

Implementación de un agente basado en sistemas embebidos para monitoreo de variables críticas en un gabinete de servidores

Yan Franco Calderón Félix
Sistemas Embebidos y en Tiempo Real
Maestría en Inteligencia Artificial
Universidad Nacional de Ingeniería
Lima, Perú
ycalderonf@uni.pe

William Hanco
Sistemas Embebidos y en Tiempo Real
Maestría en Inteligencia Artificial
Universidad Nacional de Ingeniería
Lima, Perú
whanco@gmail.com

Omar Parihuana
Sistemas Embebidos y en Tiempo Real
Maestría en Inteligencia Artificial
Universidad Nacional de Ingeniería
Lima, Perú
omar.parihuana@gmail.com

Julio Talaverano
Maestría en Inteligencia Artificial
Universidad Nacional de Ingeniería
Lima, Perú
jtalaverano@uni.edu.pe

Hilber Alexander Cieza Delgado
Maestría en Inteligencia Artificial
Universidad Nacional de Ingeniería
Lima, Perú
alexcidd@hotmail.com

Resumen- En esta investigación se presenta el diseño y desarrollo de un sistema embebido para el monitoreo de variables físicas en un gabinete de comunicaciones/servidores. En este proyecto se mide la temperatura, humedad y un mecanismo de seguridad para control de apertura de puerta. El sistema incluye dispositivos de visualización e interfaz de comunicación para monitorear las lecturas en tiempo real y alertar cuando las medidas superan los umbrales permitidos para un óptimo desempeño de los equipos dentro del gabinete.

Abstract- This research presents the design and development of an embedded system for monitoring physical variables in a communications cabinet/servers. In this project, temperature, humidity and a security mechanism for door opening control are measured. The system includes display devices and communication interface to monitor the readings in real time and alert when the measurements exceed the allowed thresholds for optimal performance of the equipment inside the cabinet

Keywords- *Arduino UNO, Sistema Embebido, Control de Humedad y Temperatura, Seguridad física*

I. SITUACIÓN PROBLEMÁTICA

A. Contexto

Las variaciones de temperatura y humedad en los distintos dispositivos electrónicos tienen efectos adversos para su correcto funcionamiento, el sobrecalentamiento, no solo afecta los umbrales de operación del componente electrónico, sino que puede afectar a todo el dispositivo en general, que deriva en degradaciones de batería, deformación, inclusive explosión de estos, a la vez los circuitos electrónicos pueden provocar errores en su operación. La humedad fuera de los umbrales óptimos puede generar condiciones de condensación y corrosión que no solo degrada al dispositivo y/o acorta su vida útil, sino dañar la lectura de valores de sensores, o inutilizar toda una tarjeta integrada y sus componentes.

Mantener las condiciones adecuadas de temperatura y humedad es esencial para la correcta operación de los equipos de cómputo y comunicaciones. De acuerdo con la ASHRAE la temperatura de los servidores debe de estar en el rango de 18 a 27 °C y la humedad relativa entre 45% y 55%.

Aunque la construcción de los Centros de datos incorpora grandes y costosos elementos de precisión para mantener estas condiciones de humedad y temperatura. Locaciones remotas o de baja escala donde todo el equipamiento de comunicaciones y computo están confinados a un único gabinete, difícilmente incorporan sistemas de climatización, energía y seguridad. Esta realidad se vive en distintas empresas donde el control de las variables esenciales de temperatura y humedad son nulas en gran parte de los casos.

B. Justificación

Técnica: Un sistema de monitoreo y control de bajo costo permitirá gestionar variables sensibles (temperatura y humedad) que puedan afectar la operación de los equipos. El uso de un sistema embebido puede ayudar a tener sistemas que trabajen de manera más óptima y con un mayor tiempo de vida.

Sociales: Diversas medianas y pequeñas empresas, así como instituciones públicas en el Perú consolidan sus plataformas informáticas en un único gabinete y esta usualmente es ubicado, en ambientes no preparados para tal fin. Un dispositivo que ayude a monitorear variables esenciales y a la vez ayudar a la seguridad física del mismo, contribuirá a tomar mejores decisiones sobre las condiciones ambientales donde están expuestos los equipos y optimizar su desempeño.

Económicas: La solución es de bajo costo, que se puede incorporar como un accesorio para los proveedores locales de gabinetes de servidores/comunicaciones. Esto permitirá que se puedan tomar mejores decisiones de las condiciones ambientales del gabinete y contribuir no solo a la estabilidad de los servicios, sino también, a la duración de estos.

Descripción de material	Cant	Costo S/.
Arduino UNO R3	1	45
Sensor de Temperatura y Humedad DHT11	1	11
Sensor Ultrasónico HC-SR04	1	8

Display LCD I2C	1	18
LEDs	3	0.2
Resistencias de 330	4	0.1
Conexiones eléctricas	1	10
Protoboard	1	30
Total		123

Tabla. 1. Costo de materiales para la implementación del sistema embebido (Elaboración propia)

Como se observa en la tabla de costo, el costo total de materiales para la implementación de un sistema embebido para un gabinete es de S/. 123.00 el cual es accesible para una pequeña o mediana empresa.

Ingenieril: La solución es expandible a múltiples sensores que contribuyan a otros aspectos de seguridad física, como incorporar un sistema de control de acceso, o un actuador que permita encender, si está disponible un sistema de ventilación/refrigeración, por ejemplo. Opciones más sofisticadas permite tener una gestión remota del dispositivo, facilitando el monitoreo y control.

C. Viabilidad

El proyecto es viable por la disponibilidad de componentes de bajo costo: sensores, microcontroladores y acceso a herramientas de desarrollo libres como Arduino IDE.

El Sistema embebido al usar componentes electrónicos de bajo costo, de fácil uso y con amplios recursos de software, material de entrenamiento y múltiples casos de uso disponibles en la industria, lo hace viable técnicamente.

Al usar componentes de bajo consumo, y con componentes cada vez más integrados, la construcción se facilita, las dimensiones no alteran la disposición de los demás componentes propios del gabinete (ya que se puede ubicar perfectamente en la parte lateral/frontal), aunque se puede energizar con baterías, el sistema se puede energizar de la red eléctrica que alimenta a los demás equipos.

II. ESTADO DEL ARTE

Con el desarrollo de tecnologías como sensores, comunicación y computación en la nube, en los últimos años han surgido varias aplicaciones inteligentes basadas en sistemas embebidos y de Internet de las cosas (IoT), incluidas las casas inteligentes, la agricultura inteligente y otras [1], [2], [3], [4], [5]. En particular, Los sistemas embebidos con IoT tiene ventajas significativas en el monitoreo y la gestión en tiempo real de diversos entornos [6], [7], [8]. Puede proporcionar interfaces para el acceso remoto, mejorando la capacidad de toma de decisiones proactiva y la gestión ambiental [9]. El diseño de sistemas IoT es crucial para garantizar el rendimiento, la confiabilidad y la escalabilidad de las aplicaciones inteligentes. En [10], se propuso e implementó una arquitectura genérica para aplicaciones IoT basada en protocolos HTTP y Message Queuing Telemetry

Transport (MQTT). Además, se diseñó un modelo de middleware para abordar la heterogeneidad de dispositivos en sistemas IoT [11], [12], que consta de módulos como interoperabilidad de datos, persistencia y análisis. Esta solución puede proporcionar protocolos de interacción estandarizados y reducir la complejidad de la integración de diferentes dispositivos.

Simultáneamente, junto con las aplicaciones de recopilación de datos, las aplicaciones de monitoreo y videovigilancia se están integrando cada vez más en los sistemas embebidos de IoT.

III. APORTES DE INNOVACIÓN

Los aportes por considerar son:

- Diseño modular que permite escalabilidad vertical para poder agregar más sensores a la solución base.
- Sistema embebido replicable de manera sencilla para una escalabilidad horizontal (de acuerdo a la cantidad de gabinetes)
- Facilita la integración con sistemas de monitoreo remoto por el uso de interfaces seriales.
- Alarmas en tiempo real que permiten integrar decisiones automatizadas.
- A través del procesamiento de archivos log se puede realizar analítica descriptiva del comportamiento del sistema sentando las bases para poder implementar algoritmo de machine learning con la data recopilada

IV. OBJETIVOS

Los objetivos que se cumplen con el desarrollo de este prototipo se listan:

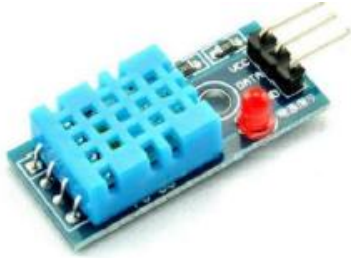
- Objetivos del hardware: construcción de un prototipo que satisfaga las necesidades de medición y monitoreo de las variables críticas.
- Objetivos del Software: Desarrollo de un programa que permite interactuar con los distintos sensores, pantallas y actuadores, con una lógica clara y a la vez sencilla. Que es perfectamente adaptable a cualquier otro entorno.
- Objetivos de la validación: Se puede ver que ante condiciones adversas de temperatura y humedad relativa, el sistema emite señales visuales, tanto en pantalla como en luz, que fácilmente se identifica como un problema.

V. MATERIALES

1. Placa Arduino Uno R3.



2. Sensor de humedad y temperatura DHT11.



3. Sensor de ultrasonido HC-SR04



4. Display LCD/I2C 16x2



5. LEDs



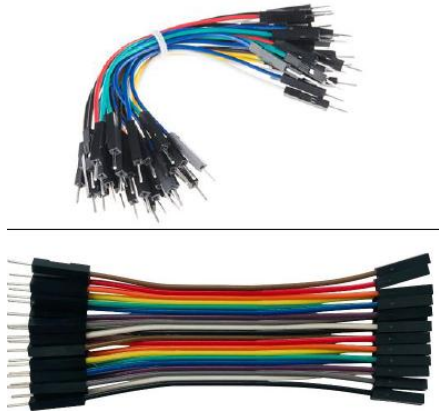
6. Resistencias



7. Protoboard



8. Cable Jumpers.



VI. ALCANCES DEL PROYECTO

El Sistema embebido permite:

- Medir la temperatura y humedad relativa del medio ambiente.
- Mostrar estos valores en una pantalla en tiempo real.
- Alertar visualmente el rebase de algunos de los umbrales de temperatura o humedad a través de activación de luces LED.
- Registrar la apertura de puerta del gabinete.
- Enviar la información de las variables de monitoreo por interfaces serial a través de la captura de log.

VII. LIMITACIONES DEL PROYECTO

El Sistema embebido, aunque preparado para ello, no realiza:

- El sistema no envía los valores de temperatura y humedad en tiempo real a un repositorio centralizado para almacenamiento del historial de registro de las variables. Esto podría ser viable, si se habilita un acceso wifi.

- El sistema no realiza acciones (aunque viable también) de encendido o control de algún sistema de ventilación y/o refrigeración.
- El sistema no realiza ningún efecto mecánico para el cerrado de una puerta del gabinete.

VIII. DESARROLLO

La implementación del sistema embebido siguió las siguientes etapas:

A. Análisis del problema y requerimientos de dispositivos de entrada, salida y señales.

Considerando el alcance del sistema la figura fig. 1. se muestra el estado seguro del sistema la cual se da cuando los sensores del agente leen lo siguiente, la puerta del gabinete está cerrada y los valores de temperatura y el porcentaje de humedad relativa se encuentran dentro de los umbrales definidos para un funcionamiento óptimos de los equipos, en este caso el efector del agente solo emite un mensaje mostrando la lectura de los sensores en una pantalla display.



Fig. 1. Estado seguro del sistema (Elaboración propia)

El sistema pasa a un estado de riesgo, como se muestra en la Fig. 2. cuando los sensores del agente leen al menos una de las 3 variables fuera de rango para un funcionamiento óptimo de los equipos dentro del gabinete, en este caso los efectores de nuestro agente encienden un LED por cada variable fuera de rango y muestran un mensaje en la

pantalla display con los valores leídos por los sensores.



Fig. 2. Estado de riesgo del sistema (Elaboración propia)

De acuerdo con el análisis realizado se determinan los siguientes elementos para el desarrollo de hardware:

- Sensores que permitan medir las variables críticas, temperatura y humedad.
- Sensor que permita medir la proximidad de un objeto para la detección de apertura de puerta.
- Actuadores visuales (3 luces LED) que indiquen visualmente una condición de alerta.

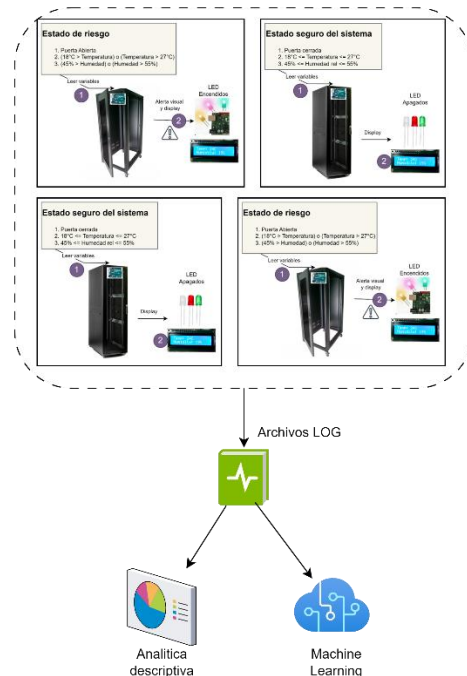


Fig. 3. Procesamiento en lote de datos para analítica descriptiva, diagnóstica y predictiva (Elaboración propia)

Adicionalmente en el trabajo presente se realizará una prueba de concepto para aprovechar e integrar la información obtenida por los sensores en el flujo de procesamiento de datos para realizar analítica descriptiva el gráfico obtenido se

presentará en la sección IX Pruebas y validación. La Fig. 3. Muestra el flujo de trabajo que consiste en la recolección de los datos (seriales) leídos por múltiples sensores de múltiples gabinetes (implementación a escala) para recopilarlos a través de archivos log, y centralizarlos en un repositorio común de datos para aprovechar esa información y poder realizar procesamiento analítico que permita una mejor comprensión de los sistemas para una toma de decisiones informadas.

Según el método de trabajo, en la sección de desarrollo de hardware se procede a la revisión de hojas técnicas de componentes para identificar sus condiciones eléctricas de operación (voltajes, pines y otros).

B. Desarrollo del hardware

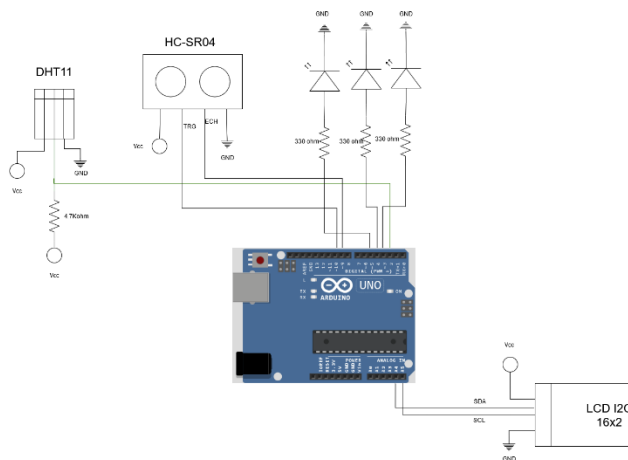


Fig. 4. Diagrama esquemático del sistema (Elaboración propia)

La Fig. 4. Muestra el diagrama esquemático de la conexión de los componentes electrónicos que serán implementados primero en simulación y luego físicamente.

Los principales componentes y conexiones son:

- El microcontrolador (Arduino UNO R3) que actúa como el cerebro del sistema, encargado de recibir y procesar las señales de los sensores, que trabaja con un voltaje de operación de 5V
- Sensor de Temperatura y Humedad DHT11, este sensor mide la temperatura en un rango de 0°C a 50°C con una precisión de $\pm 2^\circ\text{C}$ para temperatura y humedad del ambiente con una precisión de $\pm 5\%$ para humedad. Trabaja con un voltaje de 3.3.V - 5V
- Sensor Ultrasónico HC-SR04, este sensor mide la distancia y se utiliza para detectar si la puerta del gabinete está abierta o cerrada. Este sensor tiene un rango de detección de 2 a 400cm y trabaja con señales TRIG y ECHO
- Display LCD I2C 16x2, este display muestra la información de temperatura, humedad y estado de la puerta.

- 3 LEDs, Se utilizan para indicar visualmente si la temperatura, humedad o estado de la puerta están fuera de los rangos aceptables.
- Resistencias de 330 ohm, se utilizan para limitar la corriente que circula a través de los LEDs.
- Conexiones eléctricas, todas las señales y alimentación se distribuyen a través de estas conexiones entre los diferentes componentes.

Como parte de la metodología y las buenas prácticas para la implementación de sistemas embebidos se realizó la implementación y simulación del sistema en la herramienta de simulación Wokwi [8].

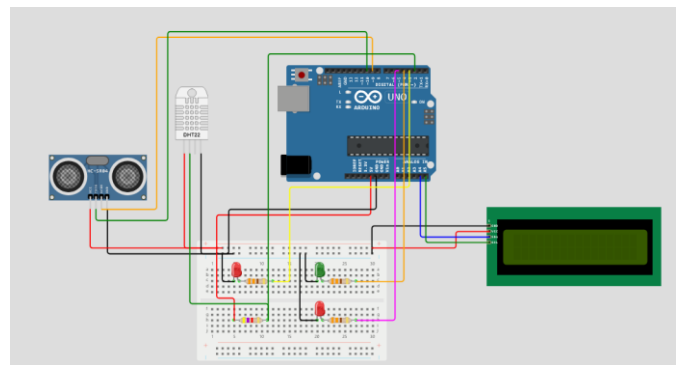


Fig. 5. esquema del sistema en la plataforma Wokwi (Elaboración propia)

Por fines de simulación en Wokwi se usó el sensor DHT22.

Luego de haber realizado el esquema y la prueba en una herramienta de simulación se procede a la implementación física del sistema embebido el cual se muestra en la fig. 6.

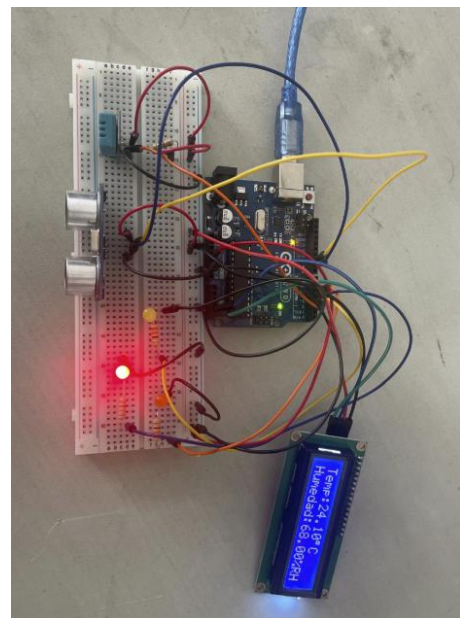


Fig. 6. Implementación física del sistema (Elaboración propia)

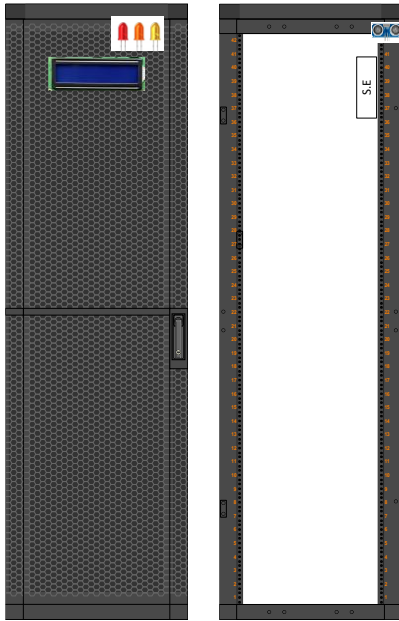


Fig. 7. Instalación del sistema embebido en el gabinete
(Elaboración propia)

En un entorno de producción se propone la ubicación física del sistema embebido en las partes laterales del gabinete para no obstaculizar algún equipo al interior ni el cableado.

Se energizará por medio de una fuente AC/DC de 9V alimentado desde una de las tomas del gabinete.

Los LEDs y el display LCD se podrían adaptar en la parte frontal del gabinete, tal como se muestra en la Fig. 7.

C. Elaboración del algoritmo (DFD)

La figura Fig. 8. Muestra el diagrama de flujo de datos elaborado como primer esbozo antes de la codificación del software, en este diagrama se muestra el inicio, la importación de librería, declaración y configuración de variables, captura de los valores leído por los sensores, la lógica de decisión del agente para accionar los efectores.

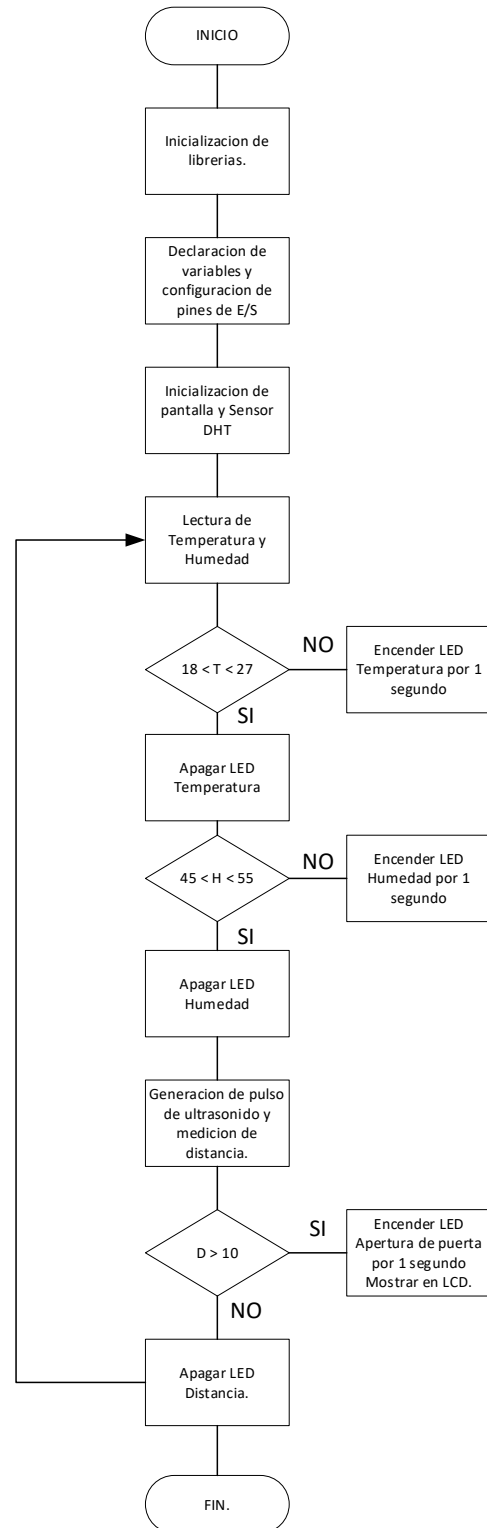


Fig. 8. Diagrama de flujo de datos de la solución (Elaboración propia)

D. Desarrollo del software

Se utiliza el entorno Arduino IDE para programar y cargar la lógica del sistema embebido en el hardware (Arduino UNO R3), a continuación, se presenta las configuraciones realizadas.

Librerías utilizadas:

- Wire.h: Para la comunicación I2C con dispositivos, como el módulo LCD.
- LiquidCrystal_I2C.h: Para controlar la pantalla LCD mediante la interfaz I2C.
- DHT.h y DHT_U.h: Para trabajar con el sensor de temperatura y humedad DHT11.

Protocolos:

- I2C: Protocolo de comunicación para conectar y controlar el módulo LCD desde el Arduino.
- Los pines digitales y analógicos del Arduino están configurados para leer datos (del sensor DHT11 y del ultrasonido) y activar dispositivos (los LEDs).

Definición de pines:

- SENSOR_TEMP (2): Pin del sensor de temperatura DHT11
- LED_TEMP (3), LED_HUM (4), LED_OPEN (5): Pines para los LEDs indicadores
- TRIG (10), ECO (9): Pines del sensor ultrasónico HC-SR04

Inicialización de variables:

- HUMEDAD y TEMPERATURA: Variables para almacenar los valores leídos de los sensores
- DURACION y DISTANCIA: Variables para procesar la señal del sensor ultrasónico

Configuración de dispositivos:

- Inicialización del display LCD
- Inicio del sensor DHT11
- Establecimiento de modos de operación de los pines (entrada/salida)

Una vez definida la configuración de variables, se procede a la codificación, el código fuente se presenta a continuación:

```
//----- MIA-202 Seccion A. Grupo 3.
// Integrantes:
// - Yan Franco Calderón Félix
// - William Hanco
// - Omar Parihuana
// - Julio Talaverano
// - Hilber Alexander Cieza Delgado
//
// ---- Implementación de un agente basado en sistemas embebidos
// ---- para monitoreo de variables críticas en un gabinete de
// servidores
// ---- Diciembre 2024
//

// Inicializacion de librerias necesarias para el uso de los
// sensores DHT y Display LCD
// basado en I2C.

#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <DHT.h>
#include <DHT_U.h>
```

```
int SENSOR_TEMP = 2; // PIN para Sensor de Temperatura
int LED_TEMP = 3; // PIN para LED que indica Temperatura fuera
// de rango.
int LED_HUM = 4; // PIN para LED que indica Humedad fuera de
// rango.
int LED_OPEN = 5; // PIN para LED que indica Puerta Abierta.
int TRIG = 10; // PIN para Sensor ultrasonico (TRIGGER)
int ECO = 9; // PIN para Sensor ultrasonico (ECHO)
int DURACION; // Variable para medir duracion del retorno de la
// onda ultrasonica
int DISTANCIA; // Variable para convertir la duracion a
// distancia.
```

```
float HUMEDAD = 0; // Variable para almacenar el % de humedad
// relativa.
float TEMPERATURA = 0; // Variable para almacenar la temperatura
// del ambiente.
```

```
DHT dht(SENSOR_TEMP, DHT11); //Inicializacion del Sensor DHT
LiquidCrystal_I2C lcd(0x27, 16, 2); //Inicializacion del LCD_I2C
```

```
void setup() {
    // Establecer modo de operacion de los PIN
    pinMode(LED_TEMP, OUTPUT);
    pinMode(LED_HUM, OUTPUT);
    pinMode(LED_OPEN, OUTPUT);
    pinMode(TRIG, OUTPUT);
    pinMode(ECO, INPUT);
    // Inicio del Display
    lcd.init();
    lcd.backlight();
    // Inicio del sensor DHT11.
    dht.begin();
    Serial.begin(9600); //Inicio de puerto serial
}
```

```
void loop() {
    HUMEDAD = dht.readHumidity(); // Leer el valor de Humedad
    TEMPERATURA = dht.readTemperature(); // Leer el valor de
    // Temperatura
```

```
    lcd.clear(); // Borrar la pantalla del LCD.
    lcd.setCursor(0,0); // Colocar el cursor primera fila primera
    // columna.
    lcd.print("Temp:"); // Mostrar: Temp.
    lcd.print(TEMPERATURA); // Mostrar el valor
    // de la temperatura leida.
    lcd.write(0XDF); // Caracter especial
    lcd.print("C");
    lcd.setCursor(0,1); // Colocar el cursor segunda fila primera
    // columna.
```

```
    lcd.print("Humedad:"); // Mostrar Humedad
    lcd.print(HUMEDAD); // Mostrar el valor de la humedad leida.
    lcd.print("%RH");
    Serial.print(TEMPERATURA);
    Serial.print(",");
    Serial.print(HUMEDAD);
    delay(1000);
```

```
    if (TEMPERATURA < 18 || TEMPERATURA > 27) {
        // Si el valor de Temperatura NO esta entre 18 y 27 grados.
        digitalWrite(LED_TEMP, HIGH);
        // Encender el LED en el PIN 3 por 1 segundo.
    }
    else{// Si el valor de temperatura esta entre 18 y 27 grados
        // apagar LED.
        digitalWrite(LED_TEMP, LOW);
    }
```

```
    if (HUMEDAD < 45 || HUMEDAD > 55){
        // Si humedad NO esta entre 45 y 55% de humedad relativa.
```

```

    digitalWrite(LED_HUM, HIGH);
// Encender LED en el PIN 4 por 1 segundo.
}
else{
    digitalWrite(LED_HUM, LOW);
// Si el valor de humedad esta entre 45 y 55% apagar LED.
}

digitalWrite(TRIG, HIGH); // Generar pulso ultrasonico.
delay(1);
digitalWrite(TRIG, LOW);
DURACION = pulseIn(ECO, HIGH); // Calcular la duracion del pulso enviado.
DISTANCIA = DURACION / 58.2; // Calcular la distancia en funcion de la duracion del pulso.

if (DISTANCIA >= 10) {
// Si la distancia es mayor o igual a 10 cm.
// Borrar la pantalla indicar que la puerta esta abierta en el LCD y prender el LED 5.
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Puerta Abierta:");
    lcd.setCursor(0,1);
    lcd.print("--CIERRELA--");
    digitalWrite(LED_OPEN, HIGH);
    Serial.println(",1");
} else {
    digitalWrite(LED_OPEN, LOW); //Apagar LED si la distancia es menor a 10cm.
    Serial.println(",0");
}
}
}

```



Fig. 9. Ejecución del sistema embebido desde la IDE de Arduino (Elaboración propia)

IX. PRUEBAS DE VALIDACIÓN

La realización de las pruebas del sistema es una parte crítica en la metodología de desarrollo, por ello, siguiendo las buenas prácticas en ingeniería de software se procedió a la elaboración de casos de prueba basado en todos los posibles escenarios que se puedan durante la operación del sistema embebido, a continuación, se presenta una tabla con el listado de casos de prueba.

Módulo	Descripción	Condición de Prueba	Datos de Entrada	Resultado Esperado
--------	-------------	---------------------	------------------	--------------------

Sensor Temperatura	Verificar alerta de temperatura baja	Temperatura < 18°C	Temperatura = 16°C	LED de temperatura encendido
Sensor Temperatura	Verificar alerta de temperatura alta	Temperatura > 27°C	Temperatura = 30°C	LED de temperatura encendido
Sensor Temperatura	Verificar rango normal de temperatura	Temperatura entre 18°C y 27°C	Temperatura = 22°C	LED de temperatura apagado
Sensor Humedad	Verificar alerta de humedad baja	Humedad < 45%	Humedad = 40%	LED de humedad encendido
Sensor Humedad	Verificar alerta de humedad alta	Humedad > 55%	Humedad = 68%	LED de humedad encendido
Sensor Humedad	Verificar rango normal de humedad	Humedad entre 45% y 55%	Humedad = 50%	LED de humedad apagado
Sensor Ultrasonido	Verificar detección de puerta abierta	Distancia >= 10 cm	Distancia = 15 cm	LED de puerta abierta encendido, mensaje en LCD
Sensor Ultrasonido	Verificar puerta cerrada	Distancia < 10 cm	Distancia = 5 cm	LED de puerta abierta apagado
Display LCD	Mostrar temperatura correctamente	Temperatura medida	Temperatura = 25.5°C	Mostrar "Temp: 25.5°C"
Display LCD	Mostrar humedad correctamente	Humedad medida	Humedad = 48%	Mostrar "Humedad: 48%RH"

Comuni- cación Serial	Enviar datos de temperatura y humedad	Lectura de sensores	Temperatu- ra = 22°C, Humedad = 50%	Imprimir "22,50"
Comuni- cación Serial	Enviar estado de puerta	Sensor ultrasonido	Puerta abierta	Imprimir ",1"
Comuni- cación Serial	Enviar estado de puerta	Sensor ultrasonido	Puerta cerrada	Imprimir ",0"

Tabla. 2. Listado de casos de prueba del sistema (Elaboración propia)

Poder generar todos los escenarios para la verificación del funcionamiento es una tarea complica, dentro del alcance de este documento se desarrollaron los siguientes casos de prueba:

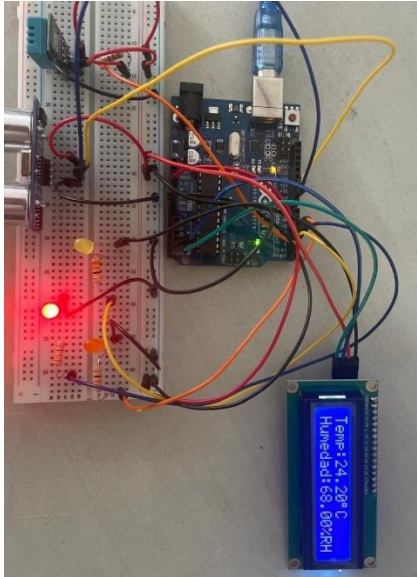


Fig. 10. Caso de prueba: humedad fuera de umbrales (Elaboración propia)

Para el caso de prueba Fig. 10. Se observa que la humedad relativa tiene un valor de 68%, sobrepasando el límite configurado de 55%, el sistema enciente el LED Rojo correspondiente a la alerta de valor de humedad fuera de rango

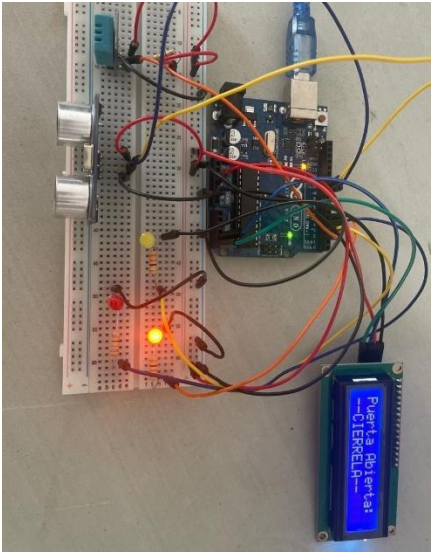


Fig. 11. Caso de prueba: Puerta abierta (Elaboración propia)
Para el caso de prueba Fig. 11. Se observa que el sensor ultrasónico no detecta un objeto (puerta), por lo tanto se activa el LED naranja.

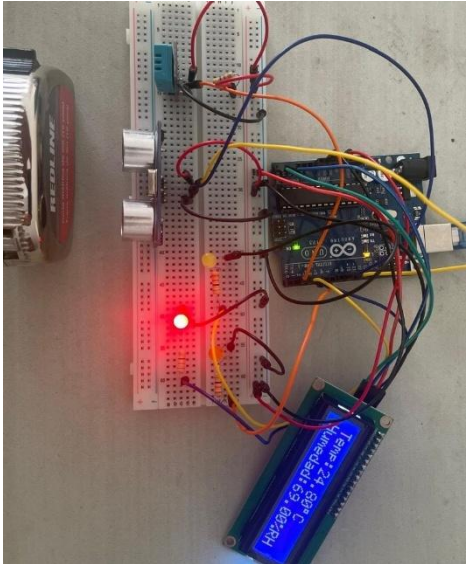


Fig. 12. Caso de prueba: Puerta Cerrada y humedad fuera de umbrales permitidos (Elaboración propia)
Para el caso de prueba Fig. 12. Se observa que el sensor ultrasónico detecta un objeto (puerta), por lo tanto no se activa el LED naranja.

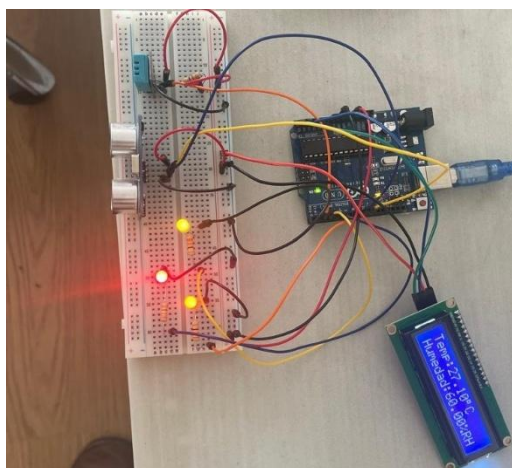


Fig. 13. Caso de prueba: Puerta abierta y humedad fuera de rango permitido, Temperatura fuera de rango permitido (Elaboración propia)

Para el caso de prueba Fig. 13. Se observa que la humedad relativa tiene un valor de 60%, sobrepasando el límite configurado de 55%, el sistema enciende el LED Rojo correspondiente a la alerta de valor de humedad fuera de rango. Se observa una lectura de 27.1°C para la temperatura, sobrepasando el límite superior configurado de 27°C, el sistema enciende el LED amarillo correspondiente a la alerta de valor de temperatura fuera de rango. Se observa que el sensor ultrasónico no tiene ningún objeto delante (puerta), el sistema enciende el LED naranja correspondiente a la alerta de puerta de gabinete abierta.

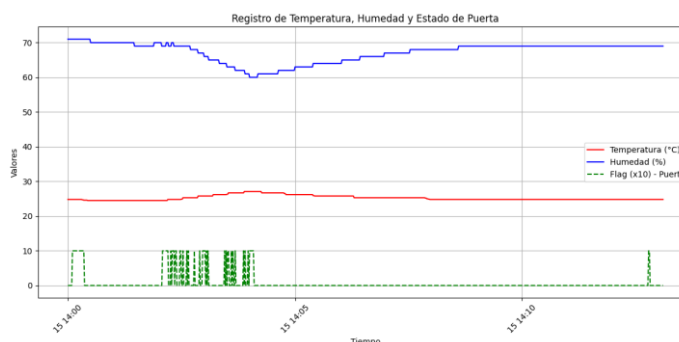


Fig. 14. Gráfico de líneas que muestra el comportamiento de las variables leídas por el sistema embebidos en un lapso de 15min aproximadamente (Elaboración propia)

Como se indica en la Fig. 3. Realizar analítica de datos es posible gracias a la comunicación serial que permite capturar en archivos log en el computador la información de las lecturas de variables de los sensores del sistema embebido en puntos en el tiempo. La Fig. 14. Muestra las 3 variables, de rojo la variación de temperatura en un rango de 15min como se visualiza tiene un comportamiento sin tanta variabilidad, también se puede observar en la figura una correlación entre la humedad y temperatura, al aumentar la temperatura, disminuye la humedad, también se observa que la entre las 2:03pm y 2:04pm las puertas del gabinete estuvieron abiertas.

X. CONCLUSIONES Y RECOMENDACIONES

1. Se logró el primer objetivo de implementar el hardware con éxito, la construcción de un prototipo que satisfaga las necesidades de medición y monitoreo de las variables críticas para un Gabinete de comunicaciones.
2. Se logró el segundo objetivo de Desarrollo de un programa que permite interactuar con los distintos sensores, pantallas y actuadores, con una lógica clara y a la vez sencilla, que permite el monitoreo y emisión de alarmas efectivas.
3. Se logró el tercer objetivo de la validación, se confirma la funcionalidad esperada. Se puede ver que, ante condiciones adversas de temperatura, el sistema emite señales visuales, tanto en pantalla como en luz, que fácilmente se identifica como un problema.
4. Se implemento un sistema de seguridad física alertando la apertura de puerta de forma visual.
5. Para futuros trabajos se recomienda la implementación actuadores que enciendan o activen equipos de control de temperatura como aire acondicionado, o maquinas deshumedecedoras de forma automática
6. Para futuros trabajos se recomienda la implementación de actuadores mecánicos como servo motores para poder realizar el cierre de puerta de los gabinetes antes determinas reglas de negocio definidas.
7. Persistir el almacenamiento, en una base de datos, de datos capturado por los sensores habilita la realización de analítica de datos, viabilizando el uso de algoritmos de Machine Learning para poder predecir, por ejemplo, estaciones con medidas de temperatura y humedad extrema de esta forma tomar acción de forma anticipada para salvaguardar el óptimo funcionamiento de nuestros equipos del data center.

XI. REFERENCIAS BIBLIOGRÁFICAS

- [1] P. Franco, J. M. Martínez, Y.-C. Kim, and M. A. Ahmed, "IoT based approach for load monitoring and activity recognition in smart homes," *IEEE Access*, vol. 9, pp. 45325–45339, 2021.
- [2] M. Esposito, A. Belli, L. Palma, and P. Pierleoni, "Design and implementation of a framework for smart home automation based on cellular IoT, MQTT, and serverless functions," *Sensors*, vol. 23, no. 9, p. 4459, May 2023.
- [3] O. Taiwo and A. E. Ezugwu, "Internet of Things-based intelligent smart home control system," *Secur. Commun. Netw.*, vol. 2021, pp. 1–17, Sep. 2021.

- [4] Y. Iliev and G. Ilieva, "A framework for smart home system with voice control using NLP methods," *Electronics*, vol. 12, no. 1, p. 116, Dec. 2022.
- [5] M. Ayaz, M. Ammad-Uddin, Z. Sharif, A. Mansour, and E. M. Aggoune, "Internet-of-Things (IoT)-based smart agriculture: Toward making the fields talk," *IEEE Access*, vol. 7, pp. 129551–129583, 2019.
- [6] K. Zheng, S. Zhao, Z. Yang, X. Xiong, and W. Xiang, "Design and implementation of LPWA-based air quality monitoring system," *IEEE Access*, vol. 4, pp. 3238–3245, 2016.
- [7] Q. Hai, C. Wang, N. Li, and C. Li, "Visual analysis of ecological hydrological intelligent process based on the
- [8] "Arduino UNO Project," Wokwi, [Online]. Available: <https://wokwi.com/>. [Accessed: 14-Dec-2024].

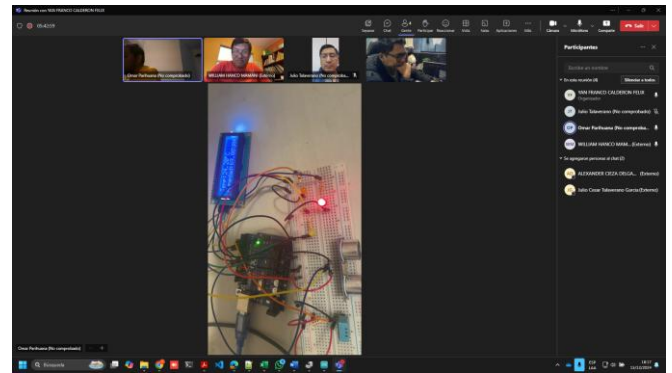


Fig. 18. Reunión 15/12/2024 presentación final

XII. ANEXOS

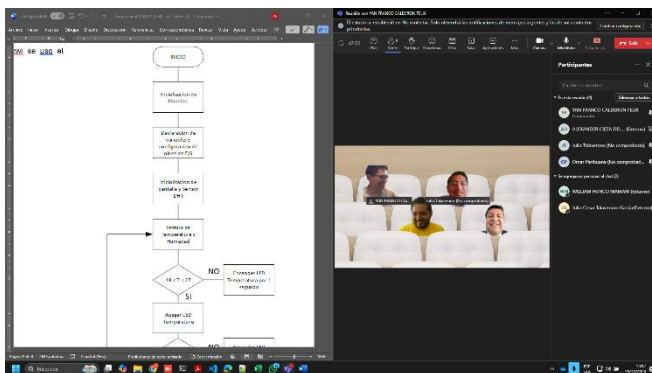


Fig. 15. Reunión 05/12/2024

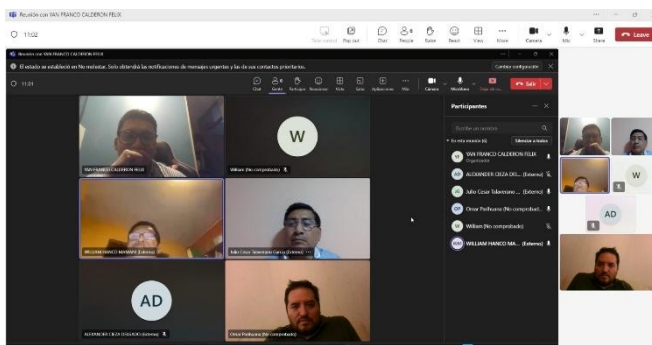


Fig. 16. reunión 12/12/2024

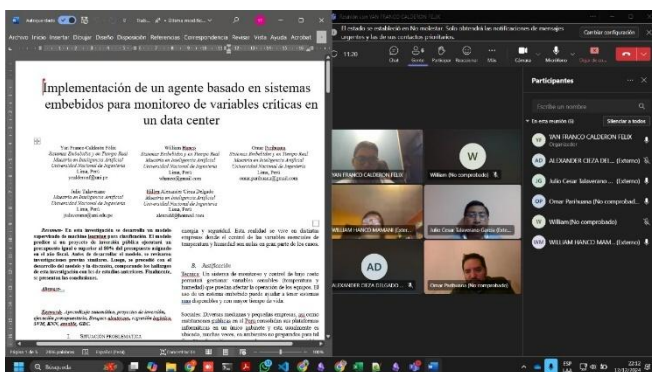


Fig. 17. Reunión 12/12/2024 presentación de Drafft documento

Proyecto y Código Fuente:

<https://github.com/yancalderon/MIA202-SisEmb>