



NOSLEEP,
INC.

SMART HOME AUTOMATION

INEED2REST.COM

CONTENT

01	INTRODUCTION	08
02	COMPONENTS USED	09
03	FRAMEWORKS AND PLATFORMS	10
04	CIRCUIT DESIGN	11
05	THEORETICAL BASIS	12
06	WIFI AND MQTT SETUP	13
07	FIREBASE INTEGRATION	14
	SENSOR FUNCTIONALITY	
	USER INTERFACE	
	LOGIC AND CONTROL FLOW	
	ALARM AND NOTIFICATION SYSTEM	
	SERVO CONTROL MECHANISM	
	OUTPUT AND RESULTS	
	CONCLUSION AND FUTURE WORK	

PROJECT OVERVIEW

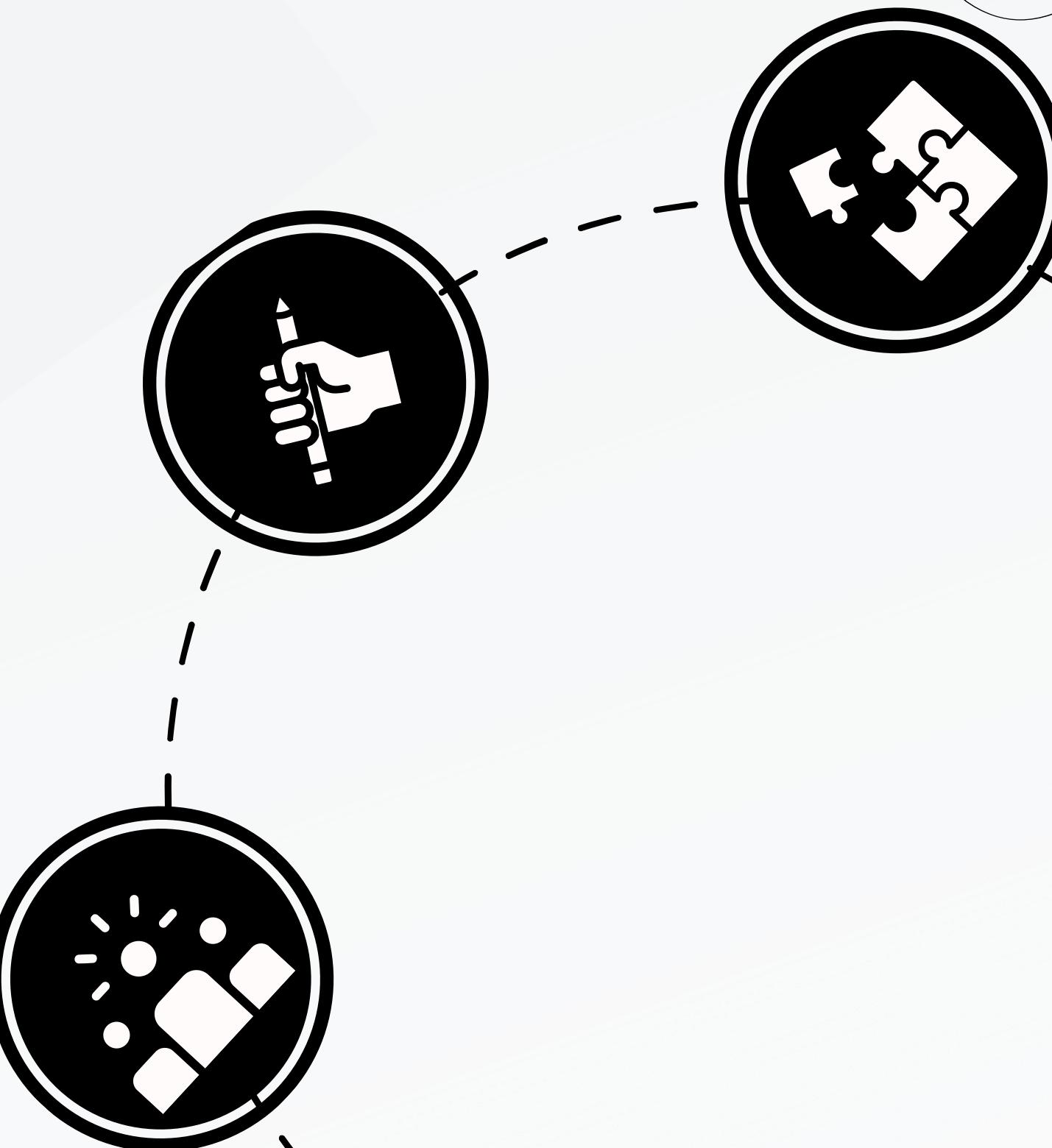


This Arduino code implements a smart home automation system using an ESP32 microcontroller. It integrates various sensors, a keypad for user input, an LCD for displaying information, and MQTT for communication with a mobile app and Firebase.



LIBRARY INCLUSIONS

```
#include <WiFi.h>
#include <PubSubClient.h>
#include <FirebaseESP32.h>
#include <DHT.h> #include <Keypad.h>
#include <LiquidCrystal_I2C.h>
#include <Servo.h>
```



WIFI AND MQTT CONFIGURATION

Mission

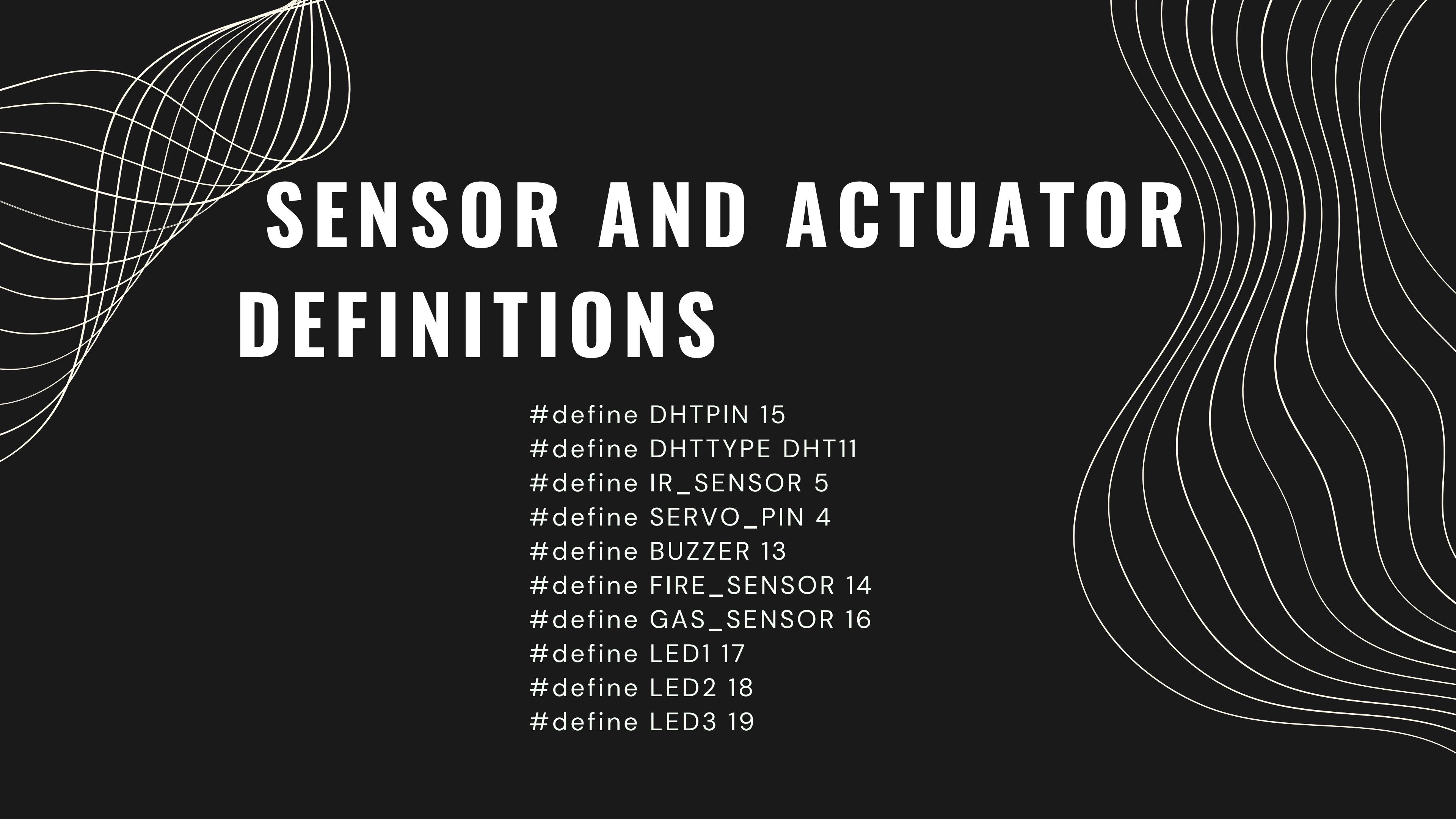


- WiFi Credentials: Replace ##### ###### with your WiFi SSID and password.
- MQTT Server: Connects to the public HiveMQ broker.
- Firebase Configuration: Set the database URL and API key for Firebase integration.

```
const char* ssid = "##### ######";  
const char* password = "##### ######";  
const char* mqtt_server = "broker.hivemq.com";  
const int mqtt_port = 1883;  
#define DATABASE_URL "YOUR_FIREBASE_DATABASE_URL"  
#define API_KEY "YOUR_FIREBASE_API_KEY"
```

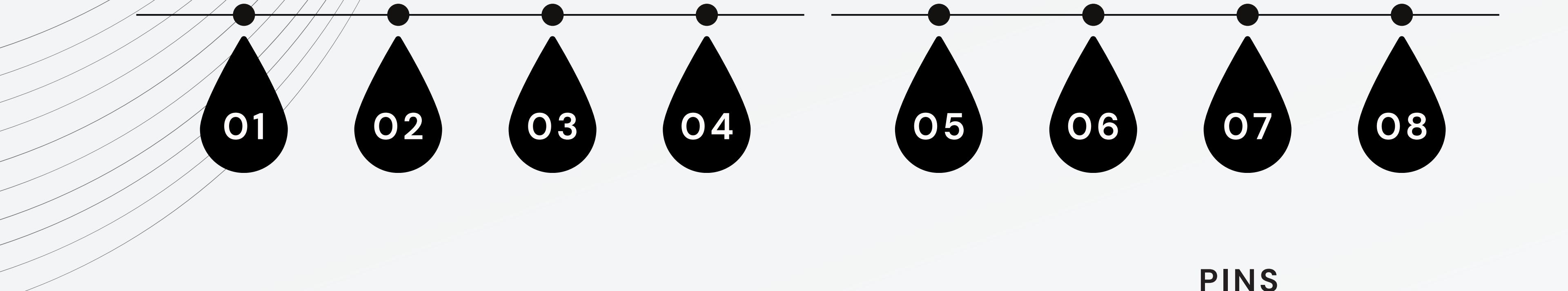
Vision





SENSOR AND ACTUATOR DEFINITIONS

```
#define DHTPIN 15
#define DHTTYPE DHT11
#define IR_SENSOR 5
#define SERVO_PIN 4
#define BUZZER 13
#define FIRE_SENSOR 14
#define GAS_SENSOR 16
#define LED1 17
#define LED2 18
#define LED3 19
```

A diagram showing a keypad grid with 4 rows and 4 columns of buttons. Above the keypad, there are 8 black teardrop shapes labeled 01 through 08 from left to right. A horizontal line with 9 black dots connects the top of each teardrop to a single horizontal line at the top of the slide. Lines also connect the bottom of each teardrop to the corresponding button on the keypad grid.

01

02

03

04

05

06

07

08

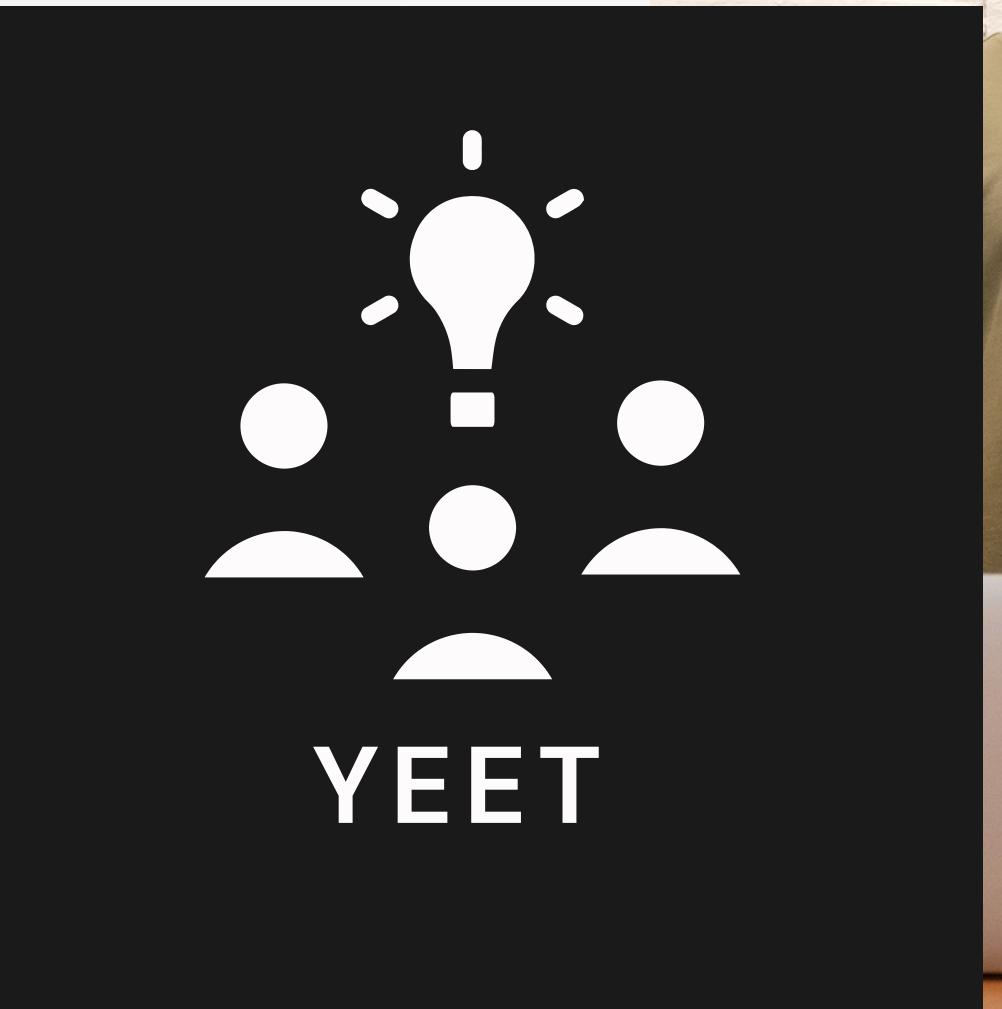
PINS

KEYPAD CONFIGURATION

```
const byte ROWS = 4;  
const byte COLS = 4;  
char keys[ROWS][COLS] = {{'1', '2', '3', 'A'}, {'4',  
'5', '6', 'B'}, {'7', '8', '9', 'C'}, {'*', '0', '#', 'D'}};  
byte rowPins[ROWS] = {27, 26, 25, 33};  
byte colPins[COLS] = {32, 35, 34, 39}; Keypad  
keypad = Keypad(makeKeymap(keys), rowPins,  
colPins, ROWS, COLS);
```

SETUP FUNCTION

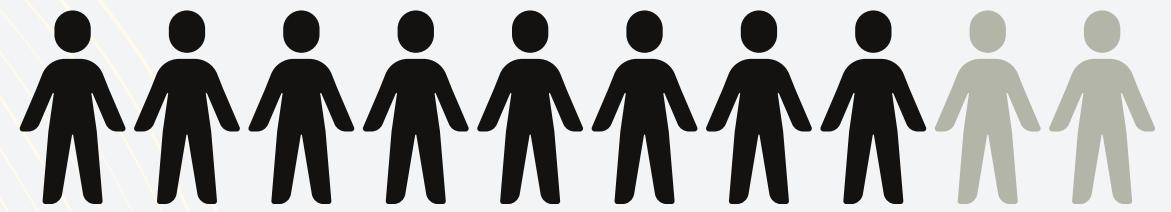
```
void setup() {  
Serial.begin(115200);  
pinMode(IR_SENSOR, INPUT);  
pinMode(BUZZER, OUTPUT);  
pinMode(FIRE_SENSOR, INPUT);  
pinMode(GAS_SENSOR, INPUT);  
pinMode(LED1, OUTPUT);  
pinMode(LED2, OUTPUT);  
pinMode(LED3, OUTPUT);  
lcd.init();  
lcd.backlight();  
servo.attach(SERVO_PIN);  
dht.begin();  
connectToWiFi();  
client.setServer(mqtt_server, mqtt_port);  
client.setCallback(callback);  
Firebase.begin(&config, &auth);  
}
```



LOOP FUNCTION

```
void loop() {  
if (!client.connected()) {  
reconnect();  
} client.loop(); // Read DHT sensor  
float humidity = dht.readHumidity();  
float temperature =  
dht.readTemperature(); // Publish and  
send sensor data  
}
```

80%



SLEEPING

EMERGENCY HANDLING

The code checks for fire and gas detection, triggering alarms and notifications:

```
if  
(digitalRead(FIRE_SENS  
OR) == HIGH) {  
emergencyActive = true;  
triggerAlarm("Fire  
detected! Alarm  
activated!");}
```

FIRST

```
if  
(digitalRead(GAS_SENS  
OR) == HIGH) {  
emergencyActive = true;  
triggerAlarm("Gas  
detected! Opening  
door!"); openDoor();}
```

SECOND

You will go into
emergency mode, each
having a unique
responds like opening
door, etc.

THIRD

USER AUTHENTICATION



Gate

```
if (userDetected && !emergencyActive) { char  
key = keypad.getKey(); if (key) { inputPassword  
+= key; // Check password } }
```



User

Password 1234. Correct pass!
Password 1221. Tried again.

FUNCTIONS FOR ACTIONS



Controls the servo to open
the door.

OPENDOOR():



Activates the buzzer and
sends notifications.

TRIGGERALARM():



Publishes messages to MQTT
and Firebase.

SENDNOTIFICATION():

SUMMARY OF FEATURES

- Real-time Monitoring: The system continuously monitors environmental conditions and user presence.
- User-Friendly Interface: The LCD and keypad provide an intuitive way for users to interact with the system.
- Emergency Response: Automated responses to detected hazards enhance home safety.
- Remote Notifications: Users receive alerts on their mobile devices, ensuring they are informed of any issues even when away from home.
- Future Enhancements: Potential improvements could include adding more sensors, integrating with additional smart home devices, and enhancing the mobile app interface for better user experience.

CONCLUSION

This code integrates various components to create a smart home system that can monitor environmental conditions, detect emergencies, and provide user interaction through a keypad and LCD display. The use of MQTT and Firebase allows for real-time data communication and notifications.

THANK'S FOR WATCHING

Yea Goodbye and Goodnight.

