

Universidad De San Carlos de Guatemala

Facultad de Ingeniería

Escuela de Ciencias y Sistemas

Arquitectura de Computadores y Ensambladores 2 B

Primer semestre 2025

Catedrático: Ing. Jurgén Andoni Ramírez Ramírez

Auxiliar: Axel Calderón

Proyecto Único - Fase 2



Sistema Inteligente de Monitoreo Ambiental para Cuartos de Servidores mediante IoT. FASE 2

Balam José Tomas Aju 201807028

Byron Enrique Rumpich sal 201907769

Iris Carolina Paz Guzmán 202101728

Miguel Adrian Tubac Agustin 202101927

Fecha de Entrega: 14/03/2025

INTRODUCCION

En la actualidad, los centros de datos y cuartos de servidores juegan un papel fundamental en la infraestructura tecnológica de cualquier organización. El mantenimiento de condiciones ambientales óptimas es crucial para evitar sobrecalentamientos, fallos en los equipos y problemas de seguridad.

Este proyecto tiene como objetivo desarrollar un Sistema Inteligente de Monitoreo Ambiental para Cuartos de Servidores mediante IoT, capaz de medir, analizar y reaccionar ante cambios en el entorno. Para ello, se emplearán sensores de temperatura, humedad, calidad del aire y proximidad, junto con actuadores como motores, ventiladores y alarmas.

El sistema utilizará un Arduino Mega para la captura de datos y actuadores, mientras que una Raspberry Pi procesará la información y la enviará a una base de datos en la nube a través de MQTT. Además, se integrará un módulo RFID para gestionar el acceso seguro al cuarto de servidores. La visualización de la información se realizará mediante una aplicación web, donde los usuarios podrán monitorear las condiciones ambientales en tiempo real y acceder a registros históricos para análisis de tendencias.

Con este sistema, se busca garantizar la seguridad y estabilidad de los servidores mediante un monitoreo constante y la automatización de respuestas ante situaciones críticas.

OBJETIVOS

Objetivos Generales:

1. Diseñar e implementar un Sistema Inteligente de Monitoreo Ambiental para cuartos de servidores basado en IoT, capaz de medir variables ambientales críticas, procesar los datos en tiempo real y generar alertas ante condiciones anómalas, asegurando la estabilidad y seguridad de los equipos.

Objetivos Especificos

1. Desarrollar un sistema de adquisición de datos utilizando sensores de temperatura, humedad, calidad del aire y proximidad, integrados con un Arduino Mega, para la captura y transmisión de información relevante.
2. Implementar un mecanismo de control de acceso mediante RFID y servomotores, permitiendo el ingreso seguro al cuarto de servidores y registrando los accesos en una base de datos.
3. Crear una aplicación web con dashboards interactivos, que permita visualizar en tiempo real los datos recolectados por los sensores y consultar registros históricos almacenados en la nube para el análisis de tendencias.

MARCO TEORICO

MARCO TEÓRICO

Internet de las Cosas (IoT) y su Aplicación en la Monitorización Ambiental

El Internet de las Cosas (IoT) se refiere a la interconexión de dispositivos electrónicos a través de una red para recopilar, analizar y compartir datos en tiempo real. En este proyecto, se emplea IoT para la monitorización y control de condiciones ambientales dentro de un cuarto de servidores, utilizando sensores y actuadores conectados a una unidad de procesamiento (Arduino Mega), que envía datos a una base de datos para su análisis y visualización en un dashboard web.

Microcontrolador Arduino Mega 2560

El Arduino Mega 2560 es un microcontrolador basado en el ATmega2560, diseñado para proyectos que requieren múltiples pines de entrada/salida digitales y analógicas. Su uso en este proyecto es crucial, ya que gestiona la lectura de sensores, control de actuadores y comunicación de datos.

Características del Arduino Mega 2560

54 pines digitales de I/O (15 de ellos con salida PWM).

16 entradas analógicas para la lectura de sensores.

4 puertos UART para comunicación serie.

Interfaces SPI e I2C para comunicación con módulos externos.

En este proyecto, el Arduino Mega gestiona:

Sensores: Temperatura, humedad, calidad del aire, proximidad y corriente.

Actuadores: Servomotor (control de acceso), motores de ventilación, buzzer y LEDs de advertencia.

RFID para control de acceso.

Generación de datos JSON para envío a la base de datos.

Sensores Utilizados y su Función

Sensor de Temperatura y Humedad (DHT11)

El DHT11 es un sensor que mide temperatura y humedad relativa, parámetros fundamentales en un cuarto de servidores. Su función es detectar condiciones que puedan afectar la estabilidad de los equipos y activar ventiladores o alertas en caso de niveles críticos.

Principio de Funcionamiento:

Utiliza un sensor capacitivo para medir la humedad.

Usa una resistencia de coeficiente negativo (NTC) para medir la temperatura.

```
#define DHTPIN 4
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);
float h = dht.readHumidity();
float t = dht.readTemperature();
```

Sensor de Calidad del Aire (MQ-135)

El sensor MQ-135 mide la concentración de CO₂ y gases contaminantes en el ambiente. En este proyecto, si el nivel de CO₂ supera un umbral determinado, se activa un ventilador para mejorar la circulación del aire.

Implementación en el Código:

```
int sensorValue = analogRead(A0);
float voltage = sensorValue * (5.0 / 1023.0);
gasVal = voltage * 200; // Conversión aproximada para CO2
```

```
digitalWrite(Triquer_izq, HIGH);  
delayMicroseconds(10);  
digitalWrite(Triquer_izq, LOW);  
izquierdo_duracion = pulseIn(Echo_izq, HIGH);  
izquierdo_distancia = (izquierdo_duracion / 2) / 29.1;
```

El ACS712 mide la corriente eléctrica consumida por los equipos en el cuarto de servidores. Si se detecta una caída en el suministro de energía, se activa un buzzer para generar una alerta sonora.

Implementación en el Código:

```
float sensibilidad = 0.185;  
float corriente = promedioCorriente(500);
```

Sensores de Proximidad (HC-SR04)

Los sensores ultrasónicos HC-SR04 miden la distancia de objetos cercanos mediante ondas ultrasónicas. Se usan para detectar la presencia de personas y controlar la apertura y cierre de la puerta del cuarto de servidores.

Funcionamiento:

Trigger envía un pulso ultrasónico.

Echo recibe la señal reflejada y calcula la distancia.

Implementación en el Código:

```
sensorA1 = analogRead(A1) * (5.0/1023.0);  
intensidad = (sensorA1 - 2.5) / sensibilidad;
```

Actuadores Utilizados y su Función

Servomotor (SG90 - Control de Acceso con RFID)

El servomotor se usa para controlar la barrera de acceso. Se activa cuando se escanea una tarjeta RFID válida y permanece abierto por 5 segundos, cerrándose automáticamente si no se detecta presencia.

Implementación en el Código:

```
servoMotor.attach(6);  
servoMotor.write(0);  
if (rfid.PICC_IsNewCardPresent() && rfid.PICC_ReadCardSerial()) {  
    angulo = 130;  
    servoMotor.write(angulo);  
}
```

Motores de Ventilación

Se activan automáticamente si la temperatura, humedad o CO₂ superan los niveles permitidos.

Implementación en el Código:

```
if (t >= temperatura_minima) {  
    digitalWrite(VEN_TEMP_PIN, HIGH);  
} else {  
    digitalWrite(VEN_TEMP_PIN, LOW);  
}
```

Buzzer (Alarma por Falta de Energía)

Si la corriente cae por debajo de un umbral, el buzzer se activa como una alerta sonora.




Implementación en el Código:

```
if (buzzerActivo) {  
    digitalWrite(BUZZER_PIN, LOW); // Buzzer encendido  
} else {  
    digitalWrite(BUZZER_PIN, HIGH); // Buzzer apagado  
}
```

LEDs Indicadores

Se usan para alertar sobre niveles críticos de temperatura, humedad y corriente.

Colores y Funciones:

-  Rojo: Temperatura alta.
-  Azul: Humedad alta.
-  Amarillo: Problemas de corriente.

Implementación en el Código:

```
if (t >= temperatura_minima) digitalWrite(pin_rojo, HIGH);  
if (h > humedad_minima) digitalWrite(pin_azul, HIGH);  
if (corriente > energia_minima) digitalWrite(pin_amarillo, HIGH);
```

Control de Acceso con RFID (Módulo RC522)

El módulo RFID RC522 se usa para validar la entrada de usuarios autorizados mediante tarjetas RFID.

Proceso:

El usuario acerca la tarjeta al lector.

Arduino verifica si el UID es válido.

Si la tarjeta es reconocida, el servo se activa y la puerta se abre.

Implementación en el Código:

Comunicación de Datos y Generación de JSON

Para almacenar datos en la base de datos, se genera un JSON con los valores de los sensores.

Implementación en el Código:

```
byte uid_autorizado1[] = {0x1C, 0xB1, 0xB5, 0x02};  
if (memcmp(rfid.uid.uidByte, uid_autorizado1, uid_tamano) == 0) {  
    servoMotor.write(130); // Abrir puerta  
}
```

```
doc["temperatura"] = t;  
doc["humedad"] = h;  
doc["Aire"] = gasVal;  
serializeJson(doc, json);  
Serial.println(json);
```

CONCLUSIONES

El desarrollo del Sistema Inteligente de Monitoreo Ambiental para Cuartos de Servidores permite la recolección, procesamiento y análisis de datos ambientales en tiempo real. Gracias a la integración de sensores de temperatura, humedad, calidad del aire, corriente y proximidad, el sistema puede detectar condiciones críticas y tomar decisiones automatizadas para optimizar la seguridad y eficiencia del cuarto de servidores.

El uso de actuadores como ventiladores, LEDs de advertencia, un buzzer y un servomotor permite que el sistema responda automáticamente a situaciones de riesgo, como altas temperaturas, mala calidad del aire o fallos eléctricos. La activación de estos mecanismos garantiza un ambiente adecuado para la operatividad de los servidores y la seguridad del personal.

El módulo RFID RC522 implementado en el proyecto permite la autenticación de usuarios mediante tarjetas de acceso, asegurando que solo personal autorizado pueda ingresar al cuarto de servidores. Este mecanismo, combinado con sensores de proximidad y un servomotor para la puerta, mejora significativamente la seguridad del sistema.

La generación de datos en formato JSON y su transmisión mediante Arduino facilita la integración con bases de datos en la nube para la consulta de registros históricos y análisis de tendencias. Esto permite que los datos sean accesibles desde una aplicación web, ofreciendo una interfaz intuitiva para la visualización de mediciones y la toma de decisiones en tiempo real.

BIBLIOGRAFIA

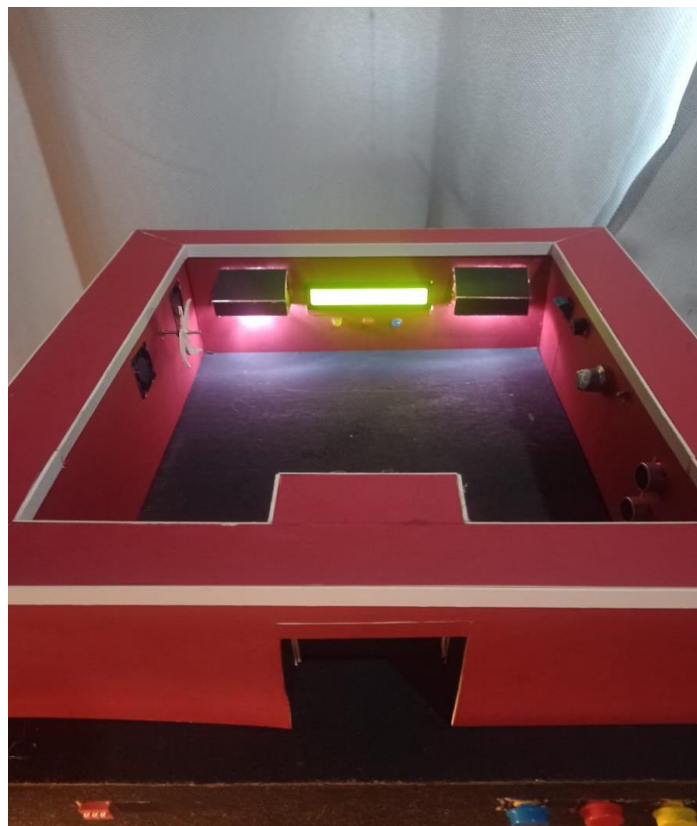
Banzy, M., & Shiloh, M. (2014). Getting started with Arduino: The open source electronics prototyping platform (3^a ed.). Maker Media, Inc.

Reas, C., & Fry, B. (2007). Processing: A programming handbook for visual designers and artists. MIT Press.

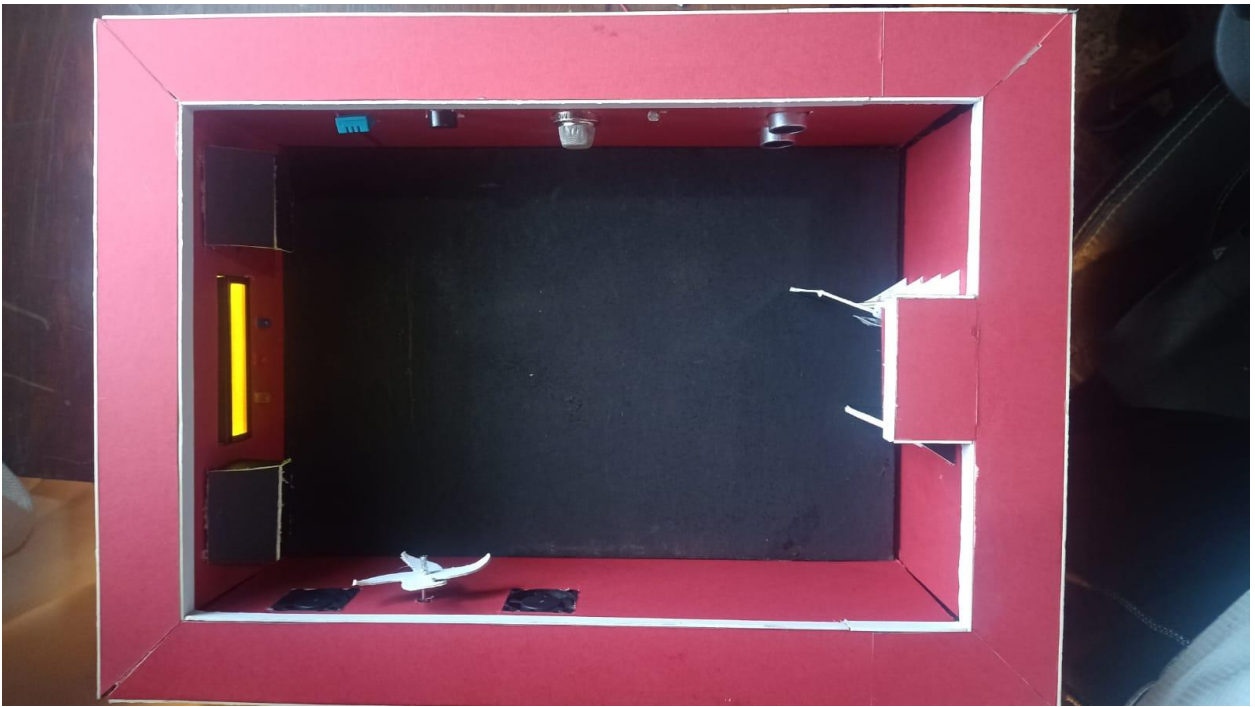
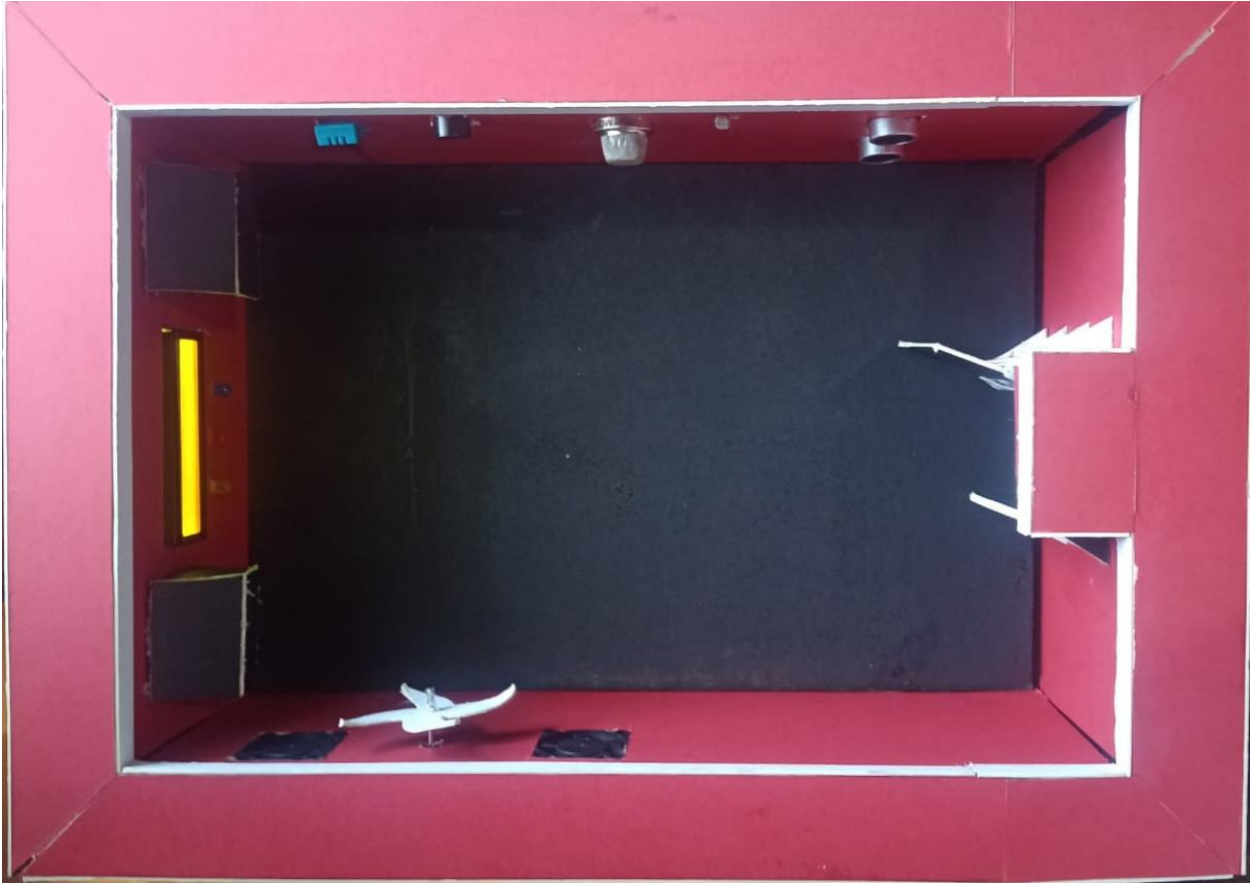
Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. Future Generation Computer Systems, 29(7), 1645–1660.

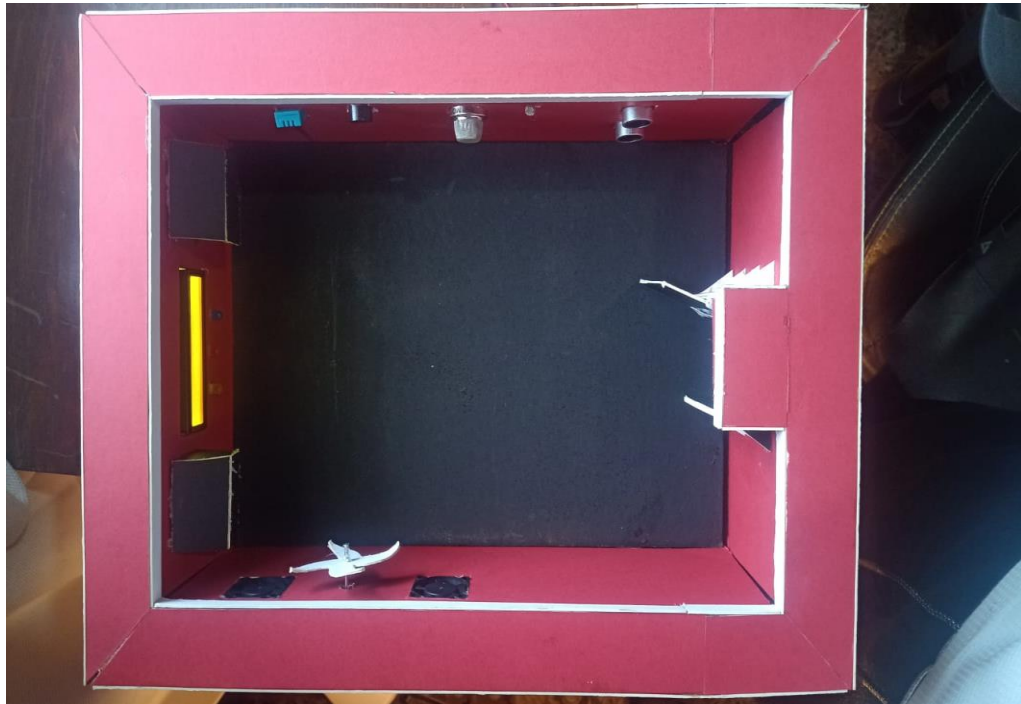
ANEXOS

1. Maqueta Fisica









Conexión de Arduino Mega

