

ANEXO

Cuestionario

Investigación de Tesis: "Diseño y construcción de un sistema domótico"

La encuesta se realiza con la finalidad de analizar el nivel de conocimiento, necesidades y el poder adquisitivo de un sector de la población de la Alcaldía Gustavo A. Madero acerca de los "Sistemas de Domótica e Internet de las cosas", que ayude a recabar información relevante en el desarrollo de tesis para la licenciatura en Ingeniería en Sistemas Electrónicos y de Telecomunicaciones.

* Indica que la pregunta es obligatoria

Estudio Socioeconómico

En las siguientes preguntas, selecciona la opción que más se adapte a tus condiciones actuales.

1. **Sexo ***

Marca solo un óvalo.

- Masculino
- Femenino Prefiero
- no decirlo Otros:
- _____

2. **Rango de edad***

Marca solo un óvalo.

- 20 - 30 años
- 30 - 40 años
- 40 - 50 años Más
- de 50 años

3. **Nivel de estudios***

Marca solo un óvalo.

- Primaria
- Secundaria
- Preparatoria
- Licenciatura
- Posgrado

4. Estado civil*

Marca solo un óvalo.

Soltero

Casado

Otros: _____

5. ¿Cuántos hijos tienes?*

Marca solo un óvalo.

No tengo

1 - 3

Más de 3

6. Aproximadamente, ¿En qué rango está tu ingreso mensual? *

Marca solo un óvalo.

Menos de \$3000

\$3000 - \$5000

\$5000 - \$10000

Más de \$10000

7. ¿Tu vivienda es propia o rentada? *

Marca solo un óvalo.

Rentada

Propia

Otros: _____

8. De la siguiente lista, selecciona los servicios con que cuenta tu vivienda *

Selecciona todas las opciones que correspondan.

Internet Residencial (Telmex, IZZI, Total Play, etc) Luz

Agua

Gas (Estacionario o cilindro)

Vigilancia (cámaras o algún sistema de seguridad)

9. ¿Cuántas habitaciones tiene tu vivienda? (Sin incluir baños) *

Marca solo un óvalo.

1 -2

3 -5

Más de 5

10. Selecciona el tipo de habitaciones con las que cuenta tu vivienda *

Selecciona todas las opciones que correspondan.

Recámara

Cocina

Comedor Sala

Baño

11. Selecciona el tipo de habitaciones con las que cuenta tu vivienda *

1

2

Más de 2

12. ¿Cuántos focos tiene tu vivienda? (incluye baños, recámaras, patios y azoteas según corresponda) *

Marca solo un óvalo.

3 - 5

6 - 10

Más de 10

13. ¿Cuántas ventanas tiene tu vivienda? *

Marca solo un óvalo.

- 1- 3
- 4 - 6
- Más de 6

14. ¿Tu vivienda cuenta con garaje propio o compartido? *

Marca solo un óvalo.

- Propio
- Compartido No
- tengo

15. ¿Cuánto de tu ingreso mensual destinas para adquirir servicios digitales dentro de tu vivienda (Internet, televisión por cable, compras por Internet, telefonía)?

Marca solo un óvalo.

- \$0 - \$1000
- \$1000 - \$2000
- Más de \$2000

16. Comúnmente, ¿Qué dispositivo utilizas para navegar por Internet? (Elige solo la opción más recurrente)

*

Marca solo un óvalo.

- SmartPhone (celular)
- Computadora
- Tablet
- Otros: _____

El Internet en tu vida.

En las siguientes preguntas, selecciona la opción que más se acerque a lo que piensas sobre el tema.

17. ¿Sabes a qué se refiere la palabra Domótica? *

Marca solo un óvalo.

- Si
-

No

- 18.** De los siguientes enunciados, según tu opinión, selecciona el que consideras que describe mejor el concepto de Domótica

Marca solo un óvalo.

- No tengo idea
- Instalación de domos en tu casa
- Conjunto de sistemas y tecnologías para automatizar tareas de una vivienda Tecnología capaz de comunicar tus cosas por Internet
- Otros: _____

- 19.** ¿Sabes a qué se le conoce como Internet de las cosas? *

Marca solo un óvalo.

- Si
- No

- 20.** De los siguientes enunciados, según tu opinión, selecciona el que consideras que describe mejor el concepto de Internet de las cosas. *

Marca solo un óvalo.

- No tengo idea
- Tecnología que permite la comunicación entre dispositivos conectados a Internet Agrupación e interconexión de dispositivos y objetos a través de una red
- Cosas que puedes buscar por Internet
- Otros: _____

- 21.** ¿Alguna vez has escuchado hablar sobre sistemas que ayudan a automatizar el hogar? *

Marca solo un óvalo.

- Si
- No

22. ¿En cuáles de los siguientes lugares, has visto este tipo de sistemas?

Selecciona todas las opciones que correspondan.

- Vivienda
- Hoteles
- Oficinas
- Centro Comercial
- Hospital
- Otros: _____

23. Enumera del 1 al 4, (siendo el 1 el más relevante y el 4 el menos relevante), la importancia * de los siguientes aspectos en tu hogar. Asigna solamente un número por columna.

Selecciona todas las opciones que correspondan.

| | Seguridad | Confort | Ahorro Energetico | Tecnologia |
|---|--------------------------|--------------------------|--------------------------|--------------------------|
| 1 | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| 2 | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| 3 | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| 4 | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

24. ¿Te interesaría invertir en una tecnología que permita controlar y mejorar los aspectos anteriores en tu hogar? *

Marca solo un óvalo.

- Si
- No

25. ¿Consideras que automatizar tu hogar es una necesidad o un lujo? *

Marca solo un óvalo.

Necesidad

Lujo Otros:

26. De los siguientes enunciados, selecciona las opciones que consideras que serían una "necesidad" tener en tu vivienda utilizando el Internet para realizar dichas tareas. *

Selecciona todas las opciones que correspondan.

- Apertura y cierre de puertas a través de Internet Apagado/encendido de
- luminaria a través de Internet
- Medición de niveles de gas dentro de tu hogar a través de Internet
- Control de temperatura a través de Internet
- Apagado/encendido de electrodomésticos a través de Internet
- Aspiradora robot con control a través de Internet
- Cámaras de vigilancia
- Cochera automática a través de Internet Asistentes
- virtuales (Alexa, Siri, Google Assistant)
- Llenado automático de cisternas o tinacos a través de Internet Sistemas de
- riego automático (Jardines o plantas) a través de Internet Sistemas de
- entretenimiento (TV, audio) a través de Internet Refrigerador inteligente
- Nada
- Otros: _____

27. De los siguientes enunciados, selecciona las opciones que consideras que serían un "lujo" tener en tu vivienda utilizando el Internet para realizar dichas tareas. *

Selecciona todas las opciones que correspondan.

- Apertura y cierre de puertas a través de Internet Apagado/encendido de
- luminaria a través de Internet
- Medición de niveles de gas dentro de tu hogar a través de Internet
- Control de temperatura a través de Internet
- Apagado/encendido de electrodomésticos a través de Internet
- Aspiradora robot con control a través de Internet
- Cámaras de vigilancia
- Cochera automática a través de Internet Asistentes
- virtuales (Alexa, Siri, Google Assistant)
- Llenado automático de cisternas o tinacos a través de Internet Sistemas de
- riego automático (Jardines o plantas) a través de Internet Sistemas de
- entretenimiento (TV, audio) a través de Internet Refrigerador inteligente
- Nada
- Otros: _____

28. ¿Con cuáles de los siguientes dispositivos inteligentes cuentas en tu vivienda? *

Selecciona todas las opciones que correspondan.

- Refrigerador inteligente
- Lavadora inteligente
- Smart TV
- Asistente Virtual (Alexa, Siri, Google Assistan, etc.)
- Focos Inteligentes
- Aire Acondicionado inteligente
- Cerradura eléctrica inteligente
- Persianas (cortinas) eléctricas
- Alarmas (contra incendios, intrusos, etc.)
- Sensor de presencia
- Control de humedad
- Otros: _____

29. De acuerdo a los dispositivos anteriores, selecciona las marcas con las que cuentas *

Selecciona todas las opciones que correspondan.

- Apple
- Steren
- Amazon
- TP-Link
- Phillips
- Huawei
- Samsung
- Siemens
- LG
- Sony Honeywll
- Belkin
- Otros: _____

30. Para intercomunicar tus dispositivos, utilizas: *

Marca solo un óvalo.

- Asistente Virtual (Alexxa, Siri, Google Assistant, otro)
- Una aplicación móvil para cada dispositivo dependiendo de la marca del fabricante
- Una aplicación móvil para todos sus dispositivos sin importar la marca del fabricante
- Nada
- Otros: _____
-

31. ¿Cómo te resulta agregar un nuevo dispositivo inteligente a tu vivienda? *

Marca solo un óvalo.

- Muy Fácil
- Fácil Difícil
- Muy difícil
-

32. De las siguientes opciones, ¿Qué proceso realizas para agregar un nuevo dispositivo a tu vivienda? *

Marca solo un óvalo.

- Descargar una nueva aplicación para configurarlo Agregarlo a
 una aplicación previamente descargada Utilizar un asistente
 virtual para configurarlo
 Todas las anteriores
 Otros: _____

33. ¿Cómo consideras que es el costo de estos dispositivos? *

Marca solo un óvalo.

- Muy Alto
 Alto Medio
 Bajo
 Muy Bajo

34. ¿Cuánto estarías dispuesto a pagar por un sistema de automatización de tu hogar? *

Marca solo un óvalo.

- Nada, no me interesa Hasta
 \$5000
 Hasta \$10000 Más
 de \$10000

35. ¿Qué tan importante/necesario te sería controlar el sistema de automatización de forma remota (desde fuera de tu casa) en tiempo real? *

Marca solo un óvalo.

- Muy importante
 Algo importante
 Importante
 Poco importante
 Nada importante

36. ¿Desde qué dispositivo te gustaría controlar las funciones de tu hogar? *

Marca solo un óvalo.

Smartphone (celular)

Tablet

Computadora

Otros: _____

37. ¿Consideras que un sistema de automatización en el hogar te permitiría ahorrar recursos *
como luz, agua, gas, etc.?

Marca solo un óvalo.

Si

No

Otros: _

38. ¿Consideras que un sistema de automatización de tareas en tu hogar, puede ayudar a prevenir accidentes? *

Marca solo un óvalo.

- Si
 No
 Otro: _____

39. ¿Crees que los sistemas de automatización le dan a tu vida comodidad respecto a las tareas cotidianas del hogar? *

Marca solo un óvalo.

- Si
 No
 Otro: _____

40. Si gustas puedes dejar algún comentario, que nos ayude a complementar tus respuestas.

Gracias por tu tiempo

Con tus respuestas, nos ayudas a tener un marco de referencia sobre el Internet de las Cosas en tu vida y poder desarrollar nuestra tesis. ¡Gracias!



UACM

¡Autonomía, educación y Libertad!

"Nada Humano me es ajeno"

Ingeniería en Sistemas Electrónicos y de Telecomunicaciones

Código

Control de Cargas y Temperatura

```

/*****
**          Librerías          **
*****/
//*****
// Anexo          //*
//*****
#include <WiFi.h> //debemos cambiar el nombre de la biblioteca para que trabaje
con la ESP32
#include <PubSubClient.h> //incluimos la librería PubSub para MQTT, aquí su
documentación https://pubsubclient.knolleary.net/api
#include <DHT.h>
/*****
**          constantes          **
*****/
// definimos los pines 32 a 34 para cada uno de los focos
#define led1 33 // foco 1 *cable verde*
#define led2 32 //foco 2 *cable amarillo*
#define led3 25 // foco 3
#define toma 26 //contacto *cable cafe*
#define puertoMqtt 1883
#define DHT_PIN 4//definimos el gpio4 para para mandar los datos del sensor dht
*cable naranja*
#define DHT_TIPO DHT22 //definimos el tipo de sensor que usamos

/*****
**          objetos y variables          **
*****/
DHT dht(DHT_PIN,DHT_TIPO); //definimos el objeto dht el cual tiene como argumentos
DHT_PIN Y DHT_TIPO
//aquí se ingresan los datos de nuestra red wifi y nuestro broker**
const char* ssid = "*****"; //** //esta variable contiene el
nombre de la red LAN (wifi) a la que nos conectaremos
const char* password = "*****"; //** //esta variable guarda la
contraseña de la LAN (wifi)
const char* mqtt_server = "192.168.1.82"; //** //en esta variable se guarda la
dirección del broker MQTT, puede ser su ip o su link
uint32_t ultimoIntentoReconexion;
uint32_t timerEnvioDatos;
//*****
WiFiClient esp32Client; //creamos el objeto que contiene el nombre del
cliente

```

```

PubSubClient client(esp32Client); //la instancia del cliente, que permite
encadenar la función

unsigned long lastMsg = 0;
#define MSG_BUFFER_SIZE (16) //definimos el tamaño máximo para guardar el
mensaje en el buffer
char msg[MSG_BUFFER_SIZE];
int value = 0;
/*****/

//declaramos la función setup_wifi() que es donde se lleva a cabo la conexión wifi
entre la esp32 y nuestra red LAN
void setup_wifi() {
  delay(10);
  // Lo primero es conectarnos a una red wifi
  Serial.println(); //hacemos un salto de renglón en el monitor serie para
comenzar a escribir
  Serial.print("Conectándose a: "); // se despliega el mensaje "Conectándose a:
dentro del monitor serie
  Serial.println(ssid); //se imprime el nombre de la red wifi a la cual se está
conectando

  WiFi.mode(WIFI_STA); //se establece el modo estación: el ESP32 se conecta a un
punto de acceso
  WiFi.begin(ssid, password); //Inicializa la configuración de red de la
biblioteca WiFi y proporciona el estado actual, se pasa el parámetro ssid y
password

  while (WiFi.status() != WL_CONNECTED) { //se devuelve el estatus de la
conexión
    delay(500); //se tiene un tiempo de espera de 500 milisegundos
    Serial.print("."); //se escriben estos puntos antes del proceso de conexión
a una red wifi
  }
  randomSeed(micros()); //inicializamos un generador de números pseudo
aleatorios cuyo parámetro Devuelve el número de microsegundos
  //desde que la placa Arduino comenzó a ejecutar el programa actual. Este
número se desbordará (volverá a cero) después de aproximadamente 70 minutos

  Serial.println("");
  Serial.println("WiFi Conectado a:"); //escribe en el monitor serie el mensaje
"WiFi Conectado a:"
  Serial.println("IP address: "); // escribe en el monitor serie el mensaje
"IP address: "
  Serial.println(WiFi.localIP()); //escribe en el monitor serie la dirección IP
del ESP32

```

```

}

//Aquí se define la función callback, la cual define el tamaño y la naturaleza del
topic y del payload, la cual se encarga de
//interpretar la información enviada e imprimirla en el monitor serie

void callback(char* topic, byte* payload, unsigned int length) {
    String topico = topic;
    //recibe al topic al cual se suscribe como una cadena de 1 byte de tamaño y al
payload o mensaje que se envía como un entero sin signo de 4bytes de longitud
    Serial.print("Mensaje recibido ["); // escribe en el monitor serie el mensaje
"Mensaje recibido ["
    Serial.print(topic); //pasamos el topic como parámetro en este mensaje
    Serial.print("] "); // completa el mensaje en el monitor serie quedando como:
"Mensaje recibido[topic]"
    for (int i = 0; i < length; i++) { //creamos un ciclo for en donde
declaramos un contador desde 0 hasta el el valor de la longitud del payload o
mensaje a enviar
        Serial.print((char)payload[i]); //guardamos la longitud del payload en un
arreglo de tamaño i
    }
    Serial.println();

    if (topico == "/led1") {

        if ((char)payload[0] == '1') { //si se recibe un "1" el pin 33 del ESP32
estará encendido
            digitalWrite(led1, HIGH);
        }
        else if ((char)payload[0] == '0') { //si se recibe un "1" el pin 33 del
ESP32 estará apagado
            digitalWrite(led1, LOW);
        }

    }
    else if (topico == "/led2") {

        if ((char)payload[0] == '1') { //si se recibe un "1" el pin 32 del ESP32
estará encendido
            digitalWrite(led2, HIGH);
        }
        else if ((char)payload[0] == '0') { //si se recibe un "1" el pin 32 del
ESP32 estará apagado
            digitalWrite(led2, LOW);
        }

    }
    else if (topico == "/led3") {

```

```

        if ((char)payload[0] == '1') { //si se recibe un "1" el pin 34 del ESP32
estará encendido
            digitalWrite(led3, HIGH);
        }
        else if ((char)payload[0] == '0') { //si se recibe un "1" el pin 34 del
ESP32 estará apagado
            digitalWrite(led3, LOW);
        }
    }
    else if (topico == "/enchufe") {

        if ((char)payload[0] == '1') { //si se recibe un "1" el pin 34 del ESP32
estará encendido
            digitalWrite(toma, HIGH);
        }
        else if ((char)payload[0] == '0') { //si se recibe un "1" el pin 34 del
ESP32 estará apagado
            digitalWrite(toma, LOW);
        }
    }
    else {
        Serial.println("error de mensaje");
    }
}

boolean reconnect() {
    // Se ejecuta un ciclo o bucle hasta que se dé la reconexión
    while (!client.connected()) {
        Serial.print("Esperando la conexión MQTT..");
        // Creamos un cliente ID aleatorio para ser asignado a la ESP32
        String clientId = "ESP32Client-";
        clientId += String(random(0xffff), HEX);
        // Se verifica que el proceso de conexión sea exitoso
        if (client.connect(clientId.c_str())) {
            Serial.println("Conectado");
            //publicamos que estamos conectados
            client.publish("/conexión", "Conectado");
            //nos suscribimos a los tópicos para controlar cada dispositivo
            client.subscribe("/led1");
            client.subscribe("/led2");
            client.subscribe("/led3");
            client.subscribe("/enchufe");
        } else {
            Serial.print("falló, rc=");

```

```

    Serial.print(client.state());
  }
  return client.connected();

  Serial.println(" try again in 5 seconds");
  // Se espera 5 segundos para volver a realizar el proceso
  delay(5000);
}
}

void setup() {
  Serial.begin(115200);
  Serial.println("iniciando Comunicación");
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
  pinMode(toma, OUTPUT);
  dht.begin(); //inicializamos dht para poder usarlo
  setup_wifi(); //llamamos a la función setup_wifi()
  client.setServer(mqtt_server, puertoMqtt); //le decimos cual es el servidor y el
puerto al que se debe conectar
  client.setCallback(callback); //el cliente pasa los parámetros de la función de
callback
}

void loop() {
  if (!client.connected()) {
    if (millis() - ultimoIntentoReconexion > 5000) {
      ultimoIntentoReconexion = millis();
      // Attempt to reconnect
      if (reconnect()) {
        ultimoIntentoReconexion = 0;
      }
    }
  }
  } else {
  //Cliente Conectado
  if (millis() - timerEnvioDatos > 5000) {
    timerEnvioDatos=millis();

    //-----
    //delay(2000);
    int temperatura=dht.readTemperature(); //guardamos el valor de temperatura
en la variable entera temperatura
    int humedad=dht.readHumidity(); //guardamos el valor de humedad en la
variable entera humedad

```

```

//-----

    snprintf (msg, 16, "%d", temperatura); //escribe la salida formateada en el
    bufer de un tamaño específico
    client.publish("/temperatura", msg); //el primer argumento es el nombre del
    topic, el segundo es el payload
    Serial.begin(115200); //
    Serial.println("Temperatura: ");//
    Serial.print(temperatura);
    Serial.println(" °C");

    snprintf (msg, 16, "%d", humedad);
    client.publish("/humedad", msg);
    Serial.println("Humedad: ");//
    Serial.print(humedad);
    Serial.println(" %");

}
client.loop();
}
}

```

Control de Acceso

```

/*****
**          Librerías          **
*****/
//*****
// Anexo          //*
//*****
#include <WiFi.h> //debemos cambiar el nombre de la biblioteca para que
trabaje con la ESP32
#include <PubSubClient.h> //incluimos la librería PubSub para MQTT aquí su
documentación https://pubsubclient.knolleary.net/api

/*****
**          constantes          **
*****/
#define openlook 33 // definimos pin para cerradura *cable verde*
#define leddoor 26 //led cerradura *cable naranja*
#define ledfire 18 //led fuego *cable café*

```

```

#define ledgas 19 //led gas *cable gris*
#define puertoMqtt 1883
#define fire 32//pin para el sensor de fuego *cable negro*
#define gas 34//Pin para el sensor de gas *cable morado*
#define ledelect 14 //pin para el led del sensor electromagnético de la
ventana *cable rojo*
#define senelect 13 //pin para la lectura del sensor electromagnético de la
ventana *cable BLANCO*

/*****
** objetos y variables **
*****/

//aquí se ingresan los datos de nuestra red wifi y nuestro broker**
const char* ssid = "*****"; //** //esta variable contiene el nombre
de la red LAN (wifi) a la que nos conectaremos
const char* password = "*****"; //** //esta variable guarda la contraseña
de la LAN (wifi)
const char* mqtt_server = "192.168.1.82"; //** //en esta variable se guarda la
dirección del broker MQTT, puede ser su IP o su link
uint32_t ultimoIntentoReconexion;
uint32_t timerEnvioDatos;
uint32_t timerEnvioDatos2;

/*****
WiFiClient esp32Client; //creamos el objeto que contiene el nombre del
cliente
PubSubClient client(esp32Client); //la instancia del cliente, que permite
encadenar la función

unsigned long lastMsg = 0;
#define MSG_BUFFER_SIZE (16) //definimos el tamaño máximo para guardar el
mensaje en el buffer
char msg[MSG_BUFFER_SIZE];

int value = 0;

/*****/

//declaramos la función setup_wifi() que es donde se lleva a cabo la conexión
wifi entre la esp32 y nuestra red LAN
void setup_wifi() {

```

```

delay(10);
// Lo primero es conectarnos a una red wifi
Serial.println(); //hacemos un salto de renglón en el monitor serie para
comenzar a escribir
Serial.print("Conectándose a: "); // se despliega el mensaje "Conectándose
a: dentro del monitor serie
Serial.println(ssid); //se imprime el nombre de la red wifi a la cual se
está conectando

WiFi.mode(WIFI_STA); //se establece el modo estación: el ESP32 se conecta
a un punto de acceso
WiFi.begin(ssid, password); //Inicializa la configuración de red de la
biblioteca WiFi y proporciona el estado actual, se pasa el parámetro ssid y
password

while (WiFi.status() != WL_CONNECTED) { //se devuelve el estatus de la
conexión
    delay(500); //se tiene un tiempo de espera de 500 milisegundos
    Serial.print("."); //se escriben estos puntos antes del proceso de
conexión a una red wifi
}

randomSeed(micros()); //inicializamos un generador de números pseudo
aleatorios cuyo parámetro Devuelve el número de microsegundos
//desde que la placa Arduino comenzó a ejecutar el programa actual. Este
número se desbordará (volverá a cero) después de aproximadamente 70 minutos

Serial.println("");
Serial.println("WiFi Conectado a:"); //escribe en el monitor serie el
mensaje "WiFi Conectado a:"
Serial.println("IP address: "); // escribe en el monitor serie el
mensaje "IP address: "
Serial.println(WiFi.localIP()); //escribe en el monitor serie la
dirección IP del ESP32
}

//Aquí se define la función callback, la cual define el tamaño y la naturaleza
del topic y del payload, la cual se encarga de
//interpretar la información enviada e imprimirla en el monitor serie

void callback(char* topic, byte* payload, unsigned int length) {
    String topico = topic;
    //recibe al topic al cual se suscribe como una cadena de 1 byte de tamaño
y al payload o mensaje que se envía como un entero sin signo de 4bytes de
longitud

```

```

    Serial.print("Mensaje recibido ["); // escribe en el monitor serie el
mensaje "Mensaje recibido ["
    Serial.print(topic); //pasamos el topic como parámetro en este mensaje
    Serial.print("] "); // completa el mensaje en el monitor serie quedando
como: "Mensaje recibido[topic]"
    for (int i = 0; i < length; i++) { //creamos un ciclo for en donde
declaramos un contador desde 0 hasta el el valor de la longitud del payload o
mensaje a enviar
        Serial.print((char)payload[i]); //guardamos la longitud del payload en
un arreglo de tamaño i
    }
    Serial.println();

    if (topico == "/openlook") {

        if ((char)payload[0] == '1') { //si se recibe un "1" el pin 33 del
ESP32 estará encendido
            digitalWrite(openlook, HIGH); //la cerradura se abre
            digitalWrite(leddoor, HIGH); //el led de la cerradura se ENCIENDE
        }
        else if ((char)payload[0] == '0') { //si se recibe un "0" el pin 33
del ESP32 estará apagado
            digitalWrite(openlook, LOW); //la cerradura se cierra
            digitalWrite(leddoor, LOW); //el led de la cerradura se APAGA
        }

    }

    else if (topico == "/fire") {

        if ((char)payload[0] == '1') { //si se recibe un "1" el pin 32 del ESP32
estará encendido
            digitalWrite(ledfire, HIGH);
        }
        else if ((char)payload[0] == '0') { //si se recibe un "1" el pin 32
del ESP32 estará apagado
            digitalWrite(ledfire, LOW);
        }

    }

    else if (topico == "/gas") {

        if ((char)payload[0] == '1') { //si se recibe un "1" el pin 34 del ESP32
estará encendido
            digitalWrite(ledgas, HIGH);
        }
    }

```

```

        else if ((char)payload[0] == '0') { //si se recibe un "1" el pin 34
del ESP32 estará apagado
            digitalWrite(ledgas, LOW);
        }
    }
    else if (topico == "/sensorventana") {

        if ((char)payload[0] == '0') { //si se recibe un "1" el pin 33 del
ESP32 estará encendido
            digitalWrite(ledeselect, LOW);
        }
        else if ((char)payload[0] == '1') { //si se recibe un "0" el pin 33
del ESP32 estará apagado
            digitalWrite(ledeselect, HIGH);
        }

    }
}

boolean reconnect() {
    // Se crea un ciclo o bucle hasta que logremos reconectarnos
    while (!client.connected()) {
        Serial.print("Esperando la conexión MQTT...");
        // Se crea un ID aleatorio para el cliente
        String clientId = "ESP32Client-";
        clientId += String(random(0xffff), HEX);
        // Se lleva a cabo el intento de conexión
        if (client.connect(clientId.c_str())) {
            Serial.println("Conectado");

            //publicamos que estamos conectados
            client.publish("/conexion", "Conectado");
            //nos suscribimos a los topicos para controlar los ledes
            client.subscribe("/openlook");           //topico de cerradura
            client.subscribe("/fire");               //topico detección de llama
            client.subscribe("/gas");                 //topico detección de gas
            client.subscribe("/sensorventana");      //topico alarma ventana
        } else {
            Serial.print("falló, rc=");
            Serial.print(client.state());
        }
    }
    return client.connected();

    Serial.println(" try again in 5 seconds");
    // Esperamos 5 segundos para volver a intentarlo nuevamente
    delay(500);
}

```

```

    }
}

void setup() {
  Serial.begin(115200);
  Serial.println("iniciando Comunicación");
  pinMode(openlook, OUTPUT);
  pinMode(ledfire, OUTPUT);
  pinMode(ledgas, OUTPUT);
  pinMode(ledeselect, OUTPUT);
  pinMode(senelect, INPUT_PULLUP);
  pinMode(leddoor, OUTPUT);

  setup_wifi();          //llamamos a la función setup_wifi()
  client.setServer(mqtt_server, puertoMqtt); //le decimos cual es el servidor
  //y el puerto al que se debe conectar
  client.setCallback(callback); //le decimos como se llama la función de
  //callback
}

void loop() {

  if (!client.connected()) {
    if (millis() - ultimoIntentoReconexion > 5000) {
      ultimoIntentoReconexion = millis();
      // Attempt to reconnect
      if (reconnect()) {
        ultimoIntentoReconexion = 0;
      }
    }
  } else if(client.connected()) {
    //Cliente Conectado
    if (millis() - timerEnvioDatos > 2500) {
      timerEnvioDatos=millis();

      delay(500);
      //Lectura de los sensores GAS Y
      //FUEGO
      int fireValor = analogRead(fire);
      int gasValor = analogRead(gas);

      char fuego;
      char gaz;

```

```

if (fireValor >= 3000){
    fuego = 0;
    Serial.println("Valor fire: ");
    Serial.println(fireValor);

} else {
    fuego = 1;
    Serial.println("Fuego detectado: ");
    Serial.println("Valor fire: ");
    Serial.println(fireValor);

}

if (gasValor <= 1500){
    gaz = 0;
    Serial.println("Valor gas: ");
    Serial.println(gasValor);

} else {
    gaz = 1;
    Serial.println("Gas detectado: ");
    Serial.println("Valor gas: ");
    Serial.println(gasValor);

}

snprintf (msg, 16, "%d",fuego ); //escribe la salida dormateada en el
//bufer de un tamaño especifico
client.publish("/fire", msg); //el primer argumento es el nombre del
//topic, el segundo es el payload
Serial.begin(115200);
Serial.println(fuego); //Envío del valor al puerto se

snprintf (msg, 16, "%d",gaz ); //escribe la salida dormateada en el bufer
//de un tamaño especifico
client.publish("/gas", msg); //el primer argumento es el nombre del
//topic, el segundo es el payload
Serial.begin(115200);
Serial.println(gaz); //Envío del valor al puerto serie

////////////////////////////////////SENSOR VENTANA////////////////////////////////////
}
else if (client.connected()) {
//Cliente Conectado
if (millis() - timerEnvioDatos2 > 5000) {
    timerEnvioDatos2=millis();

```

```
int valor = digitalRead(senelect);
Serial.begin(115200);
Serial.print("Estado: ");
Serial.print(valor); //Envío del valor al puerto serie
char bandera;
if (valor == LOW){
  Serial.println(" :Ventana Cerrada");
  bandera = 1;
  //delay(5000);
}
else{
  Serial.println(" :Ventana Abierta");
  bandera = 0;
  //delay(5000);
}

snprintf (msg, 16, "%d",bandera ); //escribe la salida formateada en el
bufer de un tamaño especifico
client.publish("/sensorventana", msg); //el primer argumento es el nombre
del topic, el segundo es el payload
Serial.println(bandera);

}
client.loop();
}
}
```

Flows NodeRed

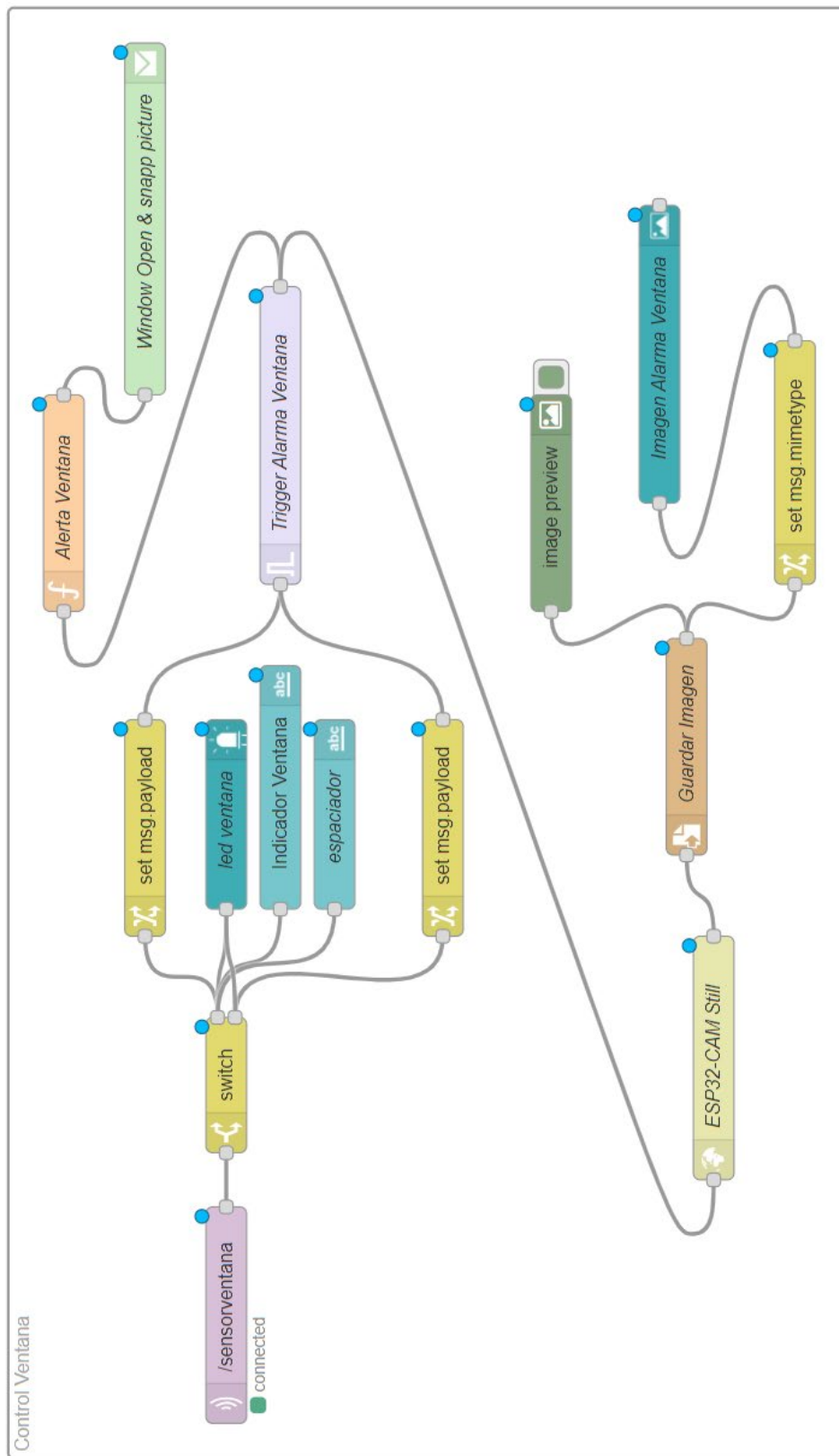


Figura 1. Nodos Control Ventana.

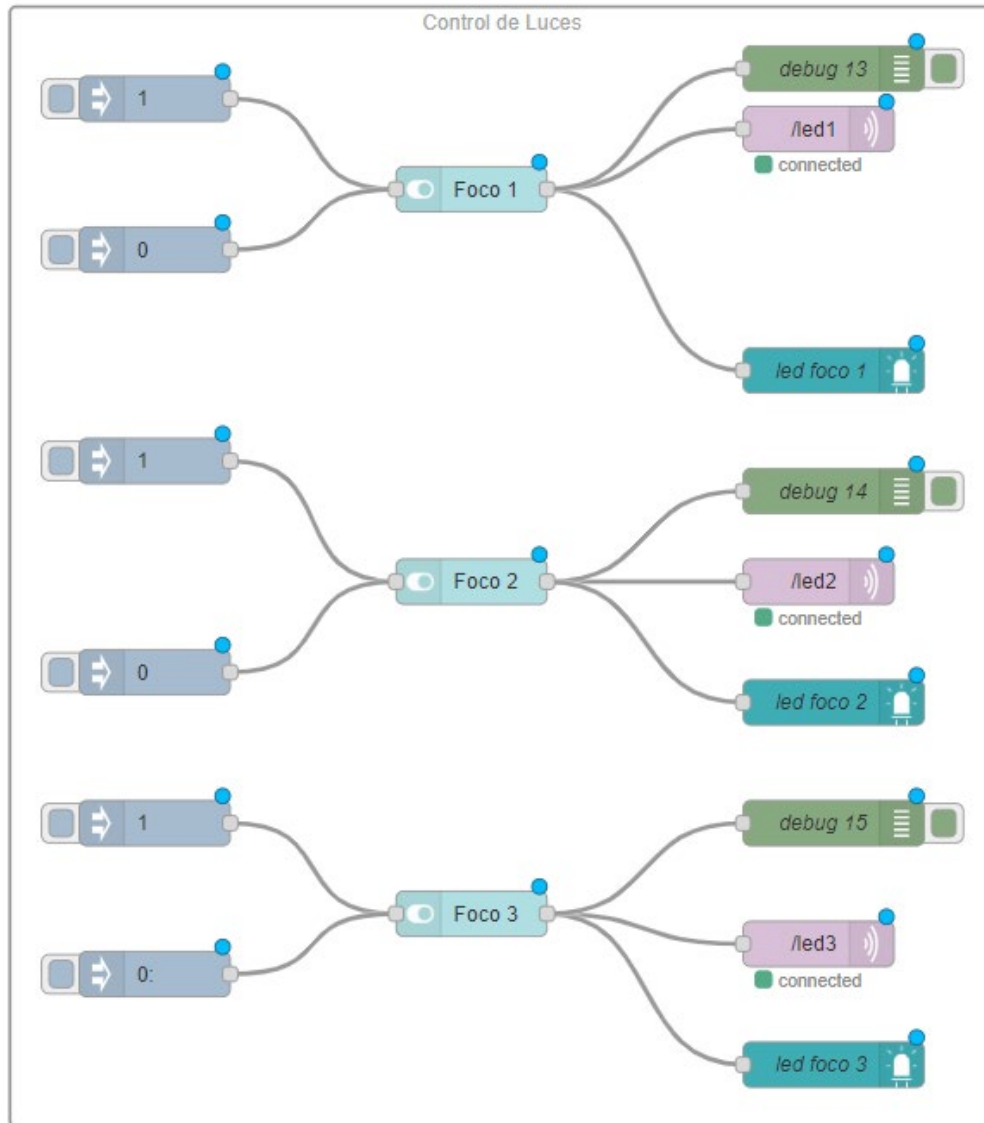


Figura 2. Control de Luces.

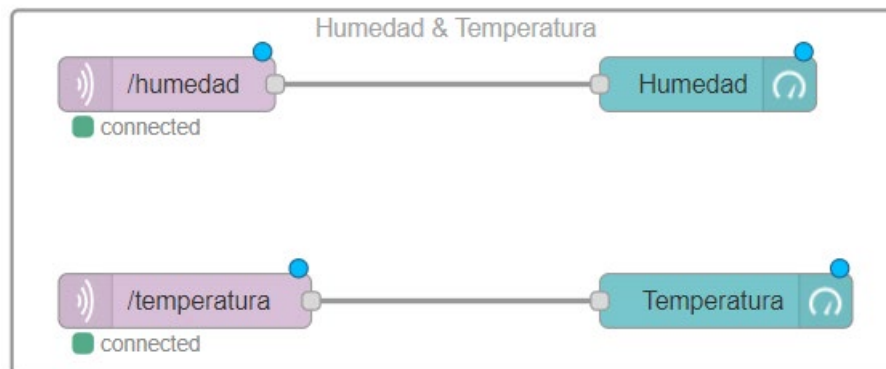


Figura 3. Sensor Temperatura y Humedad.

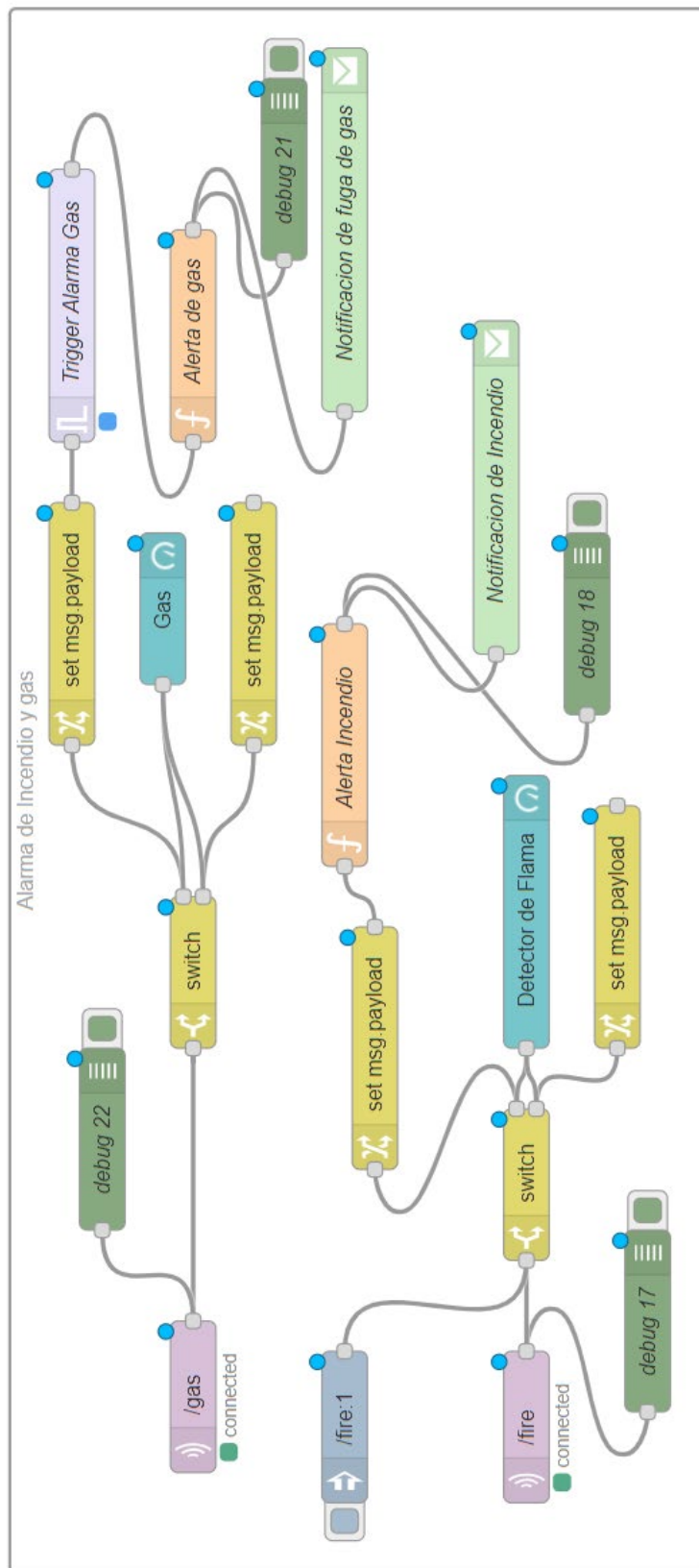


Figura 4. Detección de incendio (flama & gas).

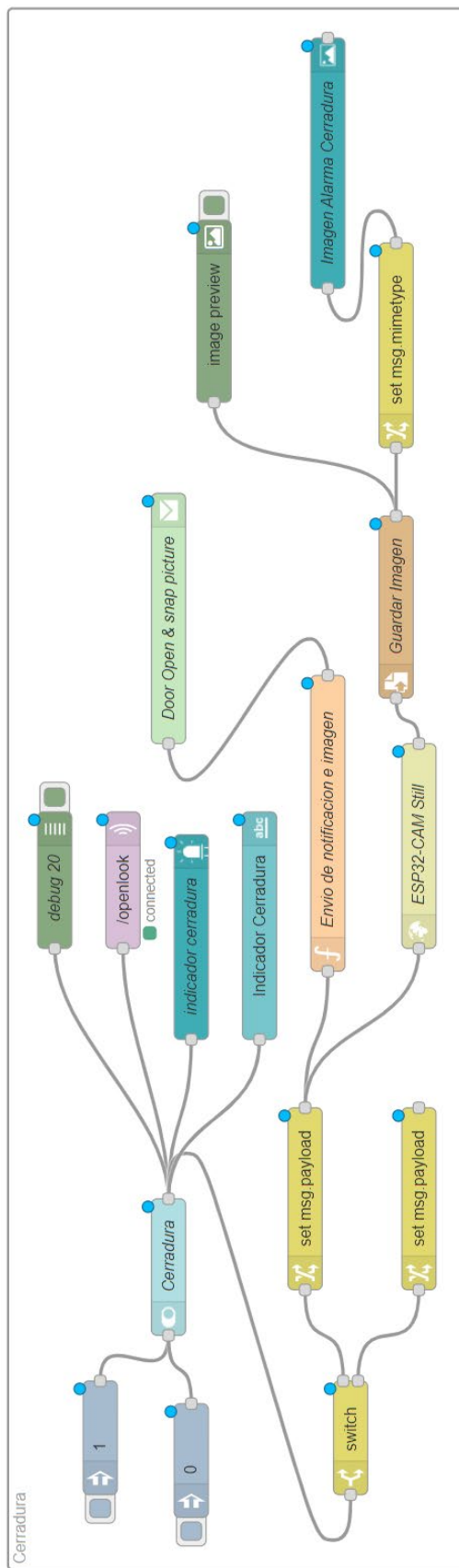


Figura 5. Control de Acceso. Cerradura.

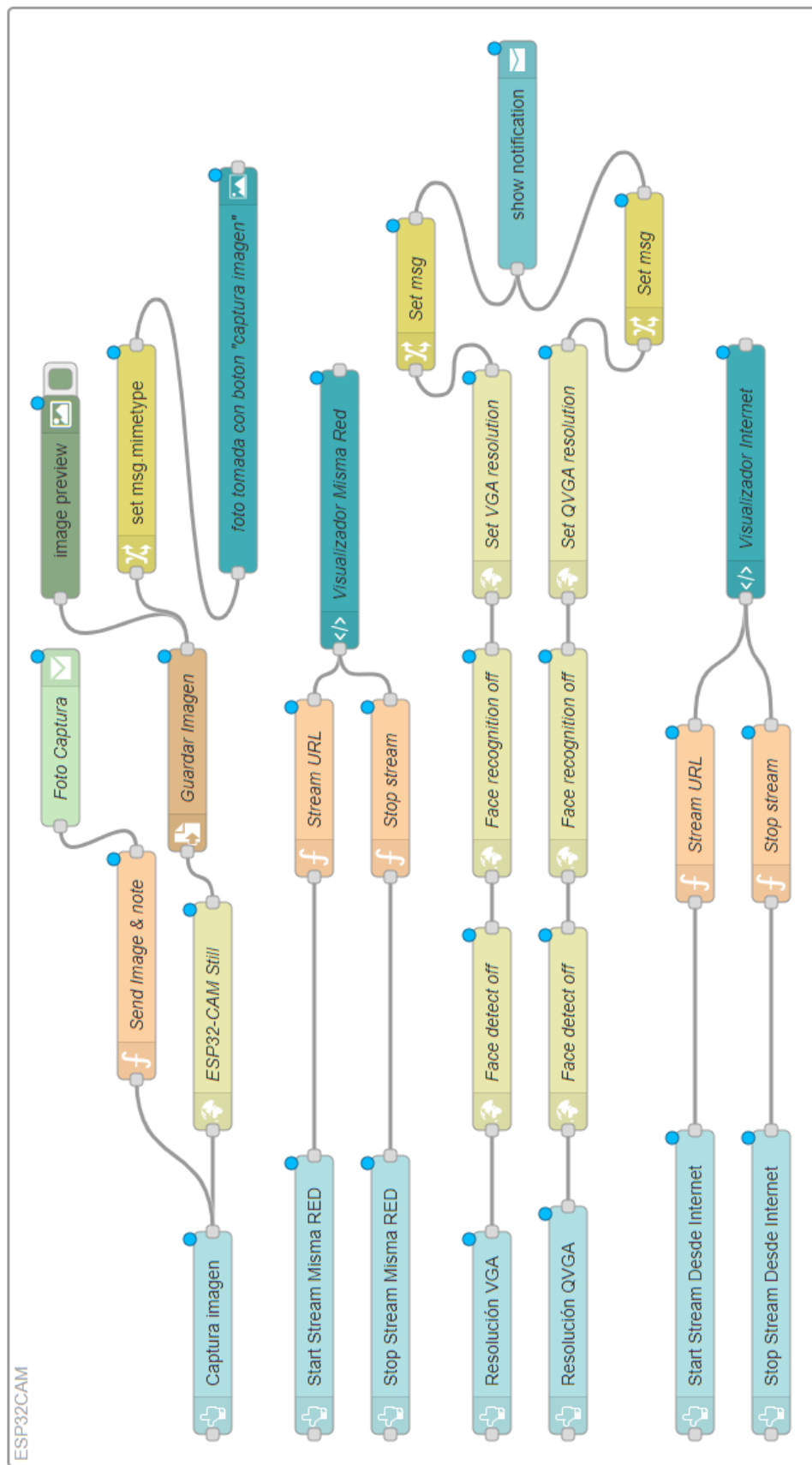


Figura 6. Cámara de Vigilancia.

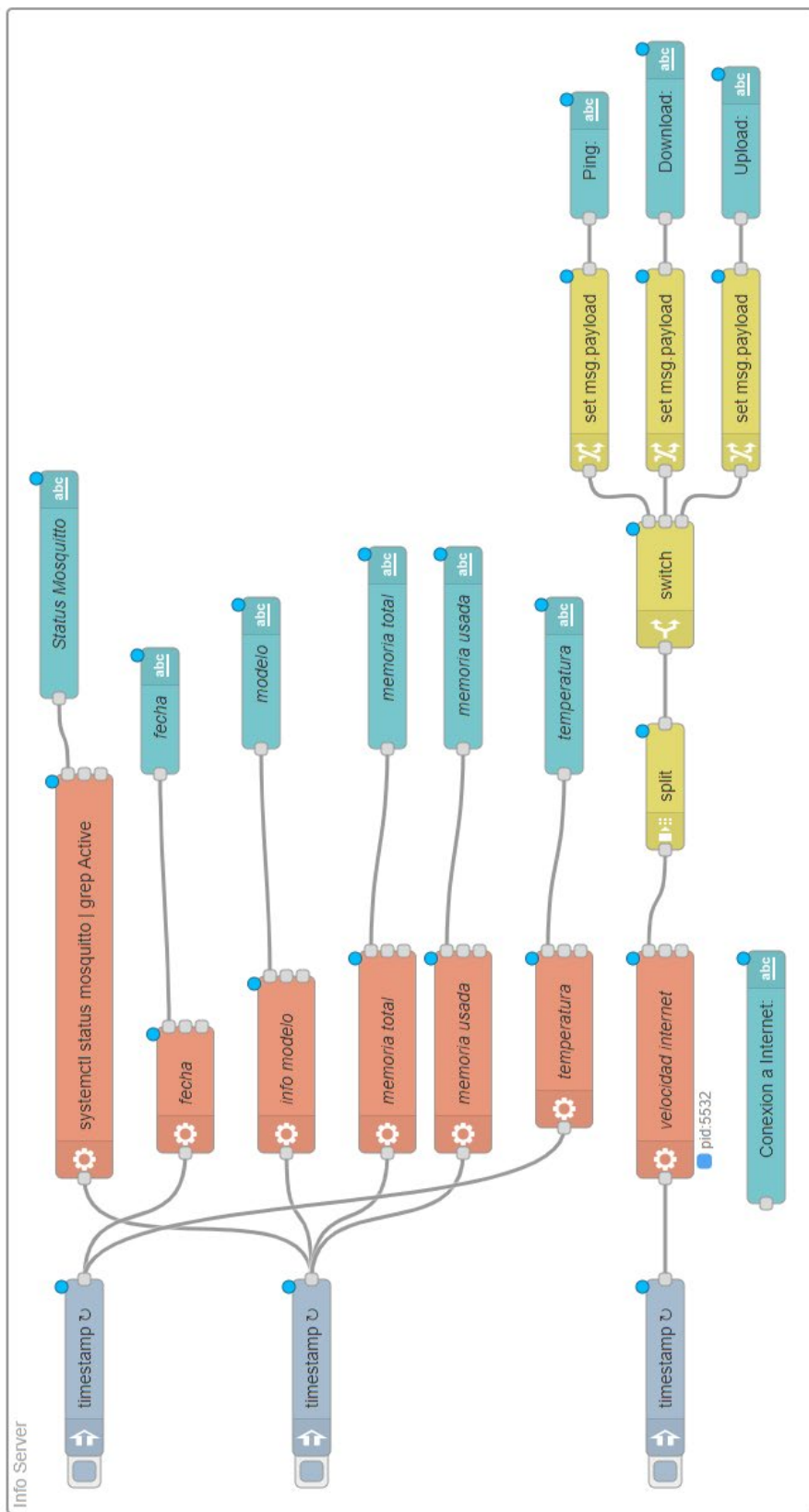


Figura 7. Información del Sistema.

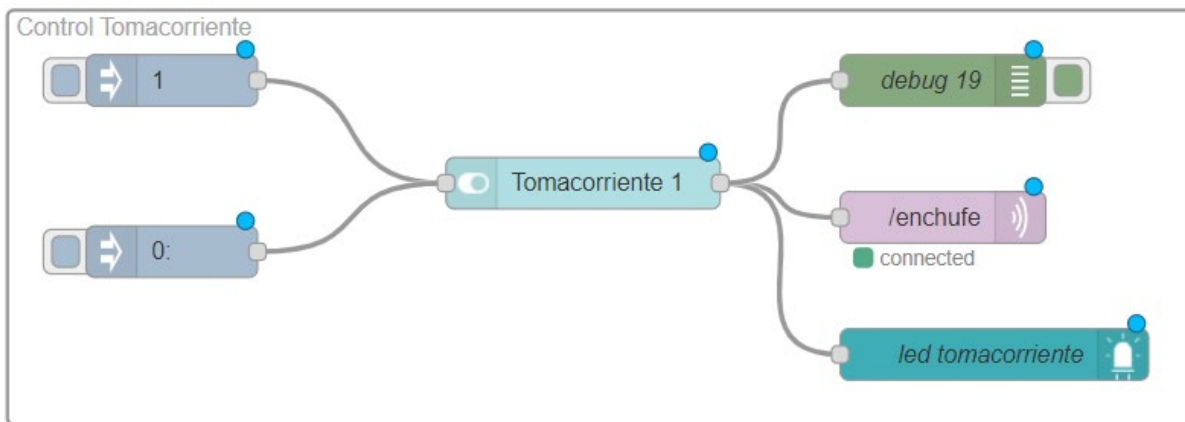


Figura 8. Control cargas Eléctricas.

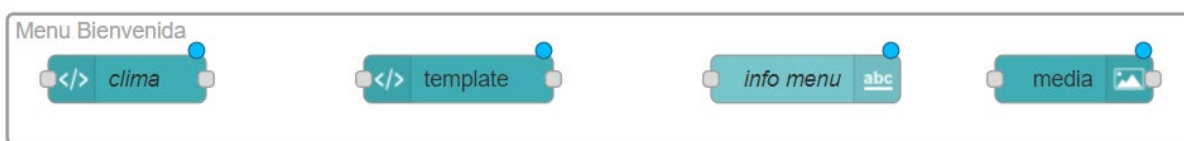


Figura 9. Menú Inicio.

Interfaz de Usuario

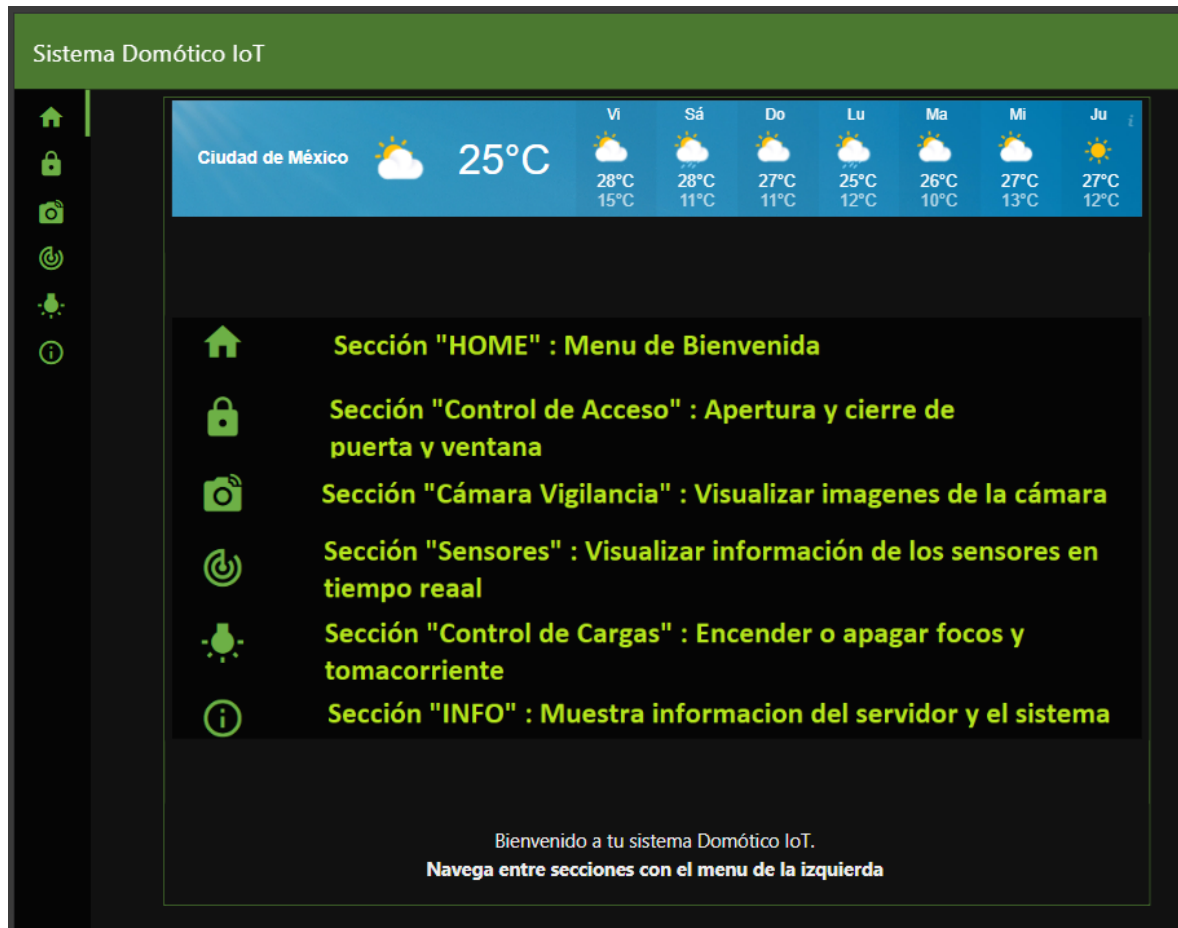


Figura 10. Menú Bienvenida

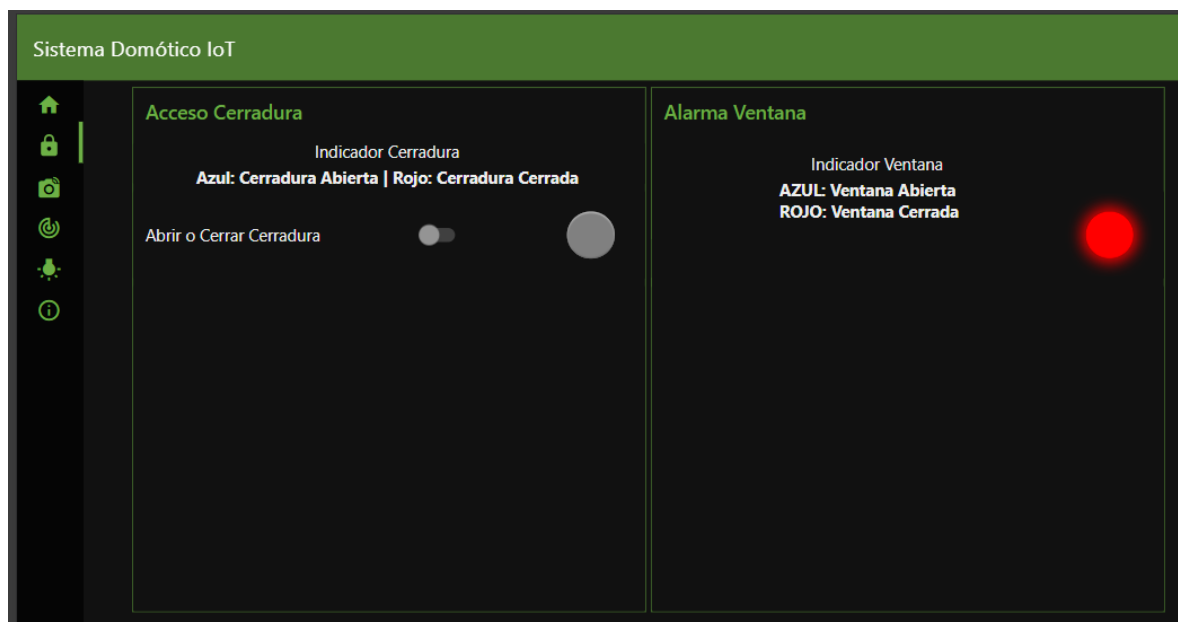


Figura 11. Control de Acceso.

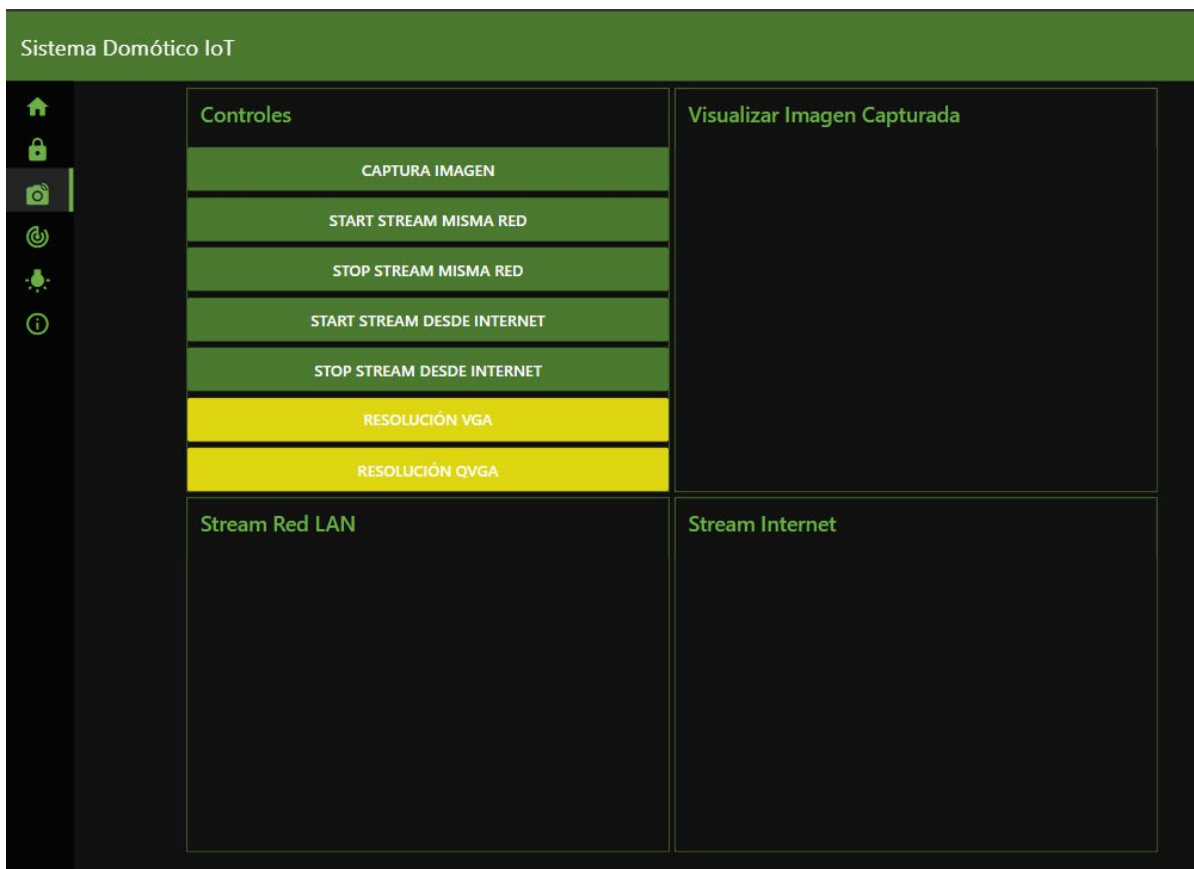


Figura 12. Control Cámara de Videovigilancia.

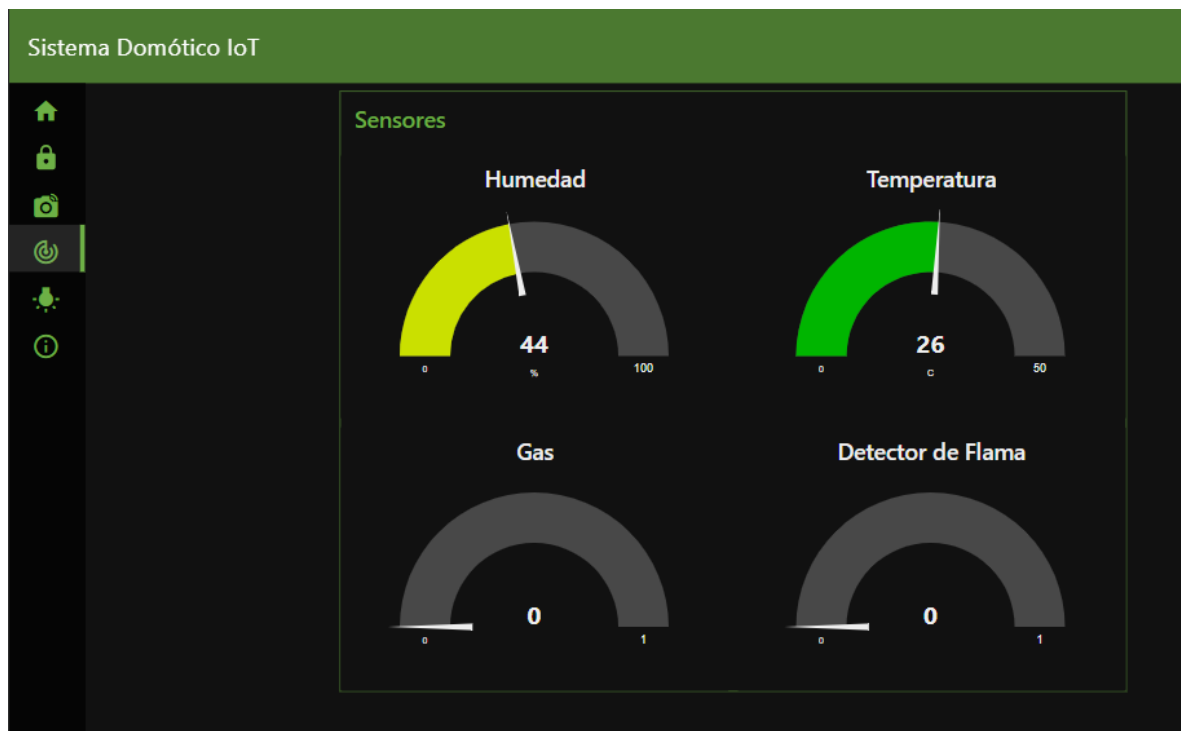


Figura 13. Sensado temperatura y Humedad, Indicador de incendio.



Figura 14. Control Cargas Eléctricas.

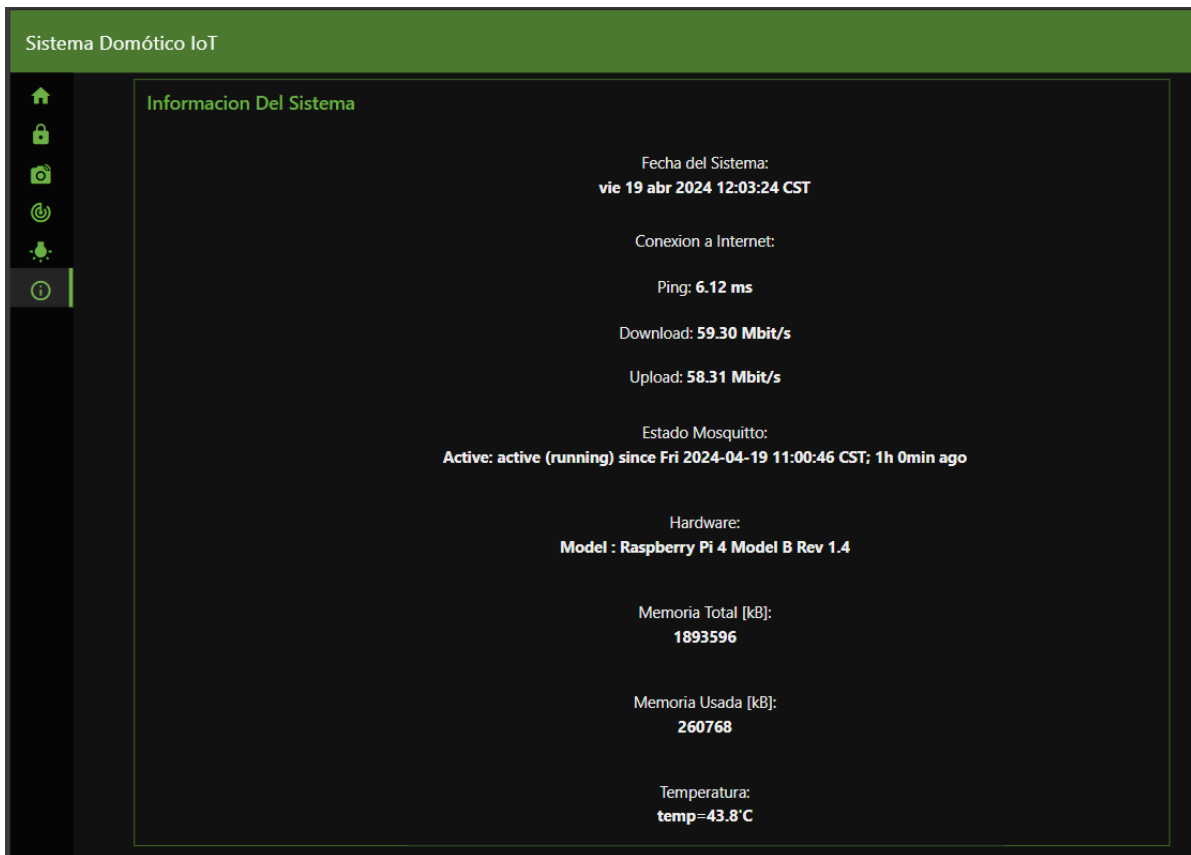


Figura 15. Información del Sistema.