

RetoEsp32

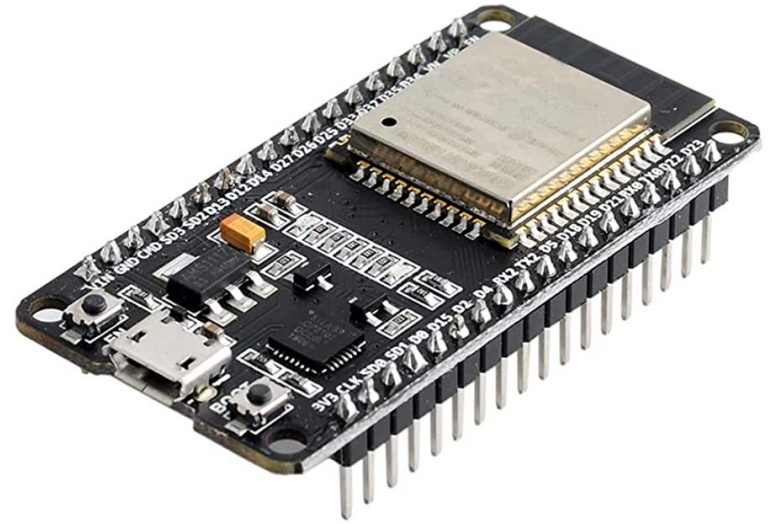
Jesús Ramírez Delgado - A01274723

Arturo Sánchez Rodríguez - A01275427

Eliuth Balderas Neri - A01703315

Introducción

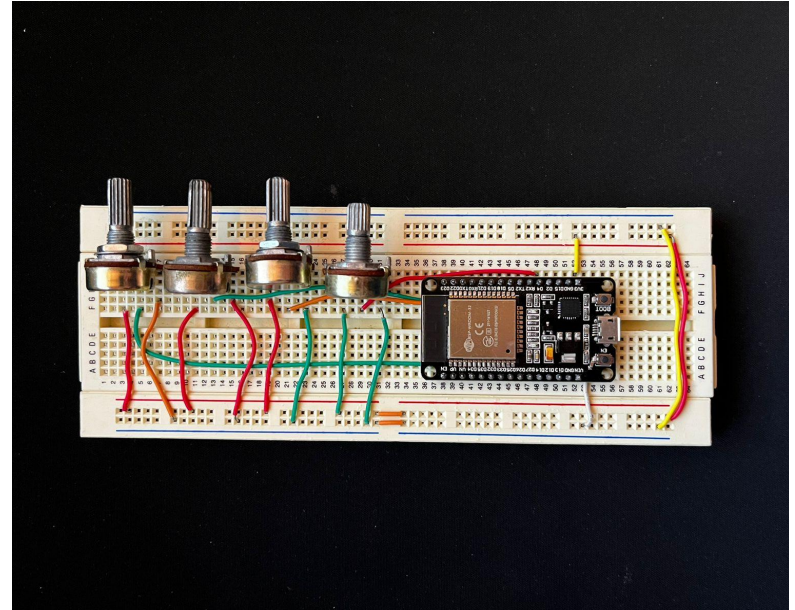
En este bloque, diseñaremos e implementaremos un sistema domótico interconectado utilizando microcontroladores. Recopilaremos datos de sensores y controlaremos actuadores en tiempo real. Afrontaremos desafíos y diseñaremos soluciones para mejorar el sistema. En resumen, aplicaremos conocimientos en un proyecto de domótica interconectado.



Prototipo

Para el prototipo utilizamos una variedad de materiales como,

- Protoboard
- Esp32
- 4 potenciómetros
- Alambre para protoboard



Código Arduino

```
ESP32Rev1 | Arduino 1.8.19
File Edit Sketch Tools Help

ESP32Rev1
#include <AzureIoTHub.h>
#include <Esp32MqttClient.h>

#include <WiFi.h>
#include <PubSubClient.h>
#include <Firebase_Rtp_client.h>
#include <addons/TokenHelper.h>

// SSID y password de red abierta
const char* ssid = "INFINITUM189";
const char* password = "Nn4$8sEcd2";

// api key del la base de datos en tiempo real
#define API_KEY "AIzaSyBM392Pip8Lmfpw5eqi4ha3vc5aWNOA"

// URL del la base de datos en tiempo real
#define DATABASE_URL "https://tcl004b-46378-default-rtdb.firebaseio.com"

// Para definir el proyecto fdbd de firebase y darle authorization
FirebaseData fdbd;

FirebaseAuth auth;
FirebaseConfig config;

unsigned long sendDataFreeMillis = 0;

int potentiometerValues[4] = {0}; // Este arreglo sirve para guardar el valor de cada potenciómetro.

void setup() {
  Serial.begin(115200);
  delay(10);
  setup_wifi();

  // Se definen los pines en los que se encuentran conectados los potenciómetros al ESP32
  pinMode(A4, INPUT);
  pinMode(A5, INPUT);
  pinMode(A6, INPUT);
  pinMode(A7, INPUT);
}

// Se hace la configuración y setup de la red
void setup_wifi() {
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
```

```
ESP32Rev1 | Arduino 1.8.19
File Edit Sketch Tools Help

ESP32Rev1
config.api_key = API_KEY;
config.database_url = DATABASE_URL;

// Se hace el signup en firebase
if (Firebase.auth().signIn(config, ssid, "", "") {
  Serial.println("Signup successful");
  Firebase.reconnectWiFi(true); // Moviendo aquí
} else {
  Serial.printf("Signup failed. Reason: %s\n", config.signer.signInErrorMessage_c_str());
}

// Assign the callback function for the long running token generation task
config.token_status_callback = tokenStatusCallback; // see addons/TokenHelper.h

Firebase.begin(config, auth);
}

void loop() {
  if (Firebase.ready() && millis() - sendDataFreeMillis > 15000 || sendDataFreeMillis == 0) {
    sendDataFreeMillis = millis();

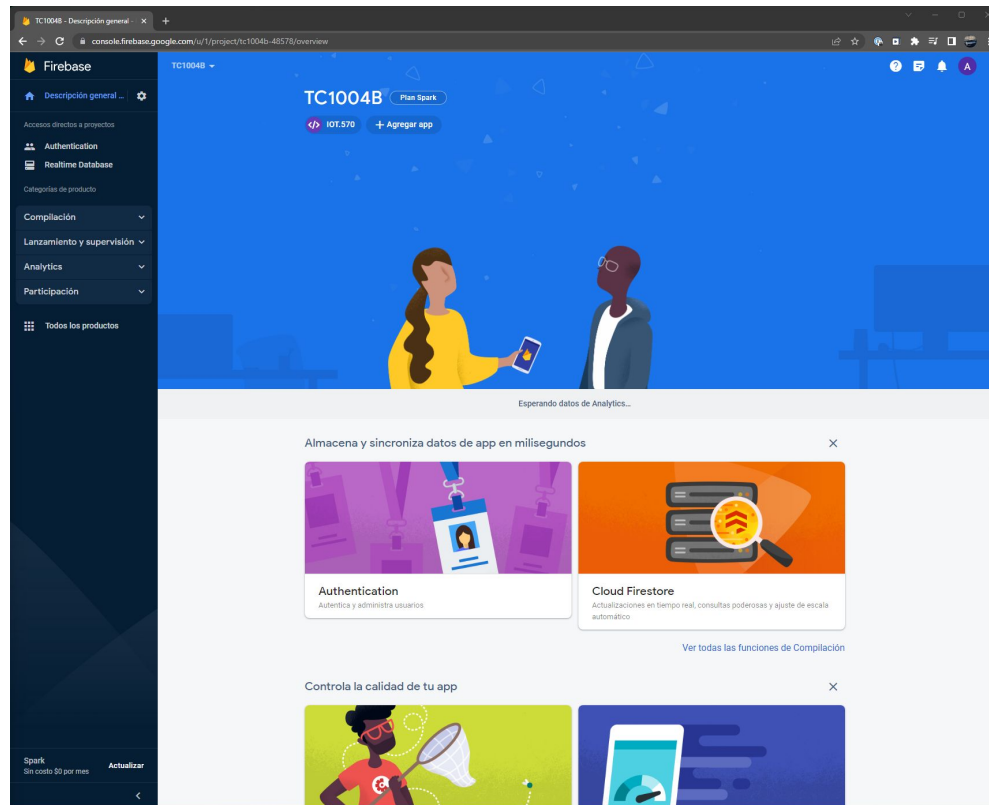
    // Leer valores analógicos de los potenciómetros
    int pot1Value = analogRead(A4);
    int pot2Value = analogRead(A5);
    int pot3Value = analogRead(A6);
    int pot4Value = analogRead(A7);

    potentiometerValues[0] = pot1Value;
    potentiometerValues[1] = pot2Value;
    potentiometerValues[2] = pot3Value;
    potentiometerValues[3] = pot4Value;

    // Imprimir los valores en el monitor serial
    Serial.print("Potenciómetro 1: ");
    Serial.println(pot1Value);
    Serial.print("Potenciómetro 2: ");
    Serial.println(pot2Value);
    Serial.print("Potenciómetro 3: ");
    Serial.println(pot3Value);
    Serial.print("Potenciómetro 4: ");
    Serial.println(pot4Value);

    // Autenticación con Firebase
    if (Firebase.auth().signIn() {
      // En este bucle se van subiendo los datos de los potenciómetros a la base de datos en tiempo real de firebase
      for (int i = 0; i < 4; i++) {
        String path = "potenciómetros/pot" + String(i + 1);
        if (Firebase.RTDB.set(int(fdbd, path_c_str(), potentiometerValues[i])) {
          Serial.println("Potenciómetro " + String(i + 1) + " value sent successfully");
          Serial.println("PATH: " + fdbd.dataPath());
          Serial.println("TYPE: " + fdbd.dataType());
        } else {
          Serial.println("Failed to send potentiometer " + String(i + 1) + " value");
          Serial.println("REASON: " + fdbd.errorReason());
        }
      }
    }
  }
}
```

FireBase



TC1004B - Realtime Database -

console.firebase.google.com/u/1/project/tc1004b-48578/database/tc1004b-48578-default-rtdb/data

Realtime Database

Datos Reglas Copias de seguridad Uso Extensiones **NUOVA**

Protege tus recursos de Realtime Database contra los abusos, como fraudes de facturación o suplantación de identidad. [Configurar la Verificación de aplicaciones](#)

<https://tc1004b-48578-default-rtdb.firebaseio.com>

```
https://tc1004b-48578-default-rtdb.firebaseio.com/  
└─ potimeters  
   └─ pot1: 752  
      └─ pot2: 401  
         └─ pot3: 0  
            └─ pot4: 0
```

TC1004B - Authentication - Fireb

console.firebase.google.com/u/1/project/tc1004b-48578/authentication/users

Authentication

Users Sign-in method Templates Usage Settings Extensiones **NUOVA**

Buscar por dirección de correo electrónico, número de teléfono o UID de usuario **Agregar usuario**

Identificador	Proveedores	Fecha de creación	Fecha de acceso	UID de usuario
(anónimo)		14 jun 2023	14 jun 2023	019gqrWrmefsiW1zgayoAXPE63
(anónimo)		14 jun 2023	14 jun 2023	0GmksGVb5fcX9aTkdm2ApjCWp...
(anónimo)		14 jun 2023	14 jun 2023	uRTTMEObG8XAagYGI5P6cfVbeH...
(anónimo)		13 jun 2023	13 jun 2023	kuclWy060g068QCCXumW3bxe...
(anónimo)		13 jun 2023	13 jun 2023	okRMqn5VivQJ09V8atUIB2hi5o582
(anónimo)		13 jun 2023	13 jun 2023	k9SjAYz2l6VVoc0e3b5mNKQxPJu2

Filas por página: 50 1 - 6 of 6 < >

Página Web

Para evitar problemas de compatibilidad desarrollamos una sencilla aplicación web con HTML, CSS y Java Script

```
<> botones.html  
JS fade.js  
<> index.html  
🖼️ LOGO TEC.png  
JS script.js  
# style.css  
# styleBotones.css
```

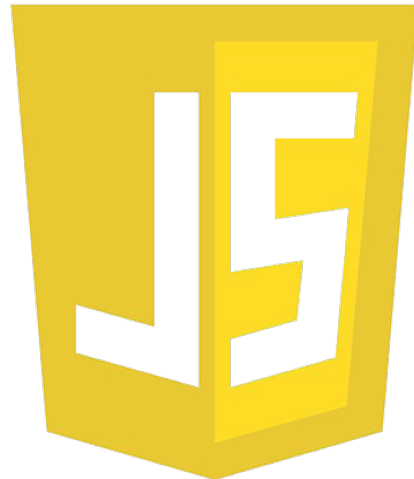
HTML



CSS



JS



Página principal **index.html** de presentación que conecta a **botones.html**



Descripción Proyecto

Mediante una placa de Arduino ESP32 se conectaron varios módulos para poder realizar diferentes medidas.

Se conectaron 4 potenciómetros (por falta de material).

Una vez en funcionamiento, se conecto a una base de datos, en este caso se implementó Firebase, y se conecto a el IDE de Arduino. Finalmente se programo esta App Web sencilla, que obtiene los datos de Firebase y tu puedes consultar en tiempo real los datos de cada módulo conectado a la placa ESP32.

PROBAR

Gracias

Página **botones.html** que tiene la capacidad de regresar a la página principal. En esta página presionando el botón “**mostrar datos**” mediante una conexión con firebase, muestra los últimos datos de la base de datos en tiempo real de cada potenciómetro.

REGRESAR

Medidas

MOSTRAR DATOS

```
{ "potentiometers": { "pot1": 752, "pot2": 401, "pot3": 0, "pot4": 0 } }
```

Lógica

Para la lógica de la pagina utilizamos Java Script, el código implementado fue el siguiente:

```
1 window.addEventListener('load', (event) => {
2   // Your web app's Firebase configuration
3   var firebaseConfig = {
4     apiKey: "AIzaSyBdNJ82PipRL0HfpVw5sqidBaZv65xWNOA",
5     authDomain: "tc1004b-48578.firebaseio.com",
6     databaseURL: "https://tc1004b-48578-default-rtdb.firebaseio.com",
7     projectId: "tc1004b-48578",
8     storageBucket: "tc1004b-48578.appspot.com",
9     messagingSenderId: "510427288538",
10    appId: "1:510427288538:web:f16e35a454faaef0c966e3",
11    measurementId: "G-ZJGXK9HFJV"
12  };
13
14  // Initialize Firebase
15  firebase.initializeApp(firebaseConfig);
16
17  var database = firebase.database();
18
19  document.getElementById('myButton').addEventListener('click', function() {
20    var ref = database.ref('/');
21
22    ref.on('value', function(snapshot) {
23      document.getElementById('myData').textContent = JSON.stringify(snapshot.val(), null, 2);
24    });
25  });
26 });
```

Conclusión

Utilizar Firebase para analizar datos recolectados por sensores como en este caso el uso de potenciómetros, ofrece varias ventajas. En primer lugar, nos permite un almacenamiento de datos en tiempo real, lo que facilita el análisis de datos y toma de decisiones basada en datos actualizados. Por otro lado, aun en este caso únicamente se utilizaron 5 variables, Firebase es una plataforma escalables, puesto a que permite manejar grandes volúmenes de datos, lo que también nos otorga un buen rendimiento. Por otro lado, Firebase es una plataforma bastante fácil de usar, ya que es intuitiva, lo que permite gestionar y analizar los datos de una manera sencilla. Adicionalmente, como Firebase guarda los datos en la nube, permite que se tenga un acceso fácil en cualquier momento desde cualquier lugar. Finalmente, como se vio en el reto, Firebase se puede integrar con otras herramientas, lo que hace que tenga muchas más posibilidades para analizar los datos recolectados.

