*Report of the Mini Project Done in Nest Digitals*

**HOME SAFETY IOT-BASED GAS LEKAGE DETECTION SYSTEM**

**Done by: TEAM AJCE**

**Jishin Bijumon**

**Joel James**

**Rohan P Thomas**

**Sharon Thomas**

**INTRODUCTION**

Home security is today an essential matter in modern homes that needs smart and automatic systems to detect potential threats efficiently. This mini-project entails the development of an IoT-based home security system using MQ5 gas sensor, ESP8266 microcontroller, DHT11 temperature and humidity sensor, red LED, white LED, and buzzer to enhance security and threat identification. The system automatically monitors air quality and environmental conditions, particularly gas leakage, and provides real-time notifications through audio and visual signals. In normal circumstances, when no dangerous gas is present, the white LED stays constantly ON to signal a safe environment. When high concentrations of dangerous gases are detected beyond a predefined threshold, the white LED switches OFF, the red LED is turned ON immediately, and then the buzzer is also triggered after a delay of 2 seconds for immediate notifications. ESP8266 enables cloud connectivity using Firebase, which enables remote data monitoring and logging. This installation enhances home security, prevents accidents, and offers a cost-effective and scalable way for smart home automation. The project centers on reliability, effectiveness, and smooth incorporation into modern homes.

**OBJECTIVE**

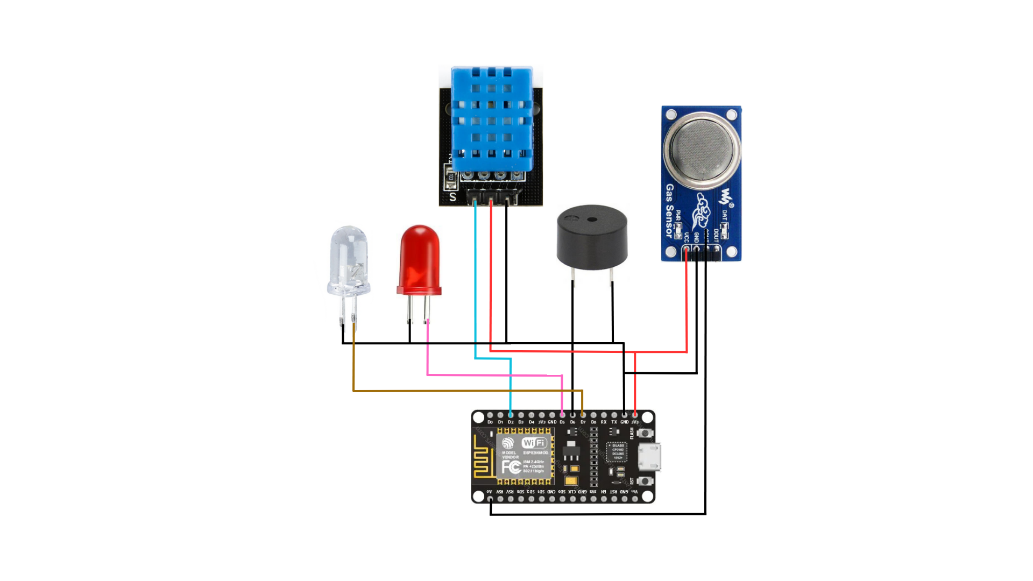
The main aim of this home security system based on IoT is to improve domestic security through real-time monitoring and detection of hazards. The system has an MQ5 gas sensor to detect toxic gas leaks, an ESP8266 microcontroller for wireless communication, a DHT11 sensor for sensing the environment, and LED & buzzer indicators for instant alerts. During normal conditions where there is no leakage of gas, a white LED is ON to show that the system is in operation and the air is safe. In the event of gas detection, the white LED will switch OFF, the warning LED will switch ON directly, and after 2 seconds, the buzzer is initiated to provide timely warnings to the occupants. Data gets logged in Firebase, facilitating remote access and notifications for proactive safety. The project aims to create an efficient, scalable, and cost-effective solution for preventing potential accidents due to gas leaks and maintaining a secure living environment.

**COMPONENTS REQUIRED**

|  |  |
| --- | --- |
| COMPONENTS | REQUIREMENT |
| ESP8266 | 1 |
| DHT11 | 1 |
| MQ5 | 1 |
| LED | 2 |
| BUZZER | 1 |
| BREAD BOARD | 1 |
| CONNECTING WIRES | As required |
| BREAD BOARD | 1 |

**WORKING PRINCIPLE**

The home safety system based on IoT works by continuously sensing environmental factors to identify gas leakage and activate effective safety notifications. The MQ5 gas sensor identifies toxic gases like LPG, H2, CH4, CO through changes in resistance as a function of gas concentration. When the system is turned ON and in normal operation without detecting any leakage of gas, a white LED is ON to show that the system is ON and the environment is safe. In case the detected level of gas is above a specific setpoint, the ESP8266 microcontroller interprets this data and switches OFF the white LED and ON a red LED instantaneously as an alarm indicator. 2 seconds later, the buzzer turns ON to provide audible signals for occupants. The DHT11 sensor also logs temperature and humidity, offering complete environmental monitoring. The ESP8266 sends the data to Firebase, where gas levels, temperature, humidity, and alert conditions are logged for remote access and monitoring. This allows users to be notified and take preventive measures effectively, making the system a potent solution for home safety and hazard prevention.

****

**WORKING OF EACH COMPONENT IN THE HOME SAFETY SYSTEM**

1. **MQ5 Gas Sensor (Gas Detection)**



* Detects flammable gases like propane, butane, and methane by measuring changes in its electrical conductivity when exposed to these gases.

The sensor's core is a tin dioxide (SnO2) sensitive layer, which exhibits lower conductivity in clean air.

* When flammable gas is present, the SnO2's conductivity increases with the gas concentration. This change in conductivity can be converted into an output signal indicating the gas concentration.

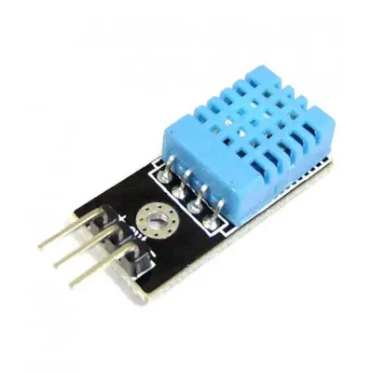
#### 

1. **ESP8266 Microcontroller**

****

* The **ESP8266** acts as the brain of the system, reading sensor values and controlling the output devices.
* It connects to **WiFi** and transmits data to **Firebase**, enabling **remote monitoring**.
* Based on sensor data, it **activates the LED and buzzer** when gas leakage is detected.

1. **DHT11 Sensor**

****

* The DHT11 measures temperature and humidity to provide additional environmental insights.
* It operates using a capacitive humidity sensor and a thermistor to determine ambient conditions.
* ESP8266 logs the readings to Firebase, allowing real-time monitoring of room temperature and air quality.

1. **LED Indicator**



* It acts as a visual warning, signaling occupants that harmful gas is present.
* Controlled by the ESP8266, it remains ON until the gas level returns to normal.
* The LED turns on immediately when MQ6 detects gas leakage.

1. **BUZZER**

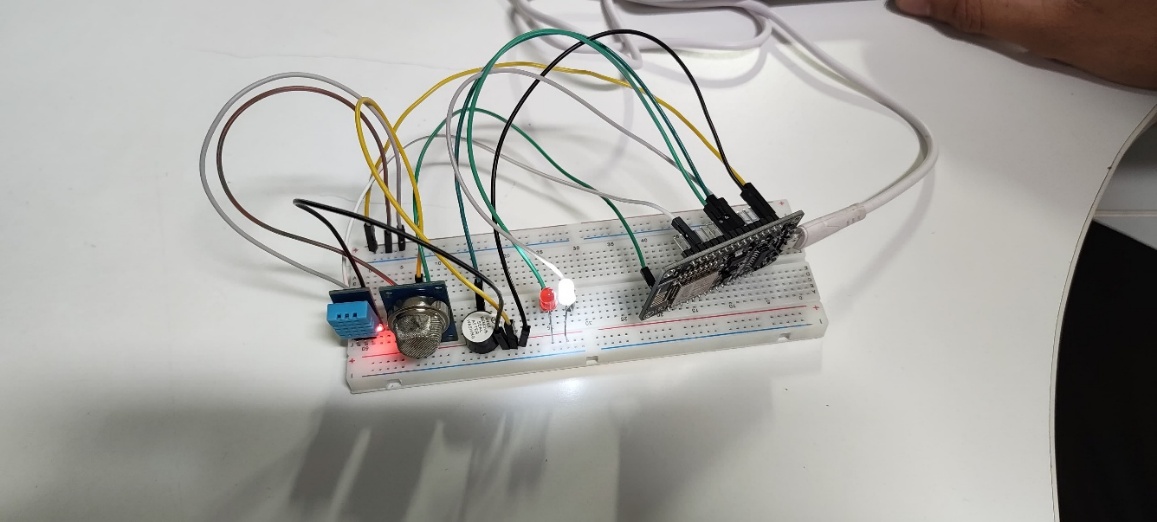
****

* After 2 seconds of LED activation, the buzzer turns on, ensuring a loud and noticeable alert.
* The delay ensures users notice the visual warning first, reducing panic while reinforcing awareness.
* The buzzer remains active until the gas concentration drops below the threshold.

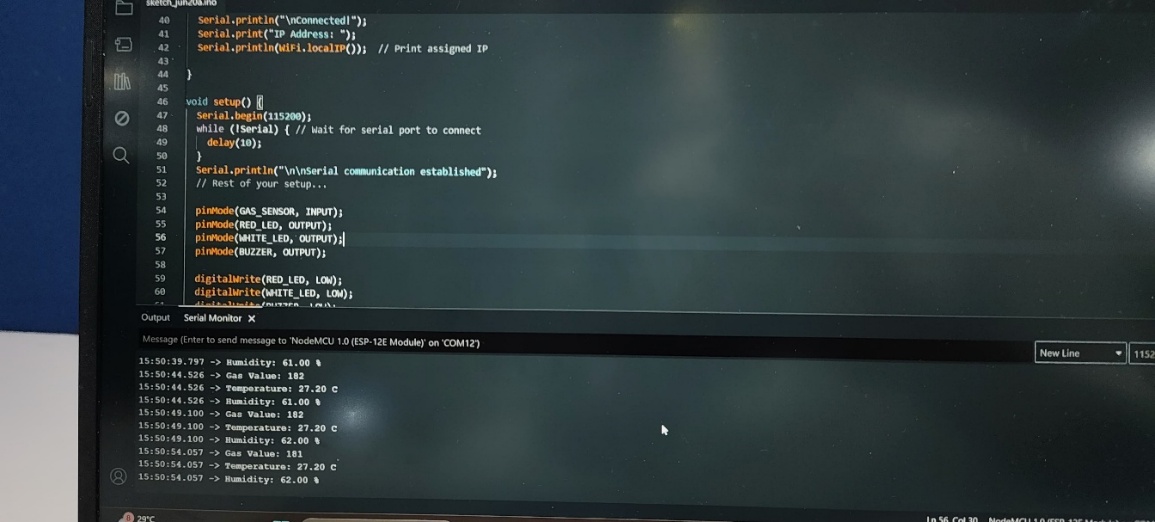
1. **Firebase Realtime Database**

* The ESP8266 **uploads gas, temperature, and humidity levels to Firebase.**
* Users can access this data remotely to ensure **home safety.**
* Firebase stores alert statuses (Gas Leak Detected or Safe), ensuring easy monitoring.

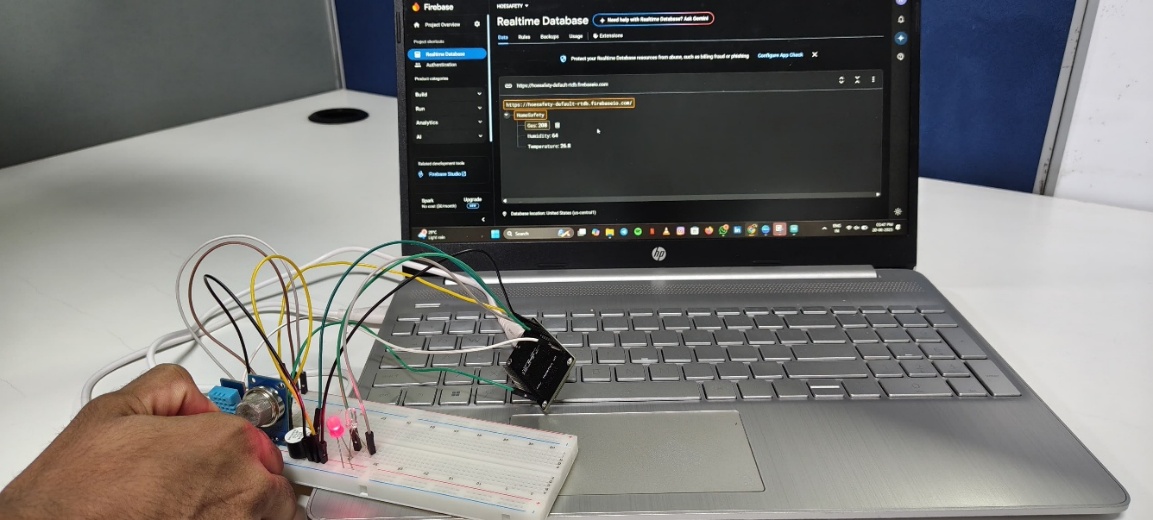
**RESULT OBTAINED**

****

**BUILT CIRCUIT**

****

**OUTPUT IN SERIAL MONITOR**

****

**OUTPUT IN FIREBASE**

**CODE FOR THE PROGRAM**

#include <ESP8266WiFi.h>

#include <Firebase\_ESP\_Client.h>

#include <DHT.h>

#include <addons/TokenHelper.h>

#include <addons/RTDBHelper.h>

// WiFi credentials

#define WIFI\_SSID "Realme 9"

#define WIFI\_PASSWORD "rp12thomas"

// Firebase credentials

#define API\_KEY "AIzaSyALYrQvipWJ40ahAm9dOhkyPVQOfifdvlE"

#define DATABASE\_URL "https://hoesafety-default-rtdb.firebaseio.com/"

#define USER\_EMAIL "rohanpthomas2004@gmail.com"

#define USER\_PASSWORD "Rp12thomas!"

// Firebase objects

FirebaseData fbd0;

FirebaseAuth auth;

FirebaseConfig config;

// Pins - Updated to match wiring diagram

#define GAS\_SENSOR A0

#define DHTPIN D2

#define DHTTYPE DHT11

#define RED\_LED D5 // Changed from original code to match diagram

#define WHITE\_LED D7 // Changed from original code to match diagram

#define BUZZER D6 // Changed from original code to match diagram

DHT dht(DHTPIN, DHTTYPE);

void connectWiFi() {

Serial.print("Connecting to WiFi");

WiFi.begin(WIFI\_SSID, WIFI\_PASSWORD);

while (WiFi.status() != WL\_CONNECTED) {

Serial.print(".");

delay(500);

}

Serial.println("\nConnected!");

Serial.print("IP Address: ");

Serial.println(WiFi.localIP()); // Print assigned IP

}

void setup() {

Serial.begin(115200);

while (!Serial) { // Wait for serial port to connect

delay(10);

}

Serial.println("\n\nSerial communication established");

// Rest of your setup...

pinMode(GAS\_SENSOR, INPUT);

pinMode(RED\_LED, OUTPUT);

pinMode(WHITE\_LED, OUTPUT);

pinMode(BUZZER, OUTPUT);

digitalWrite(RED\_LED, LOW);

digitalWrite(WHITE\_LED, LOW);

digitalWrite(BUZZER, LOW);

dht.begin();

connectWiFi();

config.api\_key = API\_KEY;

config.database\_url = DATABASE\_URL;

auth.user.email = USER\_EMAIL;

auth.user.password = USER\_PASSWORD;

Firebase.begin(&config, &auth);

Firebase.reconnectWiFi(true);

Serial.println("Firebase Connected!");

}

void loop() {

int gasValue = analogRead(GAS\_SENSOR);

float temperature = dht.readTemperature();

float humidity = dht.readHumidity();

// Print to Serial

Serial.print("Gas Value: ");

Serial.println(gasValue);

Serial.print("Temperature: ");

Serial.print(temperature);

Serial.println(" C");

Serial.print("Humidity: ");

Serial.print(humidity);

Serial.println(" %");

// Upload to Firebase

Firebase.RTDB.setInt(&fbd0, "/HomeSafety/Gas", gasValue);

Firebase.RTDB.setFloat(&fbd0, "/HomeSafety/Temperature", temperature);

Firebase.RTDB.setFloat(&fbd0, "/HomeSafety/Humidity", humidity);

// Behavior logic

if (gasValue > 200) {

digitalWrite(WHITE\_LED, LOW);

digitalWrite(RED\_LED, HIGH);

delay(2000); // wait before buzzer

digitalWrite(BUZZER, HIGH);

} else {

digitalWrite(RED\_LED, LOW);

digitalWrite(BUZZER, LOW);

digitalWrite(WHITE\_LED, HIGH);

}

delay(3000); // loop delay

} }

**Applications of the Home Safety System**

1. **Gas Leakage Detection:** Provides real-time monitoring for homes, kitchens, and industries to prevent hazardous gas leaks.
2. **Smart Home Automation:** Integrates with IoT systems for automated alerts and responses.
3. **Industrial Safety:** Used in factories handling combustible gases to enhance workplace safety.
4. Health Protection: Prevents respiratory issues caused by exposure to toxic gases.
5. **Remote Monitoring:** Enables users to track gas levels, temperature, and humidity from anywhere via Firebase cloud.
6. **Early Fire Hazard Warning:** Detects potential fire risks due to gas leaks, preventing accidents.
7. **Home Security Enhancement:** Works alongside other smart security devices for improved safety.

### **ADVANTAGES OF THE SYSTEM**

### **Real-Time Alerts:** Immediate gas detection and warning through LED and buzzer.

### **Remote Accessibility:** Users can monitor and receive notifications via Firebase from any location

### **IoT Integration:** Compatible with smart home automation for enhanced security.

### **Cost-Effective & Scalable:** Affordable sensors and microcontrollers make it easy to implement.

### **Automated Safety Measures:** No manual intervention required; system responds autonomously

### **Low Power Consumption:** Energy-efficient components ensure minimal power usage.

### **User-Friendly Design:** Simple setup and operation for homeowners and businesses.

### **CONCLUSION**

The IoT-based home safety system effectively showcases a cost-efficient and automated method to sensing gas leakage and tracking environmental parameters. With the integration of an MQ5 gas sensor, ESP8266 microcontroller, DHT11 temperature & humidity sensor, red LED, white LED, and buzzer, the system sends real-time alerts for impending threats. When the system is powered on and in safe conditions with no gas present, the white LED is ON to show that the system is on and the environment is safe. The white LED will turn OFF upon gas detection, the red LED will turn on as soon as it detects gas, and the buzzer will start ringing after 2 seconds, giving timely alerts. Moreover, Firebase cloud integration provides remote monitoring, where users are notified of home safety status remotely. The project increases domestic security and is a cost-saving and scalable remedy against accidents occasioned by gas leaks. It demonstrates the capabilities of IoT in smart home automation, with an accent on reliability, accessibility, and efficiency. Next improvements might involve mobile app notifications through Firebase Cloud Messaging (FCM) or automatic ventilation control to actively prevent hazards.