
Figura 40. Dispositivo físico del sistema armado.	59
Figura 41. Dispositivo físico con la alimentación y el sensor conectados.	59

Lista de anexos

	Pág.
Anexo 1. Certificación Platzi en el curso BÁSICO DE JAVASCRIPT	64
Anexo 2. Certificación Platzi en el curso PRÁCTICO DE JAVASCRIPT	65
Anexo 3. Certificación Platzi en el curso BÁSICO DE PROGRAMACIÓN EN GO	66
Anexo 4. Certificación Platzi en el curso ASINCRONISMO CON JAVASCRIPT	67
Anexo 5. Certificación Platzi en el curso de INTRODUCCIÓN A AWS: FUNDAMENTOS DEL CLOUD COMPUTING	68

Introducción

El segmento de servicios de salud ha evolucionado en los últimos años con una tendencia marcada a la incorporación de los avances de la electrónica moderna. La tecnología avanzó rápidamente en esa área, permitiendo ofrecer nuevos tratamientos y mejorar las condiciones de vida de millones de personas. En muchos casos, la refrigeración y el acondicionamiento del aire se han vuelto recursos indispensables para que los procedimientos se realicen y para que se mantengan los más diversos materiales en las condiciones ideales. Existe, por lo tanto, una fuerte demanda por servicios de mantenimiento de los equipos que proporcionan temperaturas adecuadas a los distintos ambientes y los productos utilizados en instituciones de salud. Existen, por ejemplo, diversos tipos de medicamentos que se necesita mantener bajo refrigeración. Lo mismo ocurre con vacunas, sueros, bolsos de sangre. Cuando esos productos se deterioran por falta de almacenamiento adecuado, el detrimento financiero es grande, debido a su alto coste. Adicionalmente, su falta puede generar consecuencias serias para la salud de quienes necesitaran recibir tratamientos con medicamentos a los que no se les ha conservado la cadena de frío. (*Club de La Refrigeracion, 2020*)

Uno de estos avances de la electrónica moderna es el desarrollo del Internet de las Cosas (IoT), que consiste en una infraestructura global para la sociedad, permitiendo servicios sofisticados, mediante la conexión de objetos (cosas) tanto física como virtualmente, basados en la tecnología actual de intercambio de información, su desarrollo y tecnología de comunicación.

Los dispositivos IoT se conectan con un proceso llamado M2M (machine to machine, o máquina a máquina) en el que dos dispositivos o máquinas cualesquiera se comunican entre sí utilizando cualquier tipo de conectividad (que puede ser cable, WiFi, Bluetooth, etc.), haciendo su trabajo sin la necesidad de que un humano intervenga. Esto se realiza necesariamente con sensores y chips

específicos que están dispuestos en los dispositivos, estos dispositivos conectados generan una gran cantidad de datos que llegan a una plataforma IoT que recolecta, procesa y analiza dicha información. Este proceso resulta relevante para el usuario porque gracias a él se pueden sacar conclusiones de los hábitos y preferencias del mismo, así como para facilitarle la vida. (Rodrigo, 2021)

Por lo tanto, para garantizar las mejores condiciones de temperatura y humedad en el entorno donde se realizan los tratamientos en el área médica, es necesario desarrollar un sistema de monitoreo de entornos de refrigeración, el cuál ayude a controlar las medidas óptimas para cada uno de los entornos que presentan los servicios de salud, para así poder conservar adecuadamente medicamentos, vacunas, suero, sangre, tejidos, órganos para trasplante, entre otros. Los equipos y sistemas de refrigeración y acondicionamiento del aire juegan un rol fundamental en esa tarea.

Este trabajo presenta un sistema de visualización y monitoreo de entornos de refrigeración, el cual cuenta con el módulo ESP8266, un sensor de temperatura DS18B20, conectado a la red local de wifi por medio del protocolo de comunicación MQTT(Message Queuing Telemetry Transport) y cuenta además con un bróker para almacenar los datos obtenidos por los sensores. VHM ingeniería como empresa enfocada en el sector salud busca brindar a sus clientes una herramienta basada en IoT mediante dispositivos que permiten el sensado de estas variables a fin de proporcionar a los clientes la información detallada de la temperatura y su comportamiento a través del tiempo por medio de un aplicativo web.

1. Descripción del problema

El monitoreo y control de la temperatura en el sector de la salud juegan un papel fundamental debido a que tanto los medicamentos, vacunas, bancos de sangre, hospitales, clínicas, laboratorios y otras instituciones dependen del mantenimiento de temperaturas adecuadas, para garantizar la conservación apropiada, todo esto requiere un sistema de monitoreo óptimo de manera constante para mantener su cadena de frío y no llegar a perjudicar el estado de los suministros.

VHM ingeniería como empresa enfocada en el sector salud busca brindar a sus clientes una herramienta basada en el internet de las cosas, la cual, mediante dispositivos embebidos permiten el sensado de estas variables a fin de proporcionar a los clientes la información detallada de la temperatura y su comportamiento a medida del tiempo en un aplicativo web.

El ministerio de protección social, resolución número 1403 de 2007 (14 de mayo) dice:

Los termolábiles deben almacenarse a temperaturas adecuadas, para evitar su descomposición con el calor, entre ellos se encuentran los productos biológicos, los que se recomienda almacenar a temperaturas de refrigeración entre 2°C y 8°C. En climas cálidos se recomienda disponer de aire climatizado para su conservación.

Contar con una área especial e independiente, debidamente dotada que ofrezca la privacidad y comodidad para el paciente y para quien aplique la prueba. Con adecuada iluminación y ventilación natural y/o artificial y su temperatura deberá estar entre 15-25°C. Este sitio podrá ser el mismo

utilizado para inyectología. (MINISTERIO DE LA PROTECCIÓN SOCIAL RESOLUCIÓN NÚMERO 1403 DE 2007, 2007)

En general, los medicamentos hormonales como insulina, levotiroxina y los anticonceptivos orales son especialmente sensibles a los cambios de temperatura. El conocimiento sobre el

deterioro que sufren los fármacos en condiciones extremas es limitado, dado que durante el desarrollo de los medicamentos solo se estudia su estabilidad en condiciones habituales de temperatura y humedad, asegurándose su calidad bajo estas condiciones. Se debe tener presente que no solo las altas temperaturas pueden afectar la estabilidad de los fármacos: por ejemplo, si la insulina se congela también pierde efectividad. (*El Mostrador*, 2020)

Dado a todo esto, es necesario contar con un sistema de monitoreo confiable y eficiente a la hora de visualizar el estado de temperatura y humedad en un entorno de refrigeración, es decir, en los sitios donde se requiera un monitoreo como, por ejemplo, hospitales, farmacias, IPS, entre otros, al no ser así, se pone en riesgo la salud de las personas por el mal estado de los medicamentos, vacunas, bancos de sangre, entre otros.

Debido a esta problemática, se genera la siguiente pregunta:

¿Cómo se puede mejorar el monitoreo de temperatura y humedad en los entornos de refrigeración como, hospitales, farmacias, IPS o sitios donde se requiera monitorear estas variables?

2. Justificación

En trabajos previos la empresa VHM Ingeniería ha desarrollado un sistema de visualización y monitoreo de entornos de refrigeración basado en tecnologías IoT, el cual cuenta con un prototipo basado en un módulo (ESP8266), con el protocolo de comunicación (HTTP) y un aplicativo web desarrollado con el lenguaje de programación PHP. Debido a esto se busca actualizar y optimizar el sistema migrando a nuevas tecnologías , las cuales son, node js, node red, MQTT(Message Queuing Telemetry Transport), por lo cual, este proyecto trae los siguientes beneficios:

2.1 Beneficios tecnológicos

- Se integraron nuevas tecnologías, node js, node-red, MQTT(Message Queuing Telemetry Transport), para la comunicación y el proceso de datos recolectados por la ESP8266.
- Se desarrolló un sistema capaz de almacenar datos de los sensores cuando no se encuentre conectado a internet.
- Se centralizó, procesó y envió los datos a un servidor, para el acceso a la información desde cualquier dispositivo con conexión a internet.

2.2 Beneficios económicos

Este proyecto trajo como beneficio económico para la empresa VHM Ingeniería, un producto, con el cual, la empresa podrá ampliar su portafolio de servicios y productos, ofreciendo un sistema más eficiente y rápido para los clientes, los cuales podrán visualizar y monitorear de una forma más óptima los datos en los entornos de refrigeración.

2.3 Beneficios sociales

Se produjo un impacto social positivo a todos los usuarios y clientes, dado que la importancia debido a medicamentos, vacunas, bancos de sangre, hospitales, clínicas, laboratorios y otras instituciones dependen del mantenimiento de temperaturas adecuadas, para garantizar la conservación apropiada de materiales sensibles que se utilizan en sus actividades, todo esto requiere un sistema de monitoreo óptimo de manera constante para mantener su cadena de frío y no llegar a perjudicar el estado de los suministros

Así mismo, ofrece una mejor forma para mantener el control en las áreas de refrigeración, el cual a su vez garantiza el estado adecuado de los medicamentos, vacunas, entre otros.

3 Alcance

3.1 Tipo de proyecto

El diseño y desarrollo del sistema de visualización y monitoreo en entornos de refrigeración , es un tipo de investigación aplicada, dado que, está encaminado al desarrollo de nuevos conocimientos basándose en los procedimientos y experimentos ya existentes por diferentes autores para ciertos puntos del desarrollo web, por eso es necesario tener una metodología exploratoria en el desarrollo del proyecto.

3.2 Resultados esperados

La investigación aplicada que se llevó acabo en el presente proyecto, dio como resultado un sistema de visualización y monitoreo en entornos de refrigeración, el cual, se puedan observar los datos de los sensores de humedad, temperatura, fecha y hora en el aplicativo web, también que almacena los datos en la micro SD cuando la tarjeta NodeMCU no se encuentre conectada a internet.

Con la implementación del protocolo de comunicación MQTT y las tecnologías node JS y vue js se consiguió una respuesta más óptima y rápida a la hora de visualizar y monitorear los sensores de los entornos de refrigeración.

4. Limitaciones y delimitaciones

4.1 Limitaciones

Para el desarrollo del proyecto es necesario contar con conocimientos en el desarrollo web y las tecnologías Node js y Vue js para cumplir con los requerimientos del cliente. El uso inadecuado, mal cuidado y manipulación indebida del sistema puede incurrir en problemas de lectura en el sensor AM2301 por parte del dispositivo, por lo que interfiere la obtención de datos del entorno de refrigeración así el aplicativo web.

4.2 Delimitaciones

4.2.1 Espacial

Este proyecto se desarrolló en las instalaciones de la empresa VHM ingeniería.

4.2.2 Temporal

El proyecto se desarrolló en 300 horas.

4.2.3 Elementos

Se implementó un módulo ESP8266 conectado por medio de wifi con el protocolo de comunicación MQTT, así mismo con un sensor de temperatura DS18B20 y el sensor Am2301, se integrará el módulo memoria micro SD para el almacenamiento de datos no enviados.

5. Objetivos

5.1. General

Desarrollar un sistema de visualización y monitoreo de entornos de refrigeración basado en tecnologías IoT para la empresa VHM ingeniería

5.2. Específicos

- Analizar el estado del arte de la implementación de nuevas tecnologías IoT, protocolos de comunicación, y software.
- Diseñar la estructura del sistema de visualización y monitoreo con los sensores.
- Realizar el prototipo del sistema de monitoreo con los sensores basado en tecnologías IoT y desarrollar el aplicativo web para la visualización de los datos registrados.
- Implementar el sistema de visualización y monitoreo en un entorno de refrigeración remoto.

6. Marco referencial

6.1. Antecedentes

(Oklilas, 2019), en su trabajo titulado “Sistema de control de la temperatura basado en el protocolo transporte de telemetría de colas de mensajes (MQTT)”

En este proyecto se trata de implementar el protocolo MQTT mediante la transferencia de datos consistentes en la temperatura y la humedad que se lleva a cabo en un experimento con las condiciones de la habitación. Algunos experimentos con condiciones frías, normales y calientes para determinar si el sensor puede funcionar bien o si se puede aplicar el protocolo MQTT. Los sensores de temperatura y humedad DHT 11 produjeron datos en forma de temperatura y humedad que se conectaron al nodo MCU como microcontrolador que tiene acceso a conectar los datos a internet usando una red Wi-Fi. (Oklilas, 2019)

(Jiménez Espinosa et al., 2010), en su trabajo titulado “Sistema de monitorización remota de la temperatura impulsado por MQTT para hogares inteligentes basados en IoT”.

Se presenta un sistema de monitorización remota de la temperatura para hogares inteligentes basados en IoT utilizando la comunicación impulsada por MQTT. Implementamos el sistema propuesto en MQTTBox para discutir sus diversos aspectos de rendimiento. Además, discutimos el estado del arte en este campo. También se presentamos aspectos de rendimiento, incluyendo la QoS y el tiempo publicado. (Jiménez Espinosa et al., 2010)

(Macheso et al., 2021) en su trabajo titulado: “Diseño de la casa inteligente ESP8266 usando MQTT y Node-RED”.

Este trabajo presenta el diseño de una solución de hogar inteligente utilizando en microcontrolador ESP8266 con MQTT y Node- RED se presenta. La casa inteligente controla las salidas de NodeMCU ESP8266 para la automatización del hogar y muestra los datos de los sensores en un tablero de Node-RED desde el nodo sensor hecho por la interfaz de NodeMCU ESP8266 y DHT 22 con el LED. En el diseño, el software Node-RED que se ejecuta en Raspberry Pi 3B+ y el protocolo de comunicación MQTT facilitan la comunicación entre el NodeMCU ESP8266 y el software node-RED. (Macheso et al., 2021)

(Eridani, n.d.), en su trabajo titulado. “Rendimiento del sistema de monitorización de sensores mediante Raspberry Pi a través del protocolo MQTT”

Este trabajo muestra el funcionamiento del sistema de monitorización de sensores construido en Raspberry Pi como implementación de Internet de las Cosas a través del protocolo MQTT. El enfoque principal es observar los datos leídos por los sensores que mostraron en el sistema de monitoreo a través del protocolo MQTT. El controlador utilizado en esta investigación es Raspberry Pi. El resultado muestra que el uso del parámetro QoS en la configuración MQTT es la clave del éxito de la transferencia de datos. (Eridani, n.d.)

(Hapsari et al., 2020), en su trabajo titulado: “Sistema de monitoreo de la calidad del aire interior con Node.js y aplicación MQTT”

En este proyecto de Internet de las cosas se construye el sistema de monitorización integrando

software y hardware con el dispositivo microcontrolador y el sensor de bajo coste MQ 135 y DHT 11. El sensor de nodo produce datos y los procesa con un microcontrolador. Los datos resultantes se transmiten inmediatamente a la nube a través de la red de Internet utilizando MQTT y se combinan con Node.js Clientes de código abierto MQTT utilizados para publicar mensajes y suscribirse a temas en corredores. La red de sensores inalámbricos también se aplica para que los nodos de sensores se puedan colocar en cualquier lugar y los datos se puedan adquirir en una pantalla de tablero. También se creó un panel de control basado en la web para facilitar el acceso a la información de la calidad del aire existente en diferentes habitaciones. (Hapsari et al., 2020)

6.2. Marco Teórico

En los siguientes ítems se presentan las definiciones y conceptos relacionados con los temas más relevantes en los que se fundamenta el proyecto.

MQTT: MQTT (Message Queue Telemetry Transport) es un protocolo ligero de publicación y suscripción utilizado sobre el protocolo TCP/IP. MQTT es abierto, sencillo y está diseñado para ser fácilmente implementados, estas características lo hacen ideal para su uso en muchas situaciones, incluyendo entornos limitados como la comunicación Máquina a Máquina (M2M) y el contexto del Internet de las Cosas (IoT) donde se necesitan códigos de huella pequeña y/o redes limitadas. El patrón de publicación-suscripción de mensajes requiere un broker de mensajes. Los brokers se encargan de distribuir los mensajes a los clientes interesados en función del tema del mensaje. (BorrowBits, 2021)

Un protocolo de publicación-suscripción permite que los mensajes se publiquen una vez y que varios consumidores (aplicaciones/dispositivos) proporcionen un desacoplamiento entre los productores y los consumidores para recibir los mensajes. Un productor envía un mensaje al tema

(sujeto). Un consumidor subvenciona un mensaje en un tema (sujeto). Un servidor de mensajes corredor de acuerdo a la publicación de suscribirse (suscribirse). en la Figura 1 se observa el diagrama de comunicación MQTT.

Bróker MQTT

Los Brokers MQTT tienen una dirección a la que pueden acceder el Publisher y el Subscriber. La tarea de este MQTT Broker es ser un enlace de transacción de datos entre el editor y el suscriptor. Los MQTT Brokers también reconocen los datos a través de una agrupación o comúnmente llamada topic. (Oklilas, 2019)

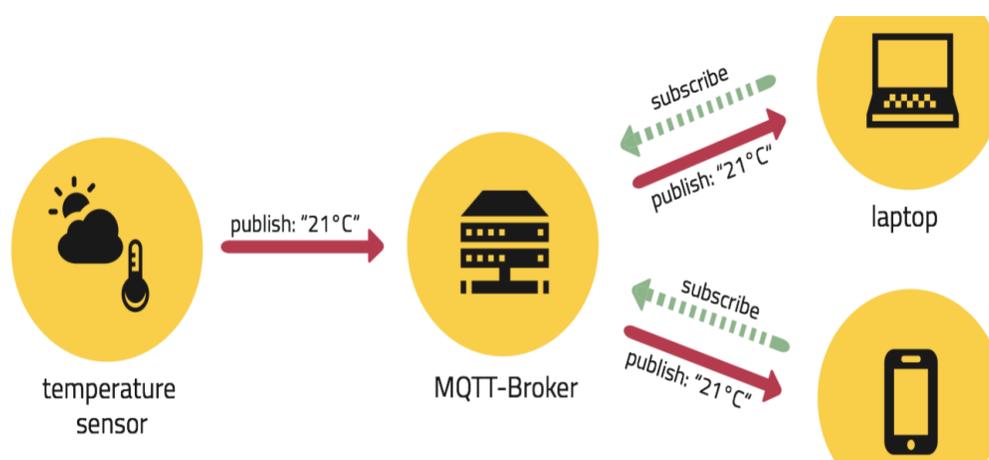


Figura 1. Diagrama de comunicación MQTT. (BorrowBits, 2021)

DS18B20 sensor de temperatura:

Sonda sumergible con sensor de temperatura DS18B20. (ver Figura 2) Es una sonda que permite tener una lectura digital de la temperatura, lo que traduce en lecturas más confiables ante cambios en el voltaje de alimentación. Ideal para utilizar en líquidos, a la intemperie o en cualquier ambiente donde no es posible que un circuito funcione adecuadamente. Es completamente compatible con Arduino, su protocolo de comunicación es 1-Wire y existen librerías listas para

utilizarse con un ATmega328p o una versión más pequeña y económica como el ATtiny85.
(DS18B20, n.d)

Características:

Rango de temperatura:-

De -55°C a 125°C.

Voltaje:

- 3 voltios a 5 voltios.



Figura 2 Sensor DS18B20. (DS18B20, n.d)

Am2315: Sensor de Temperatura y Humedad Am2315 I2c Digital Alta Precisión

Sensor de temperatura y humedad encapsulado que se conecta por I2C a cualquier microcontrolador tipo Arduino o Raspberry Pi. (ver Figura 3)

Este sensor contiene un termistor para la temperatura y un sensor capacitivo de humedad, un pequeño microcontrolador interno gestiona las lecturas y saca por I2C los resultados finales calibrados.

Además de su encapsulado, que lo hace apto para su uso en exteriores sin problemas, incluye

soportes de montaje para pared o para cajas.

La conexión es sencilla, el cable rojo a 5V, el negro al GND y el amarillo al pin I2C Clock. Además, son necesarias dos resistencias de 10K Ohms pullup conectadas desde los pines SDA y SCL al hilo de voltaje. (Electronics Co, 2020)

Características:

- Alimentación de 3.5 a 5.5V power and I/O
- Consumo máximo de 10mA durante la conversión (mientras toma datos)
- Rango de 0-100% de humedad con precisión del 2%
- Rango de -20 to 80°C de temperatura con precisión típica de $\pm 0.1^{\circ}\text{C}$
- Tasa de actualización de 0.5 Hz
- Tamaño: 98x16mm diámetro
- Peso: 82.64gr
- La dirección I2C de 7-bit del sensor es 05C.



Figura 3. Sensor Am2315. (Electronics Co, 2020)

ESP8266: Modulo Wifi Nodemcu Amica Esp8266 Esp12e

NodeMCU (ver Figura 4) es una plataforma de IoT de código abierto. Incluye el firmware que se ejecuta en el SoC Wi-Fi ESP8266 de Espressif Systems y el hardware que se basa en el módulo ESP-12. El término "NodeMCU" por defecto se refiere al firmware en lugar del DevKit.

El firmware utiliza el lenguaje de scripts Lua. Se basa en el proyecto eLua y se basa en el SDK no operativo OS de Espressif para ESP8266. Utiliza muchos proyectos de código abierto, como lua-cjson y spiffs. (Arduino, 2021)



Figura 4. Modulo Wifi Nodemcu Amica Esp8266. (Arduino, 2021)

Lector De Memoria Micro Sd Tf Spi Arduino:

El módulo (Adaptador de tarjeta Micras) es un módulo lector de tarjeta Micro SD, y la interfaz es SPI a través del controlador del sistema de archivos, (ver Figura 5) . Los usuarios de Arduino pueden usar directamente el IDE de Arduino para lectura-escritura de tarjetas SD.

Las características del módulo son las siguientes:

1. Soporte Tarjeta Micro SD, tarjeta Micro SDHC (tarjeta de alta velocidad)
2. La placa tiene un regulador de voltaje de interfaz que es de 3.3V -
3. Alimentación de 4.5V ~ 5.5V.
4. La interfaz de comunicación es una interfaz SPI estándar
5. 4 orificios de posicionamiento M2 para una fácil instalación

Circuito regulador de 3.3V: salida del regulador LDO 3.3V como chip convertidor de nivel, suministro de tarjeta Micro SD. (



Figura 5. Lector De Memoria Micro SD. (Arduino, 2021)

Node js:

Node.js es un entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor (pero no limitándose a ello) basado en el lenguaje de programación JavaScript, asíncrono, con E/S de datos en una arquitectura orientada a eventos y basado en el motor V8 de Google. Fue creado con el enfoque de ser útil en la creación de programas de red altamente escalables, como, por ejemplo, servidores web. Fue creado por Ryan Dahl en 2009 y su evolución está apadrinada por la empresa Joyent, que además tiene contratado a Dahl en plantilla. (Node.js, 2021)



Figura 6. Logo de Node js. (Node.js, 2021)

Vue js:

Vue.js (comúnmente conocido como Vue) es un framework de JavaScript de código abierto para la construcción de interfaces de usuario y aplicaciones de una sola página. Fue creado por Evan You, y es mantenido por él y por el resto de los miembros activos del equipo central que provienen de diversas empresas como Netlify y Netguru.

Las características avanzadas necesarias para aplicaciones complejas como el enrutamiento, la gestión de estados y las herramientas de construcción se ofrecen a través de librerías y paquetes de apoyo mantenidos oficialmente, con Nuxt.js como una de las soluciones más populares.

(Vue.Js, n.d.)



Figura 7. Logo de Vue js. (Vue.Js, n.d.)

6.3. Marco legal

Este proyecto está relacionado en un marco de normas, requisitos y decretos que regulan temas de interés que se abordan en su desarrollo.

Según la Resolución número 1441 de 2013 “Por la cual se definen los procedimientos y condiciones que deben cumplir los Prestadores de Servicios de Salud para habilitar los servicios y se dictan otras disposiciones”. Los medicamentos homeopáticos, Fito terapéuticos, productos biológicos, componentes anatómicos, dispositivos médicos (incluidos los sobre medida), reactivos de diagnóstico in vitro, elementos de rayos X y de uso odontológico y demás insumos asistenciales que utilice el prestador para los servicios que ofrece, incluidos los que se encuentran en los depósitos o almacenes de la institución se almacenan bajo condiciones de temperatura, humedad, ventilación, segregación y seguridad apropiadas para cada tipo de insumo de acuerdo con las condiciones definidas por el fabricante o banco de componente anatómico. El prestador debe contar con instrumentos para medir humedad relativa y temperatura, así como evidenciar su registro, control y gestión. Nevera o depósito frío para el almacenamiento de sangre o de sus componentes que cuente con un sistema de registro y control de temperatura entre 1°C y 6°C, una nevera para el almacenamiento de sueros y reactivos con termómetro interno para control de temperatura. (Ministerio de Salud y Protección Social, 2013)

Dada la Resolución número 1403 de 2007 “Por la cual se determina el Modelo de Gestión del Servicio Farmacéutico, se adopta el Manual de Condiciones Esenciales y Procedimientos y se dictan otras disposiciones” estipula lo siguiente:

Las áreas para el almacenamiento de medicamentos y dispositivos médicos deben ser independientes y señalizadas, con condiciones ambientales, temperatura y humedad relativa controladas. Es aquella debidamente adecuada para acondicionar y conservar las mezclas. Debe ser independiente del área de elaboración y tener los medios que garanticen la temperatura de refrigeración (2°C a 8°C), debidamente controlada, cuando la mezcla así lo requiera.

Condiciones de temperatura y humedad

Los sitios donde se almacenen medicamentos deben contar con mecanismos que garanticen las condiciones de temperatura y humedad relativa recomendadas por el fabricante y los registros permanentes de estas variables, utilizando para ello termómetros, higrómetros u otros instrumentos que cumplan con dichas funciones. (MINISTERIO DE LA PROTECCIÓN SOCIAL RESOLUCIÓN NÚMERO 1403 DE 2007, 2007)

7. Diseño metodológico

En la Figura 8, se observa el diagrama ilustrativo del sistema de visualización y monitoreo, donde se observa el uso del protocolo de comunicación MQTT, la tarjeta de desarrollo NodeMCU, base de datos MongoDB y el uso de node js para la interfaz gráfica.

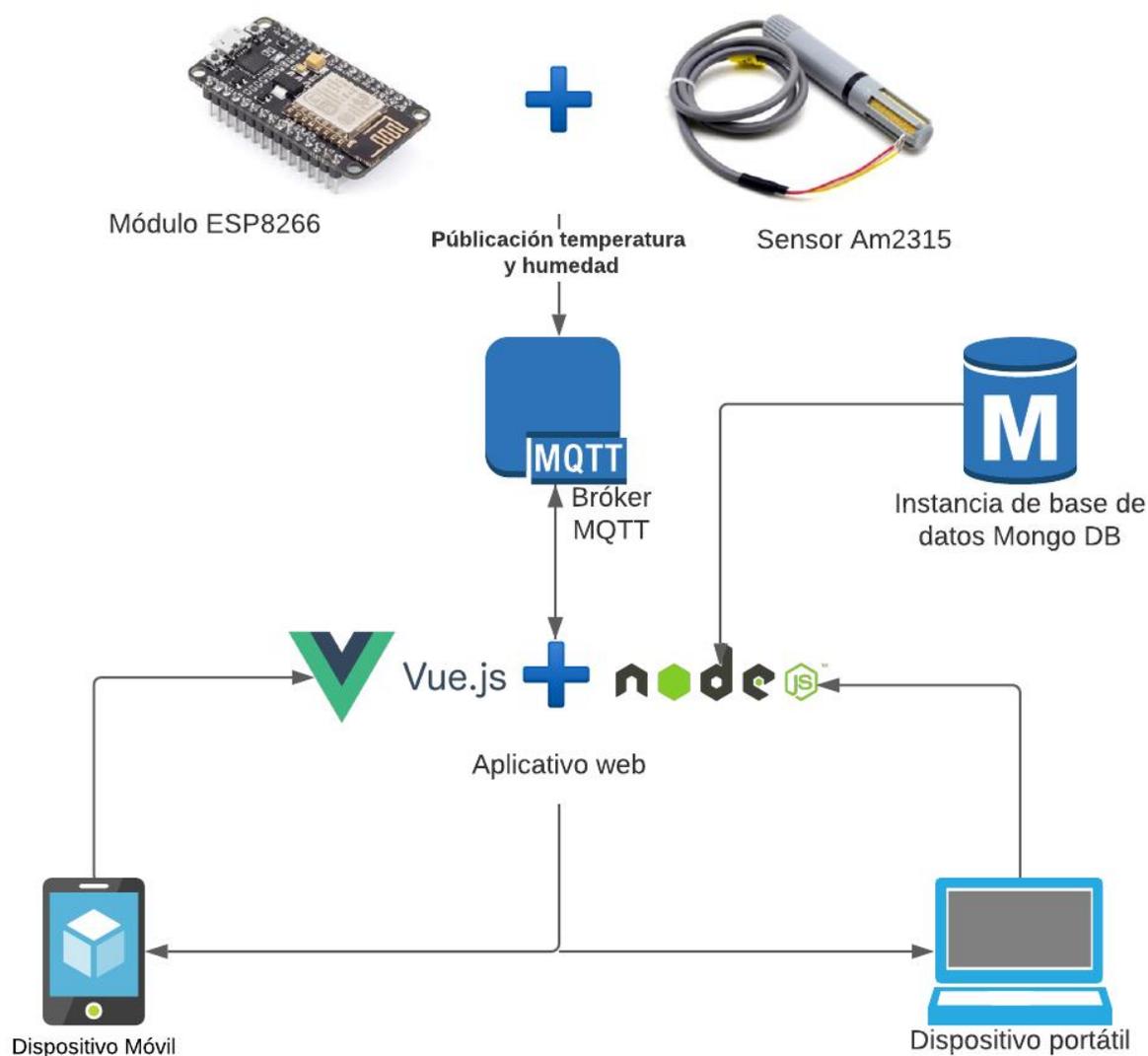


Figura 8. Representación gráfica Topología del sistema IoT.

Para dar cumplimiento a los objetivos planteados en el presente documento, se presenta a continuación las actividades y metodologías desarrolladas. .

Objetivo 1: Analizar el estado del arte de la implementación de nuevas tecnologías IoT, protocolos de comunicación, y software.

- Investigación MQTT

Metodología: Se investigó el protocolo de comunicación MQTT, las ventajas de este protocolo y las diferencias con respecto al protocolo HTTP.

- Investigación General

Metodología: Se investigó las tecnologías IoT para conectar el módulo ESP8266 con el aplicativo web por medio del protocolo MQTT, así mismo las herramientas para desarrollar el aplicativo web Node js y Vue js con los sensores Am2315 y DS18B20, y el módulo memoria micro SD.

Objetivo 2: Diseñar la estructura del sistema de visualización y monitoreo con los sensores.

- Diseñar prototipo

Metodología: Se diseñó un prototipo del sistema con estos dispositivos, módulo ESP8266, sensor DS18B20 y el módulo memoria micro SD donde se visualiza los datos primeramente en el bróker MQTT.

- Módulo de diseño

Metodología: Se diseña la interfaz gráfica del aplicativo web y las conexiones entre los sensores y el módulo ESP8266.

- Pruebas del diseño

Metodología: En esta etapa se verificó el diseño preliminar del sistema de monitoreo con el nuevo protocolo de comunicación MQTT y las conexiones entre los dispositivos.

Objetivo 3: Realizar el prototipo del sistema de monitoreo con los sensores basado en tecnologías IoT y desarrollar el aplicativo web para la visualización de los datos registrados.

- Realizar el prototipo

Metodología: Se construyó el prototipo basado en el diseño, donde se verificó la obtención de datos por parte del de los sensores al módulo ESP8266 y el correcto envío al bróker MQTT.

- Desarrollar aplicativo web

Metodología: El código se ha desarrollado con Node js y NodeRED el cual, se encarga de: Monitorear y visualizar los valores obtenidos de temperatura en una gráfica de tiempo real, así mismo facilita la lectura de datos, donde se cree usuarios y delegue acceso a las diferentes funciones.

Objetivo 4: Implementar el sistema de visualización y monitoreo en un entorno de refrigeración remoto.

- Desarrollo de almacenamiento

Metodología: El sistema siempre debe guardar los datos, por ende, se añadió el módulo memoria micro SD para el almacenamiento de los datos cuando el dispositivo se desconecte de la red WIFI.

- Implementación del sistema

Metodología: Se implementó el sistema completo con el módulo ESP8266, los sensores de temperatura y humedad, el módulo memoria micro SD y el aplicativo web, para poder visualizar y monitorear los datos de manera remota.

- Revisión del sistema

Metodología: Al tomar datos de temperatura y humedad en un entorno de refrigeración se realizó la verificación del envío correcto de datos por parte del sensor y que pudieran ser visualizados de manera correcta en el aplicativo web.

Con estas actividades se obtuvo un sistema de visualización y monitoreo de entornos de refrigeración basado en tecnologías IoT completamente óptimo para la empresa VHM Ingeniería.

8. Resultados

Para el desarrollo de este proyecto fue necesario ejecutar las etapas presentadas en el siguiente diagrama de flujo (ver Figura 9)

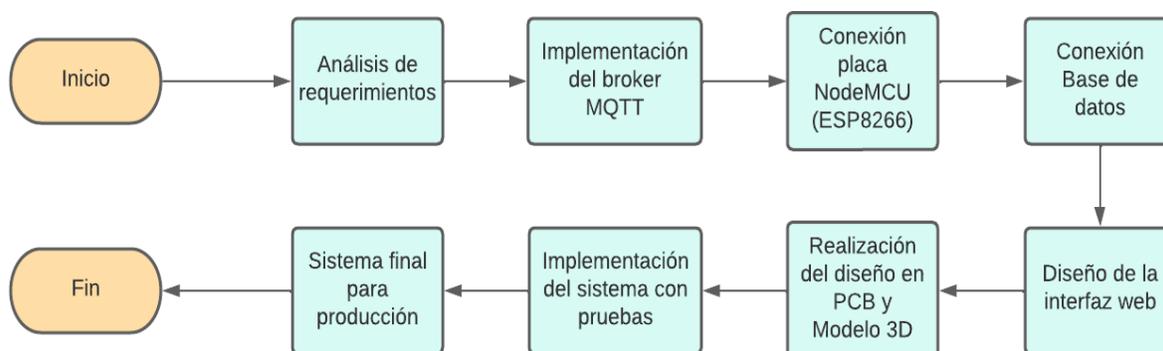


Figura 9. Diagrama de flujo del sistema de monitoreo y visualización.

8.1. Análisis de requerimientos

Se requiere que, desde la interfaz web se visualice y monitoree los datos de temperatura y humedad en tiempo real del entorno de refrigeración, por consiguiente, la tarjeta NodeMCU debe estar siempre conectada a internet por medio de WIFI, si no está conectada a internet o el bróker de mensajería se encuentre desconectado, los datos de temperatura y humedad se almacenarán el módulo lectura escritura micro SD para posteriormente extraerlos.

El sistema debe contar con una interfaz web con seguridad (Login) para poder acceder a los datos con un usuario y contraseña, para eso el cliente debe acceder con los datos correctos, la aplicación después de 20 segundos aproximadamente, se reiniciará el inicio de sesión y así el aplicativo web vuelve al Login.

- Diseño del prototipo.

Diagrama ilustrativo del sistema de visualización y monitoreo (ver Figura 10), se observa los componentes, materiales y tecnologías para la implementación del sistema, así mismo la incorporación del módulo micro SD card y el broker RabbitMQ.

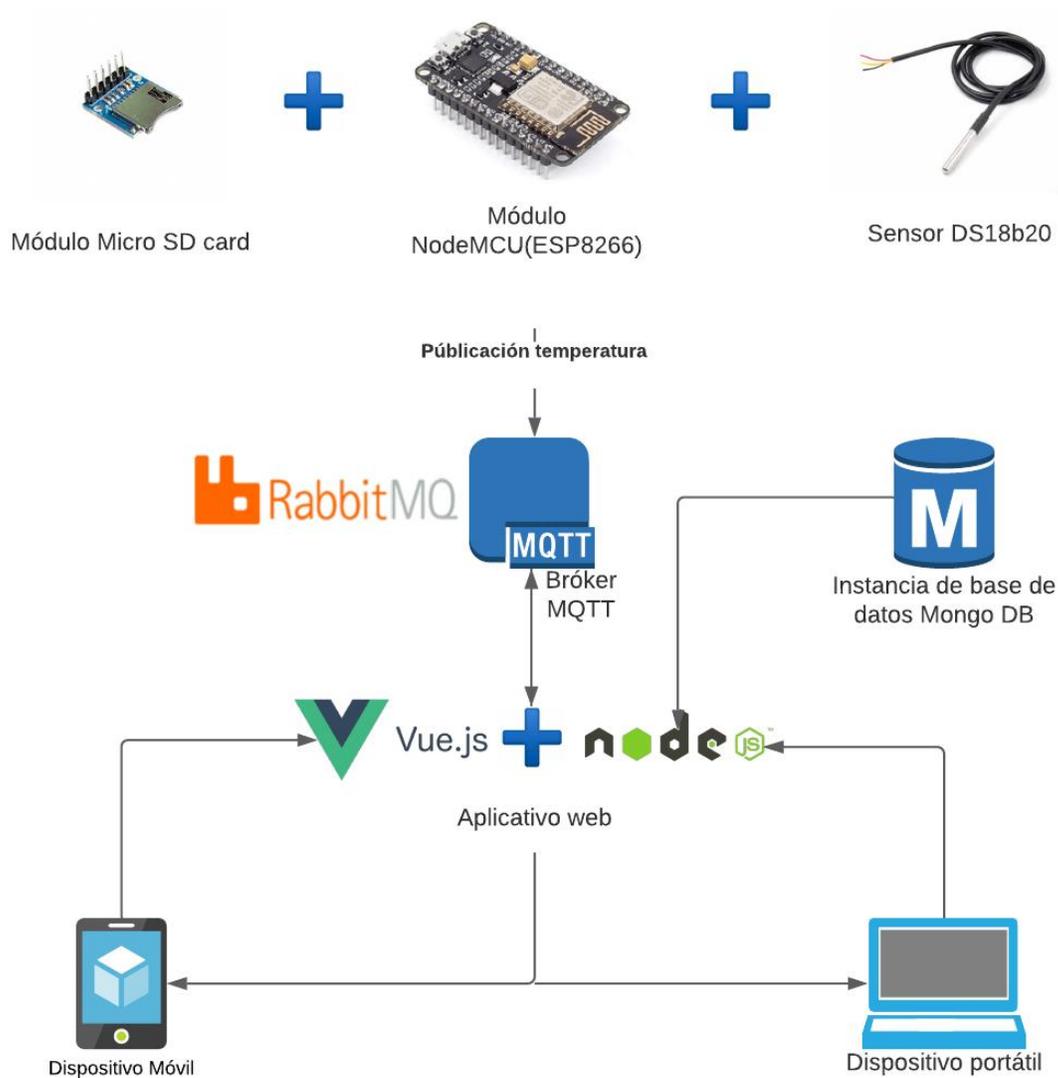


Figura 10. Diagrama del diseño del sistema de visualización y monitoreo.

8.2. Implementación del bróker MQTT

Se investiga el protocolo de comunicación MQTT, las ventajas y diferencias con respecto al protocolo de comunicación HTTP con el fin de entender los beneficios que representa usar este protocolo de comunicación.

Con la investigación a MQTT, se observó que, es un protocolo ligero de publicación y suscripción utilizado sobre el protocolo TCP/IP. MQTT es abierto, sencillo y está diseñado para ser fácilmente implementado. Estas características lo hacen ideal para su uso en muchas situaciones, incluyendo entornos limitados como la comunicación Máquina a Máquina (M2M) y el contexto del Internet de las Cosas (IoT) donde se necesitan códigos de redes limitadas. El patrón de publicación-suscripción de mensajes requiere un broker de mensajes.

Los brokers se encargan de distribuir los mensajes a los clientes interesados en función del tema del mensaje, dado a esto se investigaron los 3 principales brokers MQTT, con la finalidad de encontrar el mejor y más óptimo broker, entre ellos se investigaron:

- Mosquitto MQTT

Mosquitto es un corredor MQTT gratuito de código abierto que se ejecuta en Windows y Linux. Se puede usar un bróker basado en la nube. Eclipse tiene un intermediario MQTT público gratuito y un servidor COAP.

- Apache ActiveMQ

Apache ActiveMQ® es el intermediario de mensajes basado en Java, de código abierto y multiprotocolo más popular.

- RabbitMQ

RabbitMQ es liviano y fácil de implementar en las instalaciones y en la nube. (ver Figura 11) Figura 11. *Broker RabbitMQ. (RabbitMQ, n.d.)* Admite múltiples protocolos de mensajería, entre ellos MQTT. RabbitMQ se puede implementar en configuraciones distribuidas y federadas para cumplir con los requisitos de alta disponibilidad y gran escala.

Según el estudio de los 3 brockes se llegó a la conclusión que, el brocker que mejor se adapta al desarrollo del proyecto es RabbitMQ, dado a que es el intermediario de mensajes de código abierto más implementado, ligero, y se ejecuta en muchos sistemas operativos y entornos de nube, también proporciona una amplia gama de herramientas de desarrollo para los lenguajes más populares.



Figura 11. Broker RabbitMQ. (RabbitMQ, n.d.)

Por lo tanto, se realiza una serie de pruebas y adaptación del entorno de trabajo con RabbitMQ y Arduino IDE, dado que el entorno de desarrollo integrado de Arduino permite trabajar con la placa NodeMCU(ESP8266), sumado a esto se realizó una guía paso a paso, donde se adecua el entorno de trabajo, y así poder llevar a cabo el desarrollo del sistema.

En la Figura 12, se visualiza el total de datos enviados por la tarjeta NodeMCU con un intervalo de 1 segundo. Datos enviados por la tarjeta NodeMUC con el sensor al bróker RabbitMQ, los

cuales se pueden observar en la Figura 13, se observa el ID que identifica al sensor y en la ciudad que se encuentra, Temperatura del entorno antes de introducirse al entorno de refrigeración, y la humedad del ambiente.

A su vez el consumidor, desarrollado con el lenguaje de programación Golang, consume los datos de RabbitMQ y los envía a la base de datos MongoDB.

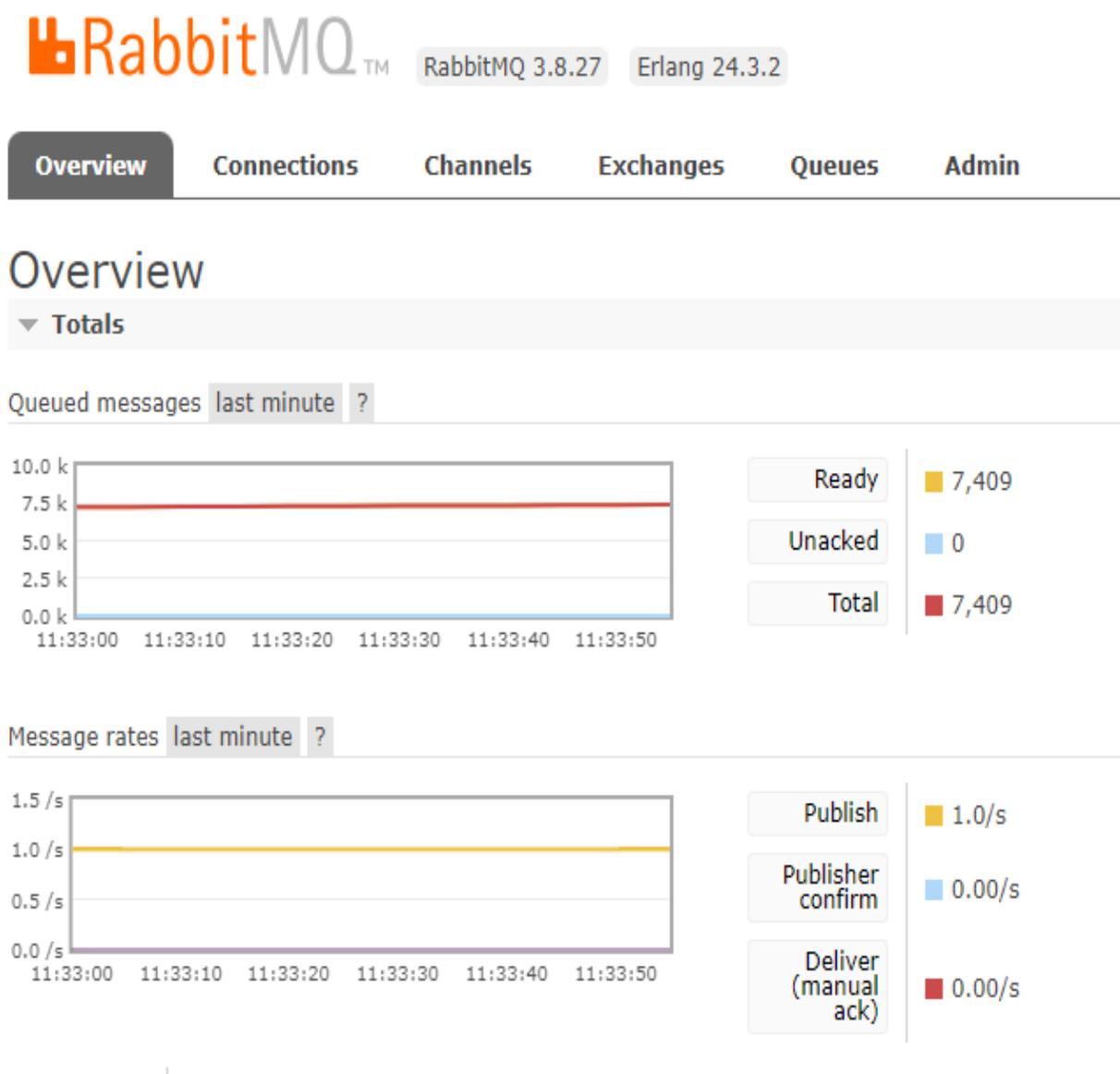


Figura 12. Resumen del total de datos leídos por RabbitMQ.

Overview	Connections	Channels	Exchanges	Queues	Admin
Properties Payload 81 bytes Encoding: string	delivery_mode: 1 headers: x-mqtt-dup: false x-mqtt-publish-qos: 0 <pre>{"id": "123Cucuta", "Temperatura": -2.00, "Humedad": 75, "Fecha": "2022-05-16 16:02:10"}</pre>				
<hr/> Message 24					
The server reported 1 messages remaining.					
Exchange Routing Key Redelivered Properties Payload 81 bytes Encoding: string	amq.topic outTopic 0 delivery_mode: 1 headers: x-mqtt-dup: false x-mqtt-publish-qos: 0 <pre>{"id": "123Cucuta", "Temperatura": -2.13, "Humedad": 75, "Fecha": "2022-05-16 16:02:20"}</pre>				
<hr/> Message 25					
The server reported 0 messages remaining.					
Exchange Routing Key Redelivered Properties Payload 81 bytes Encoding: string	amq.topic outTopic 0 delivery_mode: 1 headers: x-mqtt-dup: false x-mqtt-publish-qos: 0 <pre>{"id": "123Cucuta", "Temperatura": -2.06, "Humedad": 75, "Fecha": "2022-05-16 16:02:31"}</pre>				

Figura 13. RabbitMQ datos enviados por la tarjeta NodeMCU. (RabbitMQ, n.d.)

Adicionalmente, se realizan pruebas de manera local y remota a un objeto del servicio de Amazon (AWS), llamado IoT core donde se puede visualizar los datos enviados desde la tarjeta NodeMCU(ESP8266), en la Figura 14 se visualizan los datos que enviamos a AWS.



Figura 14. AWS servicio IoT core MQTT. (AWS IoT Core, 2022)

8.3. Conexión placa NodeMCU(ESP8266)

Las conexiones se realizaron en el programa Autodesk EAGLE (ver Figura 15) y Fritzing, cuenta con su respectivo diagrama circuito esquemático y PCB (circuito impreso), también llamado Board o footprint.



Figura 15. AUTODESK EAGLE. (Autodesk Eagle, n.d.)

En la Figura 16, se observan los respectivos componentes del sistema los cuales son:

Materiales:

- Placa NodeMCU (ESP8266)
- Módulo micro SD Card
- Puerto Jack 3.5mm para el sensor de temperatura DS18B20
- Puerto Jack Power a 5V para alimentación

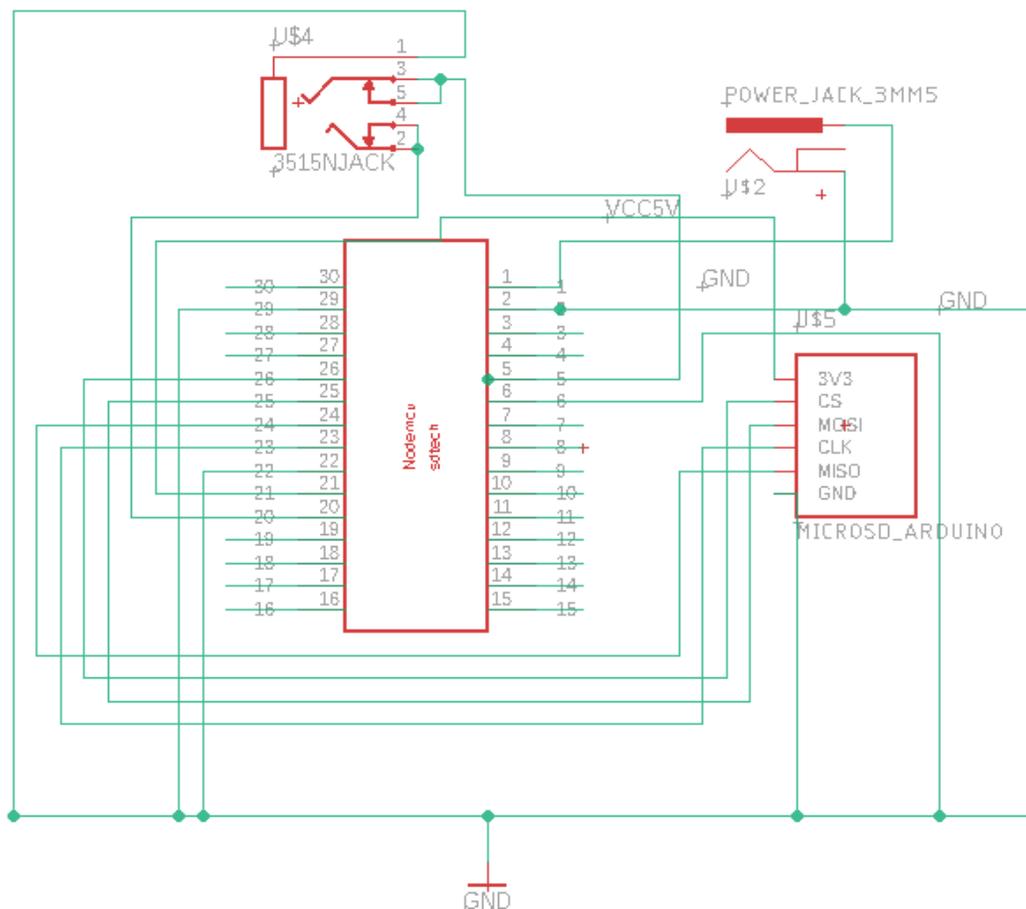


Figura 16. Diagrama circuito esquemático en EAGLE.

En la Figura 17 se observa el PCB (circuito impreso) diseñado en EAGLE, con su respectivos componentes, el router se realizó de forma automática por el programa, para la distribución de pistas, y así poder interconectar cada componente de la mejor forma posible, en este caso se realizó

doble capa, dado que, cuando se realizó la primera versión era muy grande y ocupaba mucho espacio, con la doble capa se puede reducir el tamaño que hay entre componentes.

Para visualizar mejor la conexión entre la tarjeta NodeMCU (ESP8266) y el sensor de temperatura ds18b20, en la Figura 18 se creó un diagrama con el programa Fritzing.

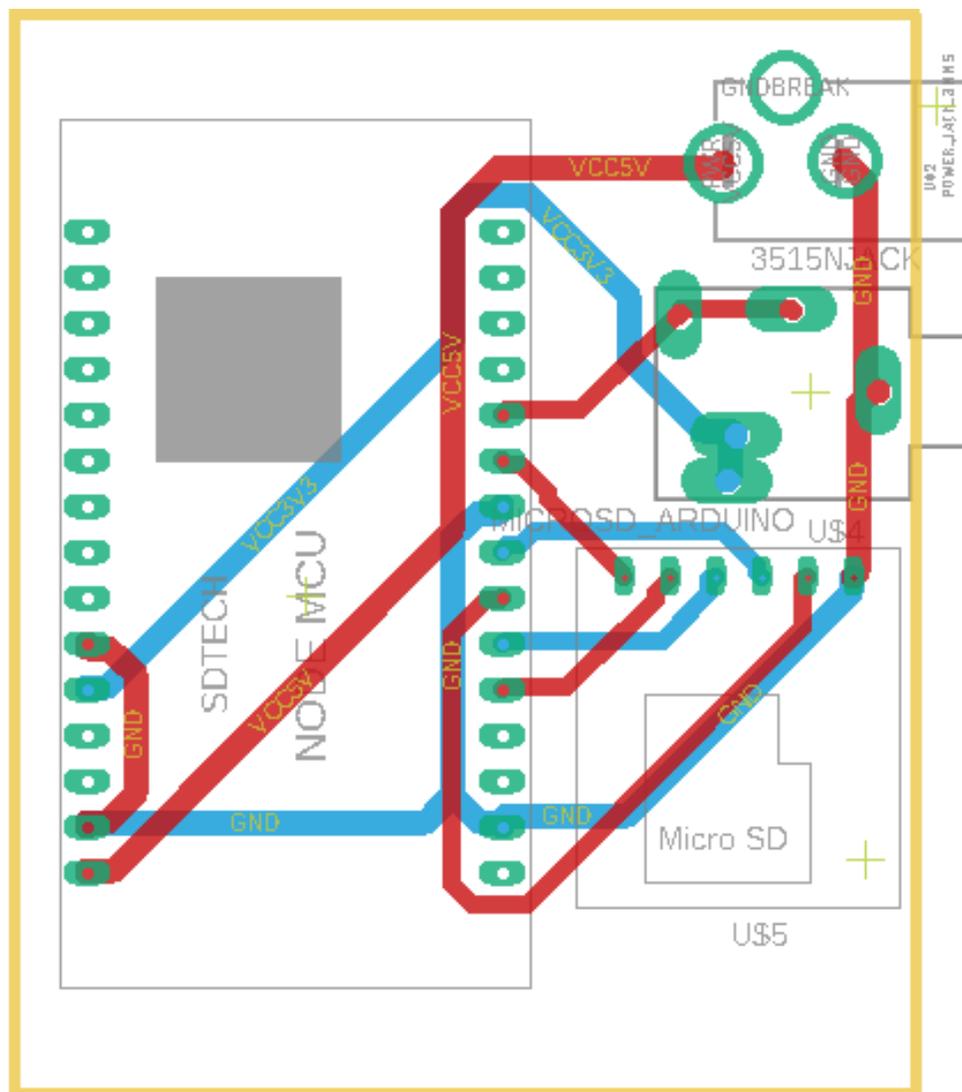


Figura 17. Diagrama circuito PCB en EAGLE.

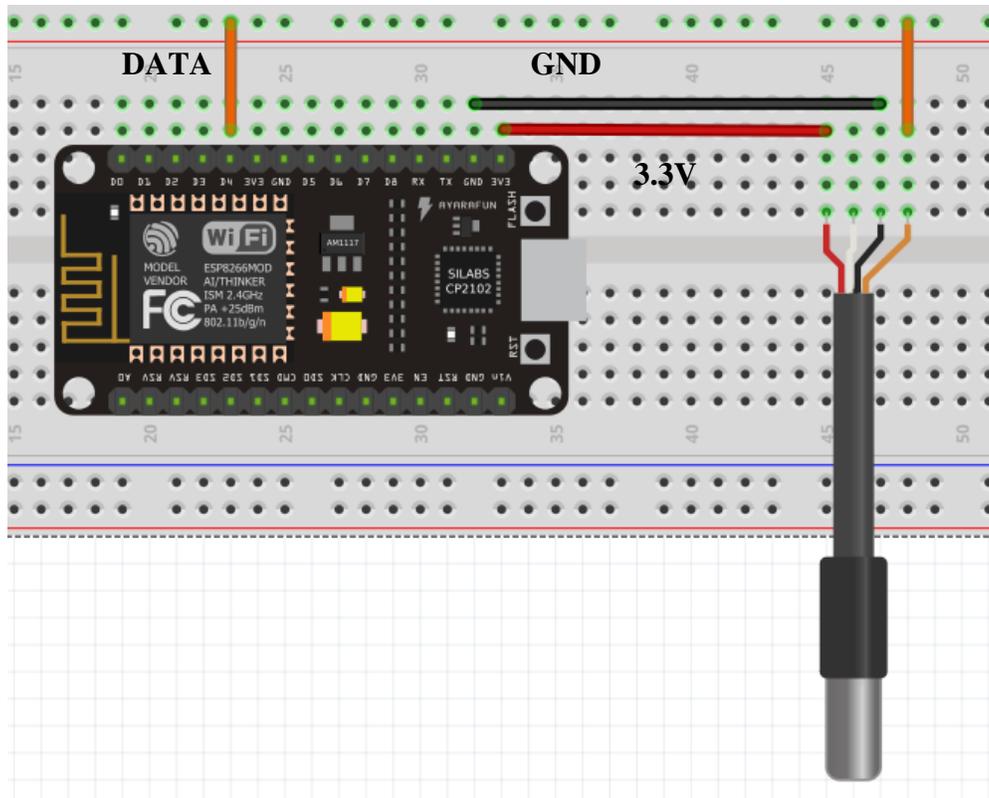


Figura 18. Diagrama circuito PCB en Fritzing.

8.4. Conexión Base de datos

En la Figura 19, se observa la base de datos MongoDB Compass, MongoDB almacena los datos recibidos los cuales son :

- Disp.: El id del dispositivo en este caso 123Cucuta y 123Manizales.
- Temperatura: Almacena los datos de temperatura en un Array
- Humedad: Almacena los datos de humedad en un Array
- Tempactual: Es el último dato adquirido de temperatura enviado desde el dispositivo.
- Humeactual: Es el último dato adquirido de humedad enviado desde el dispositivo.

El consumidor extrae los datos desde el bróker RabbitMQ por medio del cliente desarrollado en

el lenguaje de programación Golang, y los envía a la base de datos MongoDB en formato tipo JSON, MongoDB cuenta con dos sensores, uno en la ciudad de Cúcuta y el otro en la ciudad de Manizales, después de que mongo almacena los datos el aplicativo desarrollado en Node-RED accede a estos datos en formato JSON y los clasifica dependiendo los keys y values para visualizar los datos por separados, así mismo los adecúa para representarlos en el aplicativo web.

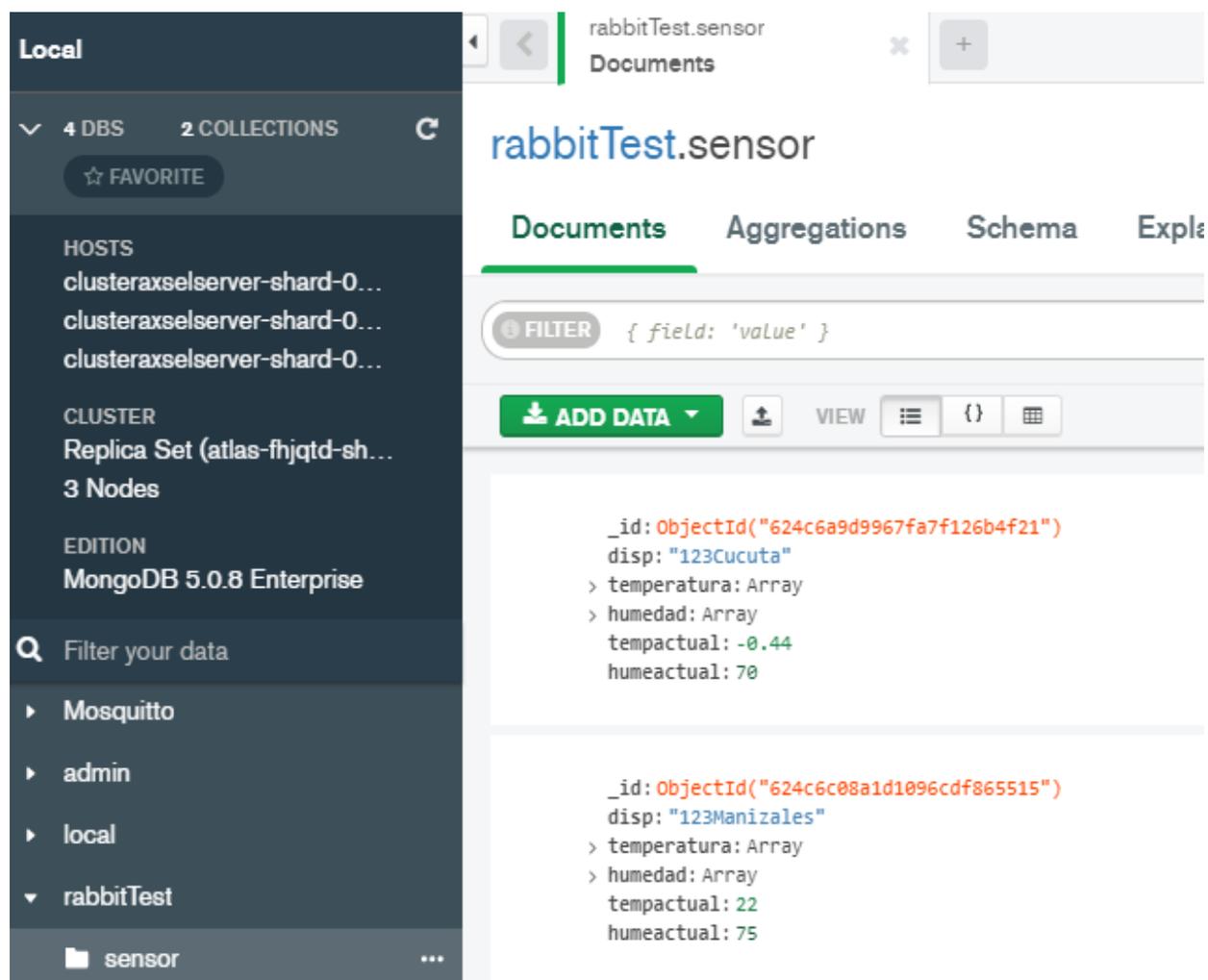


Figura 19. Base de Datos MongoDB compass con dos dispositivos.

En la Figura 20, se observa una vista general de los dispositivos en MongoDB compass.

The screenshot shows the MongoDB Compass interface for a collection named 'rabbitTest.sensor'. At the top, it displays 'Documents' and a search filter '{ field: 'value' }'. Below this, there are statistics: 2 documents, 20.5KB storage size, 2.8KB average size, 1 index, 36.9KB total size, and 36.9KB average size. The interface includes tabs for Documents, Aggregations, Schema, Explain Plan, Indexes, and Validation. A table below shows two documents with columns for _id, ObjectID, disp String, temperatura Array, humedad Array, and tempactu.

	_id	ObjectID	disp String	temperatura Array	humedad Array	tempactu
1	624c6a9d9967fa7f126b4f21		"123Cucuta"	[] 404 elements	[] 6 elements	-0.44
2	624c6c88a1d1096cdf865515		"123Manizales"	[] 7 elements	[] 6 elements	22

Figura 20. Base de Datos MongoDB compass overview.

8.5. Diseño de la interfaz web

El desarrollo del aplicativo web se desarrolló con Node js, por medio de una herramienta de desarrollo basada en flujo para programación visual, Node-RED desarrollada originalmente por IBM para conectar dispositivos de hardware, API y servicios en línea como parte de IoT.

La aplicación cuenta con un Login como se observa en la Figura 21, con este usuario y contraseña anteriormente establecido, podemos acceder a la visualización de los sensores.

The screenshot shows a web login form titled 'VHM Ingeniería Sign-in'. It has a blue header with the word 'Dashboard'. Below the header, there are two input fields: 'username *' and 'password *'. At the bottom of the form, there are two buttons: 'SIGN IN' and 'CANCEL'.

Figura 21. Login VHM Ingeniería.

Se puede acceder a la aplicación desde un celular, Tablet o computador con acceso a internet,

para así poder visualizar y monitorear los datos, como temperatura y humedad del entorno de refrigeración, en la Figura 22, se visualiza el bloque find(verde), este bloque se conecta a la base de datos MongoDB y así adquiere los datos para visualizarlos de forma gráfica, estos nodos se programan con JavaScript para adaptar cada bloque a un flujo adecuado, con sus respectivas especificaciones

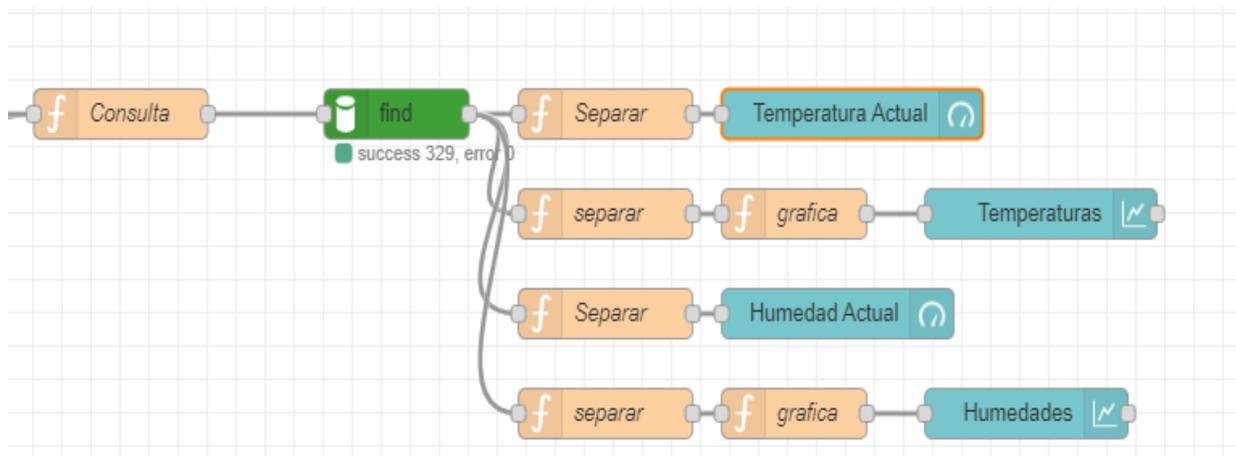


Figura 22. Flujo de la aplicación web con Node-RED.

(ver Figura 23), flujo en Node-RED del inicio de sesión con sus respectivas opciones como: datos correctos, usuario no existe, contraseña incorrecta y sistema utilizado por otro usuario.

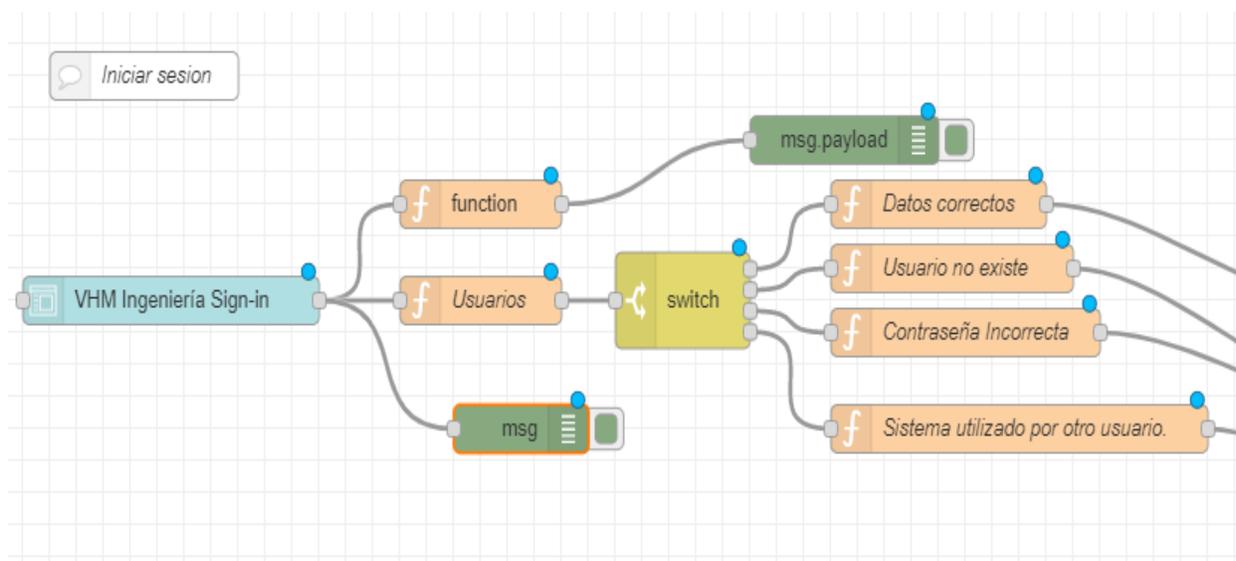


Figura 23. Flujo del Login con Node-RED.

En la Figura 24 y Figura 25, se observa los sensores con sus respectivos datos de temperatura y humedad, en la primera imagen se encuentran los datos del sensor en la ciudad de Cúcuta, el dispositivo se encuentra en una nevera donde se realizan pruebas de datos, y en la segunda imagen en la ciudad de Manizales, este dispositivo se encuentra allá porque mi codirector se encontraba en esa ciudad, es decir que, se puede conectar desde cualquier parte, y acceder a estos sensores de manera remota.

Estos datos de temperatura y humedad se miden en tiempo real, dado el caso que se presente una caída de tensión en la red eléctrica, el sistema almacena los datos en la micro SD, mientras el router se conecta a internet tarda en conectarse alrededor de 1 a 3 minutos, cuando ya se encuentre conectada la tarjeta NodeMCU(ESP8266) a internet, se puede acceder a los datos almacenados en la micro SD por medio de WIFI y con el protocolo de transferencia de archivos FTP.

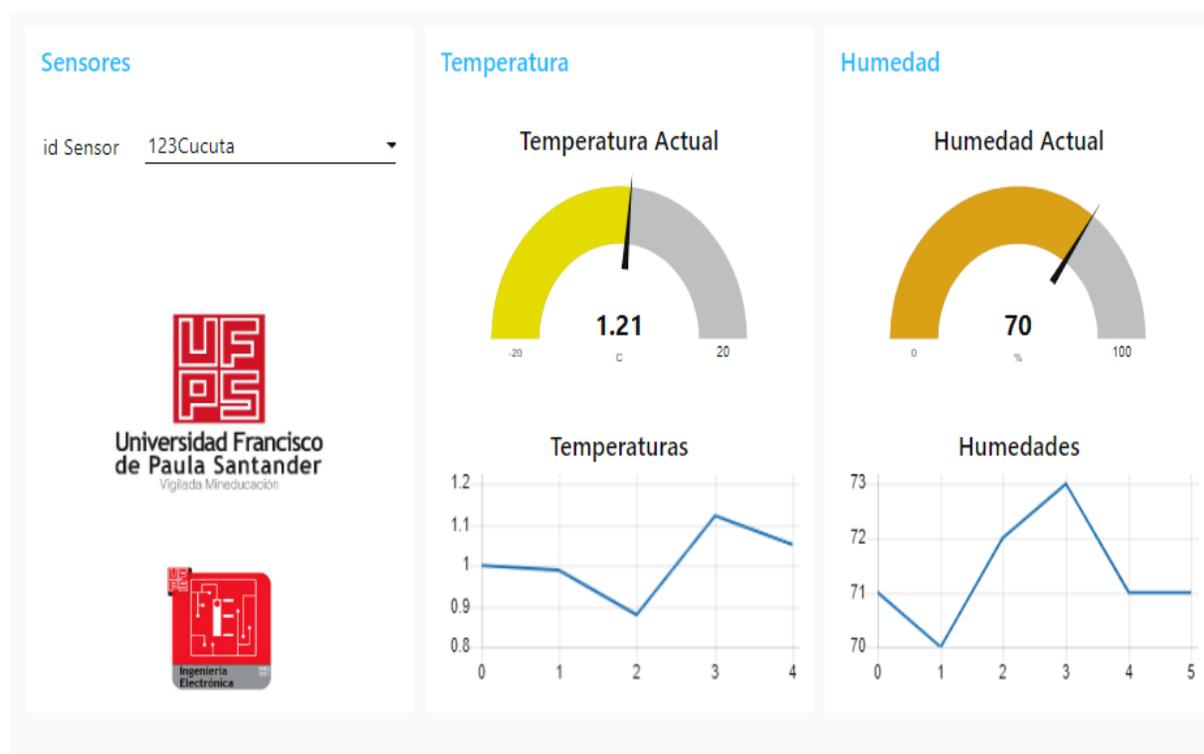


Figura 24 . Visualización y monitoreo del sensor de Cúcuta.

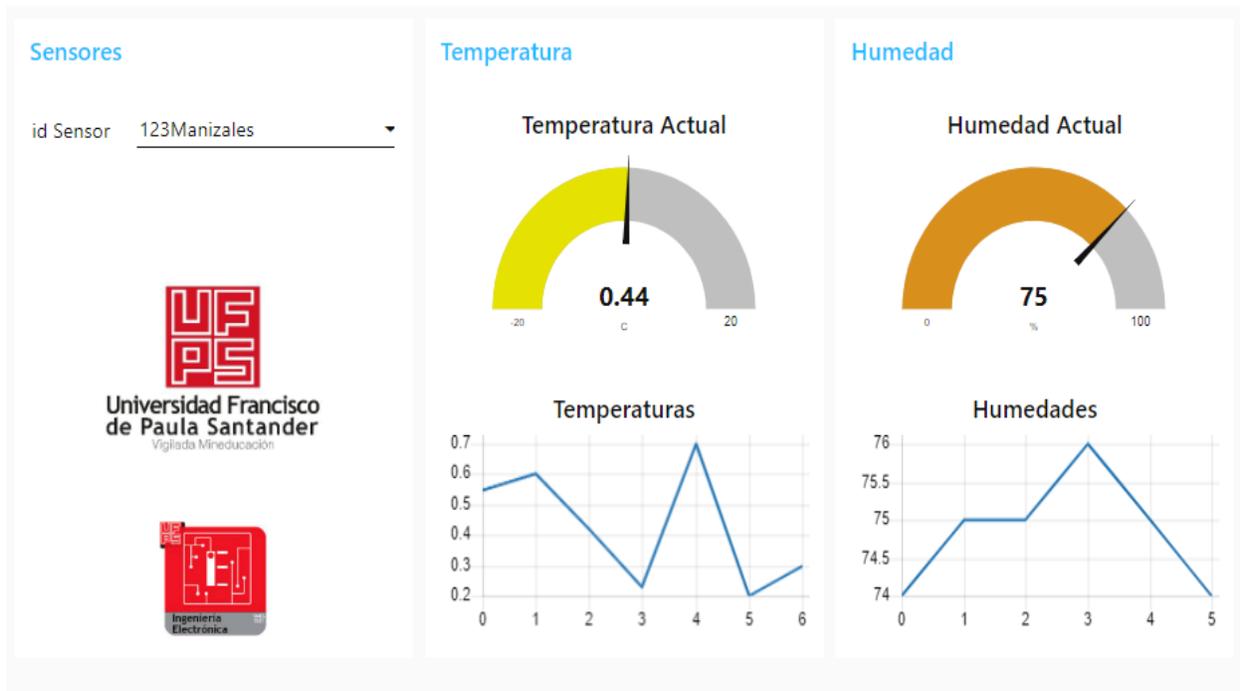


Figura 25. Visualización y monitoreo del sensor de Manizales.

8.6. Realización del diseño en PCB y Modelo 3D

Anteriormente, en el inciso 11.3 se realizó el circuito esquemático y el PCB con el programa EAGLE, dado que este programa no tiene directamente modelo 3D, se opta por implementar nuevamente el diseño del sistema en Fusión 360, ya que, en este programa se pueden realizar las 3 cosas en un mismo lugar como son: Diagrama circuito esquemático, diseño del PCB y modelo 3D, en el modelo 3D también se realizó la caja contenedora de la PCB.

En la Figura 26, se observa el diseño PCB con sus respectivas dimensiones de 5x6cm, en este diseño se omitió el recubrimiento de cobre o polígono de cobre, este distribuye el GND en toda la placa, dado que este recubrimiento afecta a la transmisión de la señal WIFI y disminuye el rango del WIFI, se opta por eliminar el polígono de cobre.

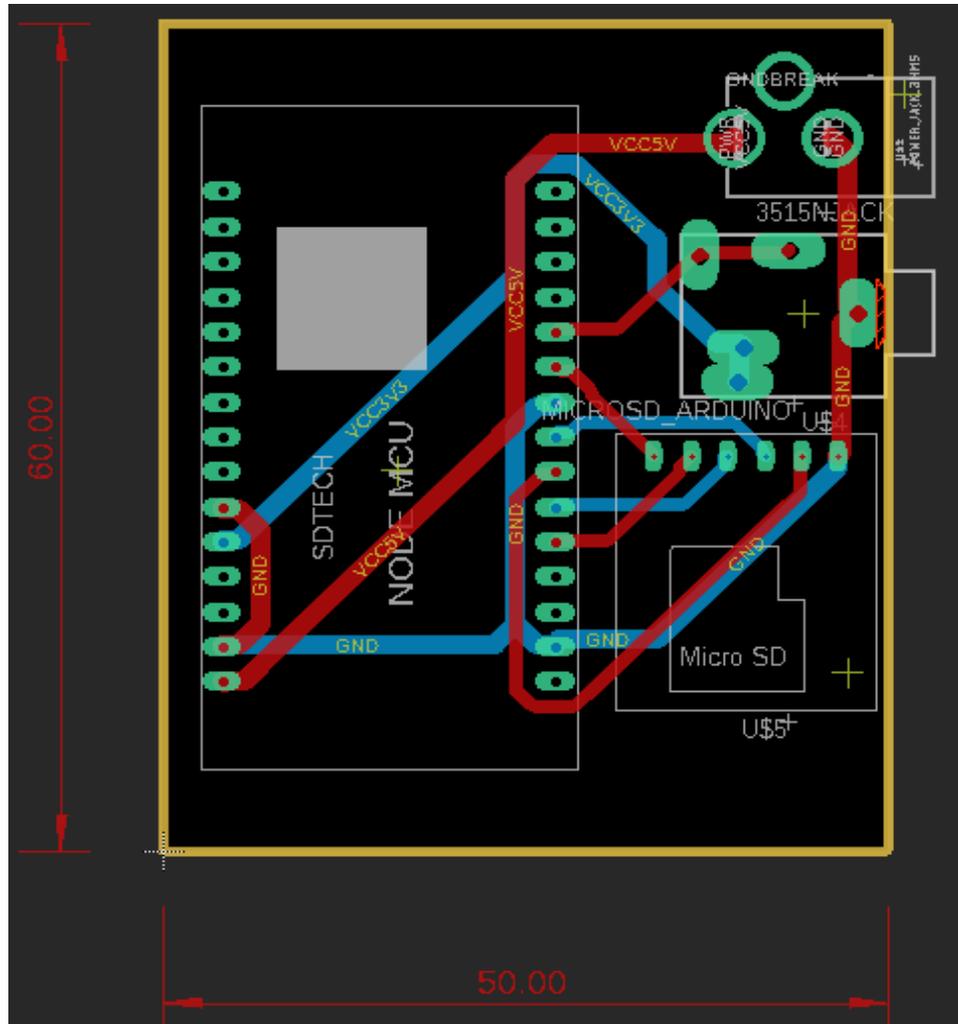


Figura 26. Diseño PCB en Fusión 360.

Los archivos Gerber son documentos vectoriales que identifican la norma para la producción de circuitos impresos. Estos archivos tienen descripciones de las conexiones eléctricas, pistas, vías y pastillas de una PCB junto con todas las instrucciones para su producción. (*Blog Surtel Electrónica*, 2021) En la Figura 27, se observa el Top side, que es el lado superior de la PCB, y en la Figura 28. *PCB en 3D con sus componentes*. se observa la PBC con sus respectivos componentes en 3D, la librería micro SD se creó desde 0, se hizo el diagrama esquemático, el componente con los pines para el PCB y el modelo 3D, los demás componentes se modificaron de su librería original y se implementó el modelo 3D de cada uno.

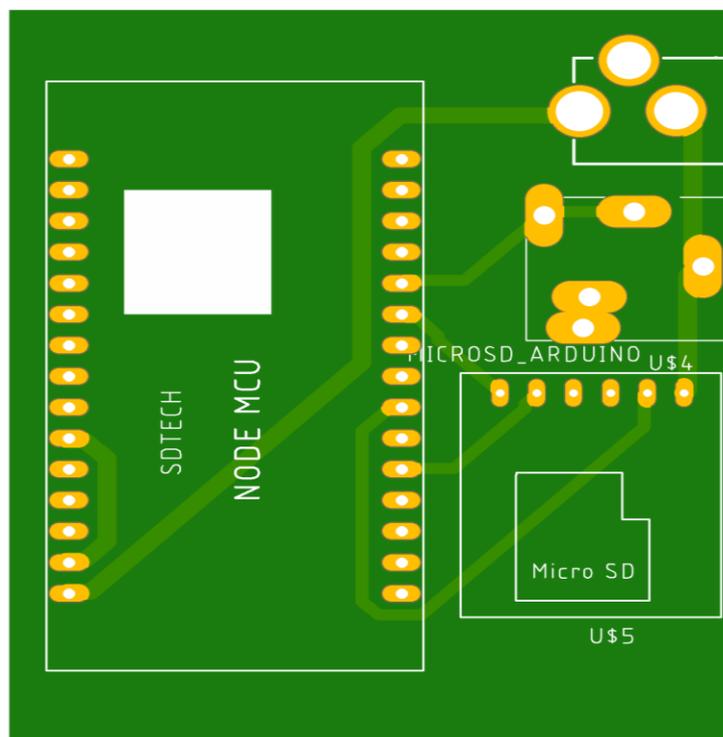


Figura 27. Diagrama de archivos Gerber.

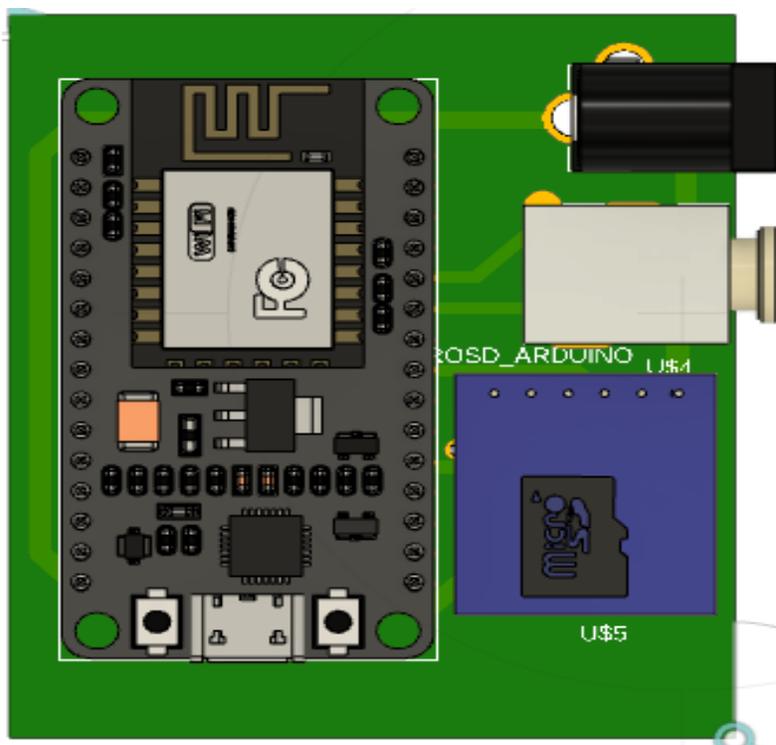


Figura 28. PCB en 3D con sus componentes.

Fusión 360 es una herramienta poderosa para la realización de modelos 3D, sabiendo esto, se diseña la caja contenedora a medida de la PCB como se observa en la Figura 29 y Figura 30, y así ingrese en forma de encaje y ocupe el menor espacio posible.

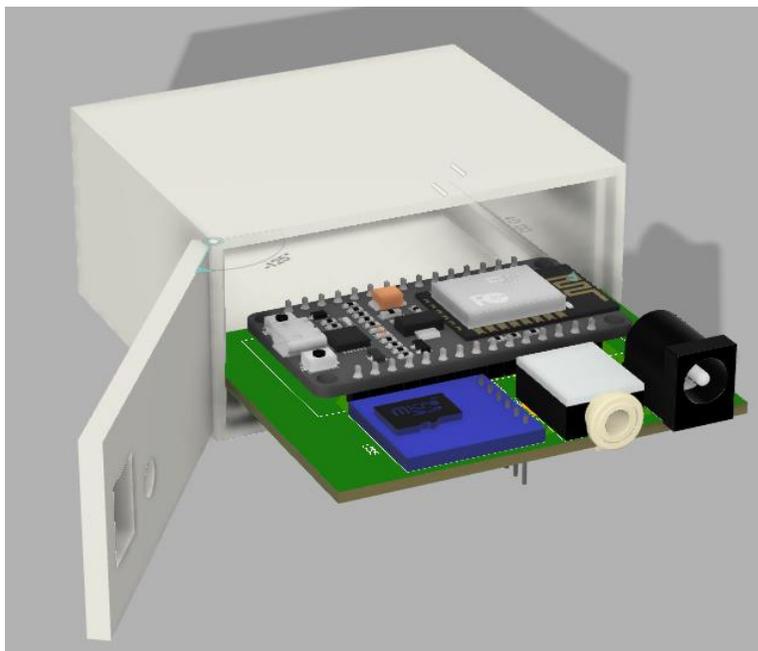


Figura 29. Modelo 3D con la PCB expuesta.



Figura 30. Modelo 3D con tapa cerrada.

8.7. Implementación del sistema con pruebas

La fabricación de la pieza PCB física, se realizó con la maquina EP-2006H, la cual está diseñada para la realización de PCB de alta precisión, en la Figura 32, se suben los archivos Gerber que se generaron en EAGLE Y Fusion 360, que son : copper_bottom.gbr, copper_top.gbr, y para realizar las perforaciones el archivo drill, drill_1_16.xln.



Figura 31. PCB PROTOTYPE MACHINE.

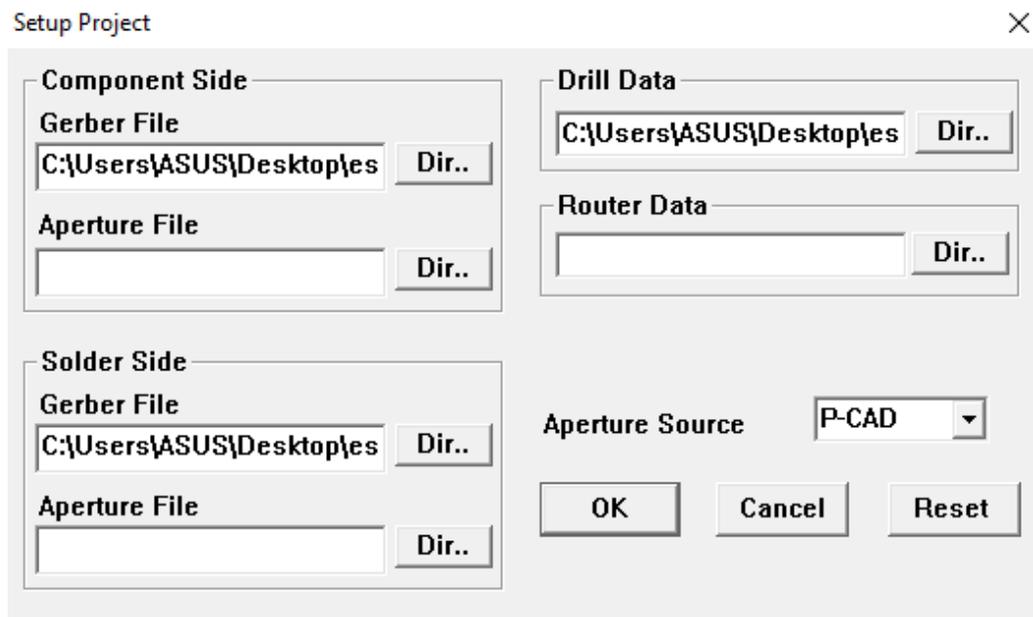


Figura 32. Configuración de los archivos Gerber.

En el programa PCB PROTOTYPE MACHINE calcula el área donde va a trabajar la máquina y realiza pruebas de contacto con la placa, así mismo calcula la distancia entre pistas y los huecos de cada componente, se debe hacer el cambio de broca para cada uno de los procedimientos a realizar como son: apertura de huecos(drill), enrutamiento de pistas, desbaste del cobre restante de la placa, prueba del área de trabajo. En la Figura 33, se observa el diagrama con todos los cálculos que realizó el programa para empezar a grabar en la baquela.

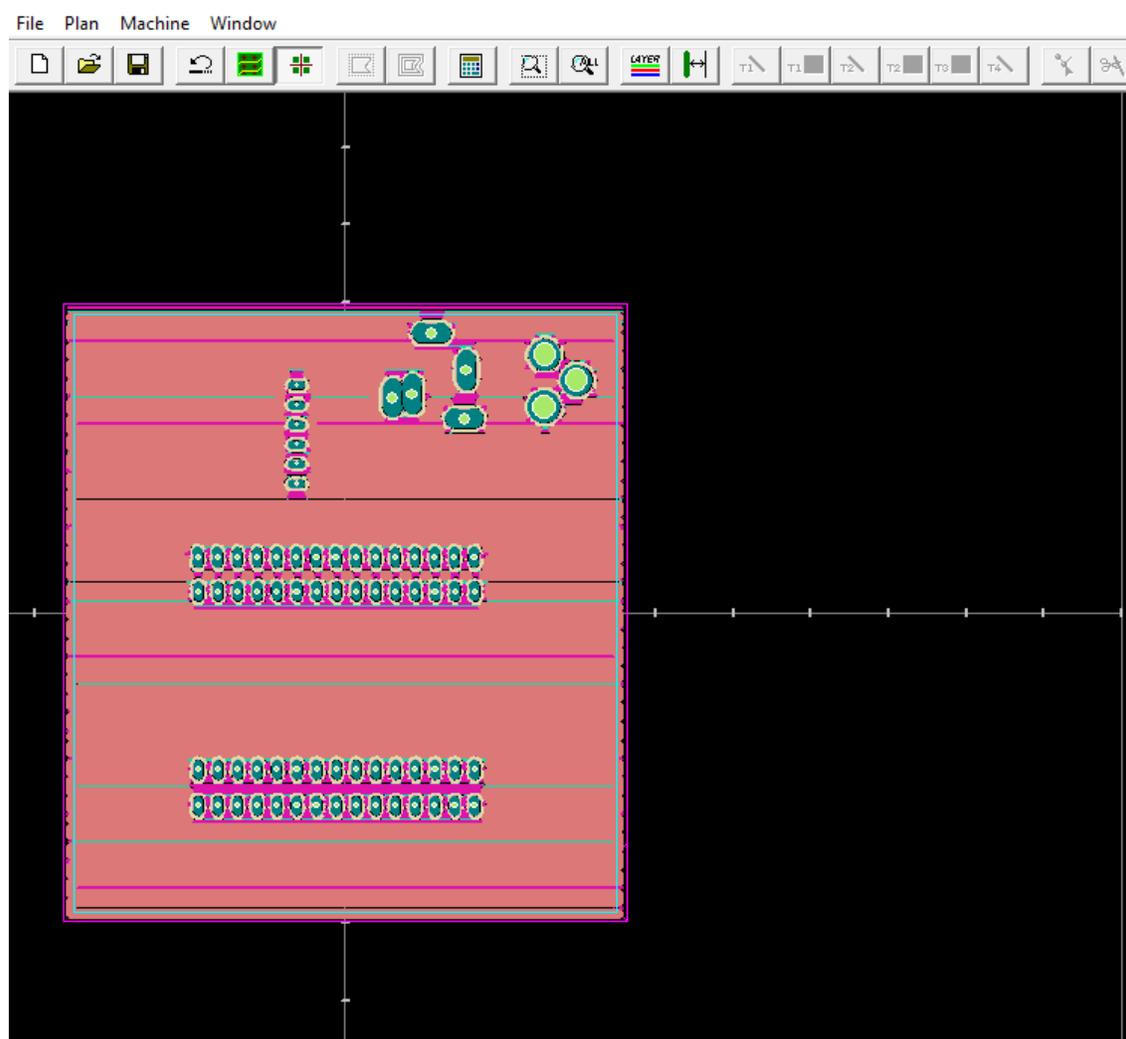


Figura 33. Diagrama con cálculos en PCB PROTOTYPE.



Figura 34. Máquina EP-2006H realizando el PCB.

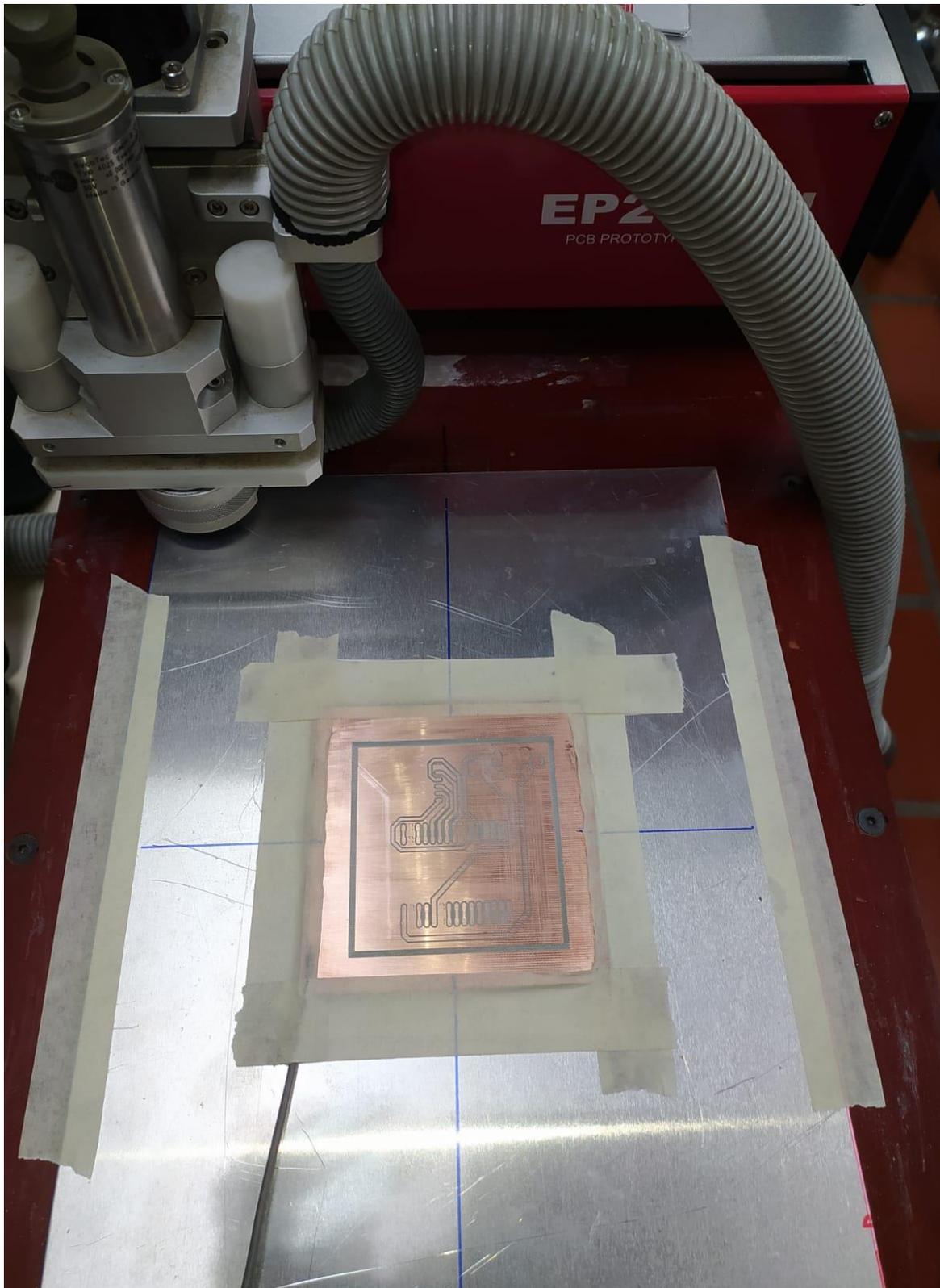


Figura 35. Máquina EP-2006H con el PCB.

Como se puede ver, anteriormente en la Figura 34 y Figura 35, se observa la máquina EP-2006H realizando el PCB que se diseñó en Fusion 360 y se configuró con el programa PCB PROTOTYPE, cabe recalcar que, la PCB de las figuras anteriores son de la primera versión del sistema donde solo es de una capa, en la segunda versión se realizó con dos capas para ocupar menos espacio y hacer el dispositivo lo más pequeño posible.

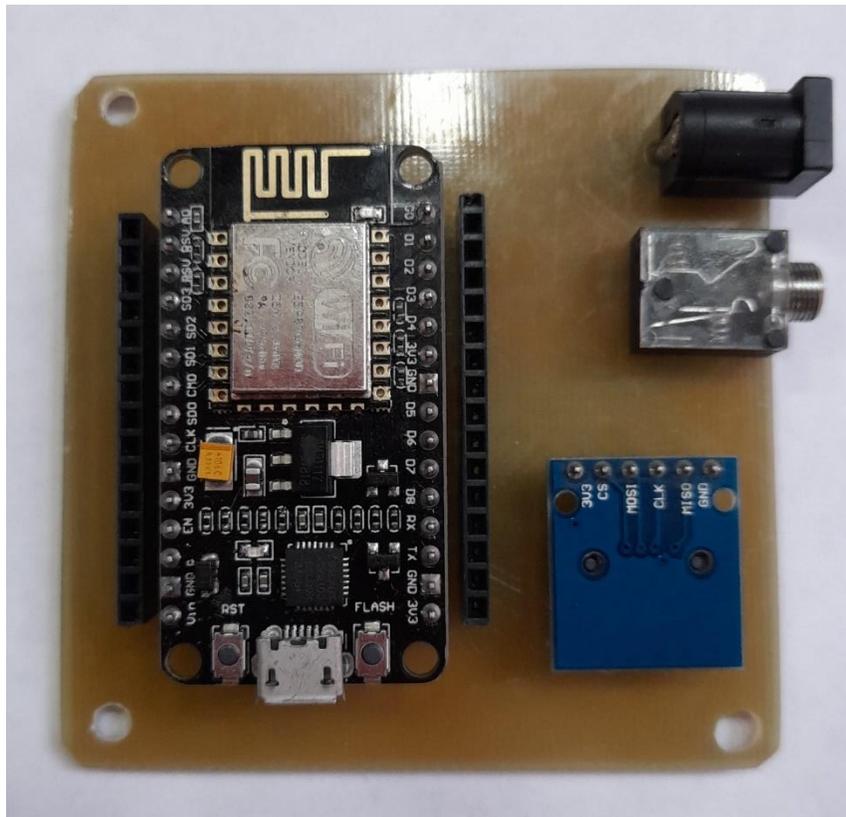


Figura 36. Primera versión del PCB con sus componentes.

La primera versión (ver Figura 36), también cuenta con el diseño en 3D y la caja contenedora fue imprimida en una maquina 3D, en esta versión se realizó perforaciones a las esquinas de la PCB para que no se moviera en la caja como se observa en la Figura 37

En la Figura 38, se observa la segunda versión de la PCB con sus componentes soldados y sin recubrimiento de cobre.

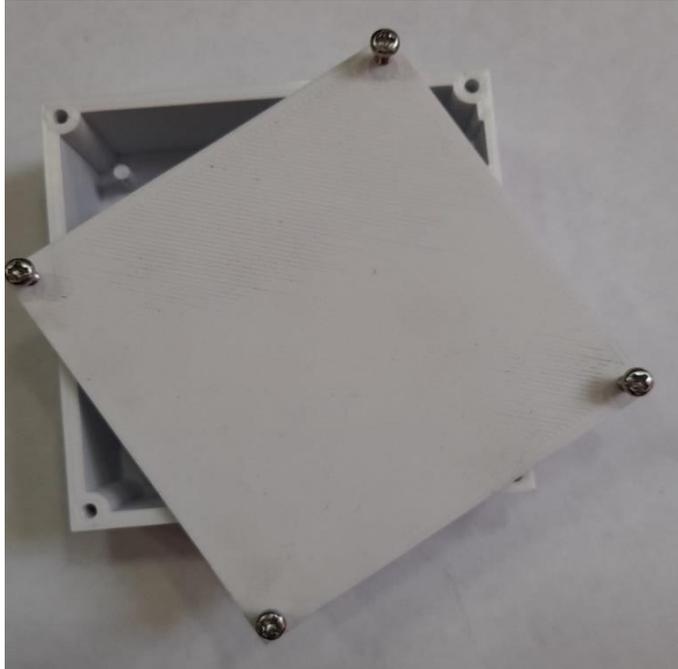


Figura 37. Caja contenedora de la PCB.

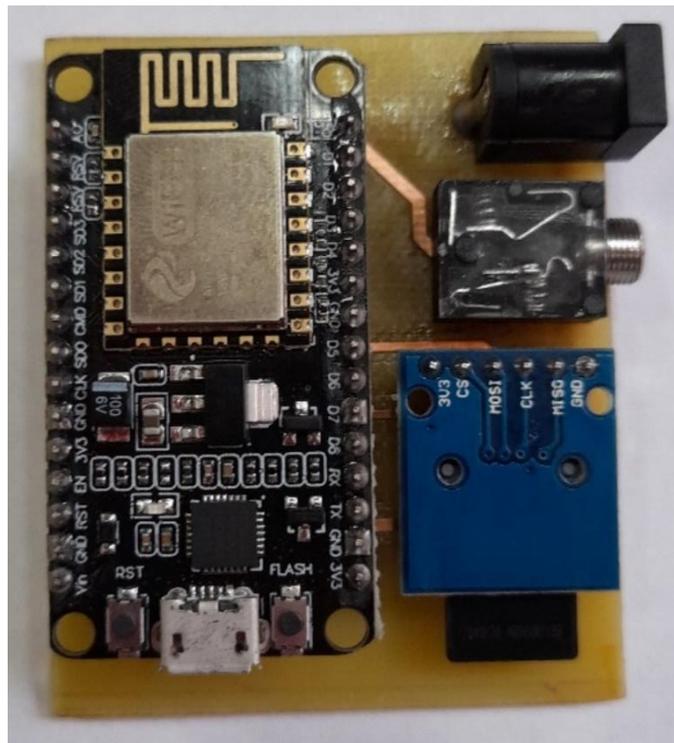


Figura 38. Segunda versión del PCB con sus componentes

8.8. Sistema final para producción

Teniendo en cuenta todo lo realizado anteriormente, se obtiene como resultado un sistema de monitoreo y visualización para entornos de refrigeración para la empresa VHM Ingeniería, el sistema enviará datos de temperatura y humedad en tiempo real al aplicativo web, pasando por el broker RabbitMQ, Consumidor Go, MongoDB, Node JS Y Node-RED, donde se visualizará de forma grafica los datos de temperatura y humedad del entorno de refrigeración.

En las Figura 39, Figura 40 y Figura 41, se observa el dispositivo físico del sistema, el cual es un prototipo viable para llevar a producción, es decir, realizar más sistemas como el presentado, para múltiples entornos de refrigeración.

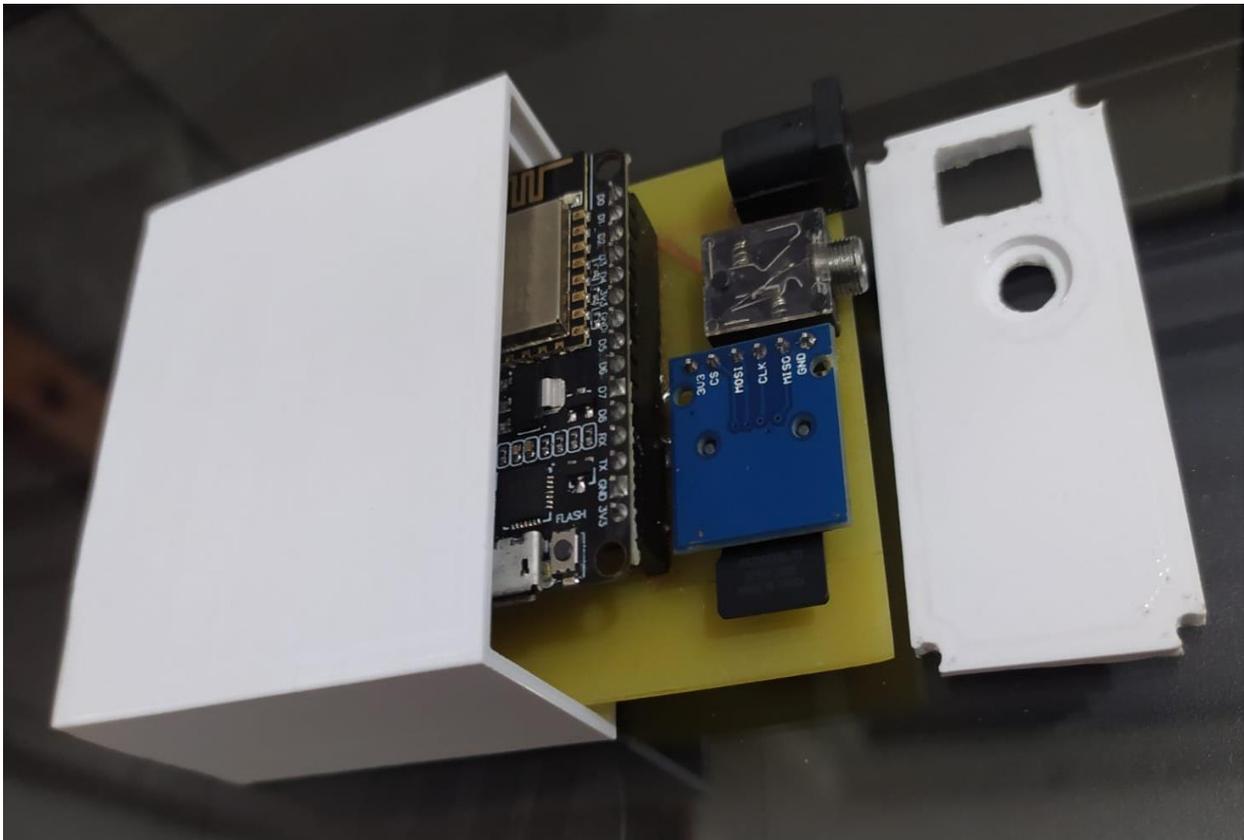


Figura 39. Dispositivo físico del sistema sin armar.



Figura 40. Dispositivo físico del sistema armado.

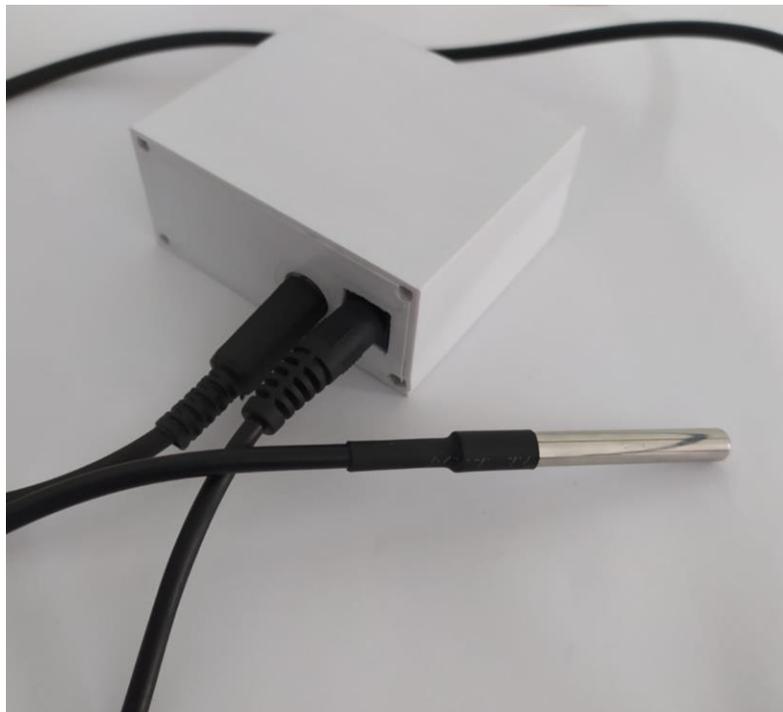


Figura 41. Dispositivo físico con la alimentación y el sensor conectados.

9. Conclusiones

Se visualizó y monitoreó en el aplicativo web los datos de temperatura, humedad, fecha y hora de un entorno de refrigeración usando el sensor AM2301 de forma remota, sumado a eso, los datos obtenidos por la tarjeta NodeMCU ya no se perderán cuando la tarjeta no esté conectada a internet gracias a la micro SD que almacena los datos.

Para poder visualizar desde cualquier parte del mundo los datos de temperatura y humedad, fue necesario crear una máquina virtual, para que el sistema funcione en la nube y no localmente, en este caso, con el servicio que ofrece AWS, Amazon EC2, el cual me proporciona la capacidad de computación escalable en la nube y permite ejecutar aplicaciones en la nube.

Se observó en la primera versión del sistema de visualización y monitoreo en entornos de refrigeración que, el dispositivo físico era muy grande, y que a su vez contenía el polígono de cobre, el cual interfería con la transmisión correcta del WIFI, disminuyendo su alcance máximo. Debido a esto se modifica el diseño de la PCB por uno más compacto que elimina los excesos de cobre, solucionando de esta manera los problemas de la primera versión.

Gracias a las tecnologías de IoT, se puede analizar y tener una visualización local y global de las variables bajo estudio en cualquier ambiente o entorno, de esta manera se puede acceder a un panorama de datos recolectados, para poder vigilar que los datos de temperatura y humedad sean adecuados y cumplan con las normas establecidos por los reguladores, para el entorno de refrigeración específico.

Referencias

- Blog Surtel Electrónica*. (2021.). Recuperado de: <https://www.surtel.es/blog/que-son-los-archivos-gerber/>
- MINISTERIO DE LA PROTECCIÓN SOCIAL RESOLUCIÓN NÚMERO 1403 DE 2007, 0 14 de mayo 245 (2007).
- Node.js*. (2021). Recuperado de: <https://nodejs.org/es/about/>
- Autodesk Eagle*. (2022). Recuperado de: <http://eagle.autodesk.com/>
- AWS IoT Core*. (2022). Recuperado de: <https://aws.amazon.com/es/iot-core/>
- El Mostrador*. (n.d.). Recuperado de: <https://www.elmostrador.cl/agenda-pais/2020/01/13/cuanto-afecta-el-calor-a-los-medicamentos-y-como-conservarlos-para-evitar-alteraciones/>
- DS18B20 sensor de temperatura para líquidos con Arduino*. (n.d.). Recuperado de: <https://programarfacil.com/blog/arduino-blog/ds18b20-sensor-temperatura-arduino/>
- Electronics Co. (2020). *Digital temperature and humidity sensor AM2315*. Recuperado de: <https://cdn-shop.adafruit.com/datasheets/AM2315.pdf>
- Eridani, D. (n.d.). Performance of Sensors Monitoring System using Raspberry Pi through MQTT Protocol. *2018 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, 587–590.
- Hapsari, A. A., Junesco Vresdian, D., Aldiansyah, M., Dionova, B. W., & Windari, A. C. (2020). Indoor Air Quality Monitoring System with Node.js and MQTT Application. *Proceeding - 1st International Conference on Information Technology, Advanced Mechanical and Electrical Engineering, ICITAMEE 2020*, 144–149.

<https://doi.org/10.1109/ICITAMEE50454.2020.9398324>

Rodrigo (2021.). Recupedaro de: <https://hardzone.es/reportajes/que-es/internet-cosas-iot/>

Jiménez Espinosa, E., Domínguez González, M., Pérez Hernández, R., Montero San José, L., &

Cun González, R. (2010). Methodological proposal to improve the process of teaching

learning of the subject Water Quality. *Revista Ciencias Técnicas Agropecuarias*, 19(1), 95–

97.

Macheso, P., Manda, T. D., Chisale, S., Dzipire, N., Mlatho, J., & Mukanyiligira, D. (2021).

Design of ESP8266 Smart Home Using MQTT and. 502–505.

RabbitMQ. (2022.). *Messaging that just works* — Recupedaro de: <https://www.rabbitmq.com/>

Ministerio de Salud y Protección Social. (2013). Resolución 1441. *Biblioteca Digital Minsalud*,

2013, 209. Recupedaro de:

[https://www.minsalud.gov.co/sites/rid/Lists/BibliotecaDigital/RIDE/DE/DIJ/resolucion-](https://www.minsalud.gov.co/sites/rid/Lists/BibliotecaDigital/RIDE/DE/DIJ/resolucion-1441-de-2013.pdf)

1441-de-2013.pdf

Módulo SD para Arduino (2021.). *¿cómo usar tarjetas de memoria en Arduino?* Recupedaro

de: <https://www.tarjetasdememoria.info/micro-sd-para-arduino/>

ESP8266. (2021). *MODULO WIFI NODEMCU AMICA* Recupedaro de:

<https://www.mactronica.com.co/modulo-wifi-nodemcu-amica-esp8266-esp12e>

MongoDB. (2021). Recupedaro de: <https://www.mongodb.com/es/products/compass>

BorrowBits. (2021.). *MQTT vs HTTP: ¿qué protocolo es mejor para IoT* - Recupedaro de:

<https://borrowbits.com/2020/04/mqtt-vs-http-que-protocolo-es-mejor-para-iot/>

Oklilas, F. (2019). *Temperature Monitoring System Based on Protocol Message Queue*

Telemetry Transport (MQTT). 61–66.

Club de La Refrigeracion. (2020.). *Refrigeración es esencial para establecimientos de salud -*

Recupedaro de: <https://refrigerationclub.com/es-es/refrigeracion-es-esencial-para-establecimientos-de-salud/>

Vue.js. (n.d.). Recupedaro de: <https://vuejs.org/>

Anexos

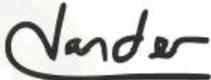
Anexo 1. Certificación Platzi en el curso BÁSICO DE JAVASCRIPT

 **Platzi**

Certifica a
JULIAN DAVID MOLINA VARGAS

Por participar y aprobar el

**CURSO
BÁSICO DE JAVASCRIPT**


Christian Van Der Henst S
COO DE PLATZI




John Freddy Vega
CEO DE PLATZI

Certificación de aprobación online:
Aprobado el 20 de DICIEMBRE de 2021
19 horas de teoría y práctica
https://platzi.com/@Julian_Molina/
Código: 1a6f7649-6263-4975-ba1f-f6d453c51d0

Anexo 2. Certificación Platzi en el curso PRÁCTICO DE JAVASCRIPT



Certifica a
JULIAN DAVID MOLINA VARGAS

Por participar y aprobar el

CURSO
**PRÁCTICO DE
JAVASCRIPT**



Christian Van Der Henst S
COO DE PLATZI



John Freddy Vega
CEO DE PLATZI

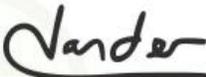
Certificación de aprobación online:
Aprobado el 2 de FEBRERO de 2022
16 horas de teoría y práctica
https://platzi.com/@Julian_Molina/
Código: aad92b7e-d085-4107-bf3d-f8a19ffeed06

Anexo 3. Certificación Platzi en el curso BÁSICO DE PROGRAMACIÓN EN GO



Certifica a
JULIAN DAVID MOLINA VARGAS
Por participar y aprobar el

**CURSO BÁSICO DE
PROGRAMACIÓN EN GO**



Christian Van Der Henst S
COO DE PLATZI



John Freddy Vega
CEO DE PLATZI

Certificación de aprobación online:
Aprobado el 16 de FEBRERO de 2022
22 horas de teoría y práctica
https://platzi.com/@Julian_Molina/
Código: 348ace6b-6cbd-4887-9f82-2d0801ca5d0c

Anexo 4. Certificación Platzi en el curso ASINCRONISMO CON JAVASCRIPT



Certifica a
JULIAN DAVID MOLINA VARGAS
Por participar y aprobar el curso de



**ASINCRONISMO
CON JAVASCRIPT**



Christian Van Der Henst S
COO DE PLATZI

Certificación de aprobación online:
Aprobado el 5 de MARZO de 2022
7 horas de teoría y práctica
https://platzi.com/@Julian_Molina/
Código: 04096035-4e25-4bda-bfcb-208b594060bc



John Freddy Vega
CEO DE PLATZI

Anexo 5. Certificación Platzi en el curso de INTRODUCCIÓN A AWS: FUNDAMENTOS DEL CLOUD COMPUTING



Certifica a

JULIAN DAVID MOLINA VARGAS

Por participar y aprobar el

CURSO DE

**INTRODUCCIÓN A AWS: FUNDAMENTOS
DEL CLOUD COMPUTING**



Aprobado el 29 de MARZO de 2022

Certificación de aprobación online:
14 horas de teoría y práctica

https://platzi.com/@Julian_Molina/

Código: 63080244-c8f8-4b67-b1d1-cd2503f6bc16