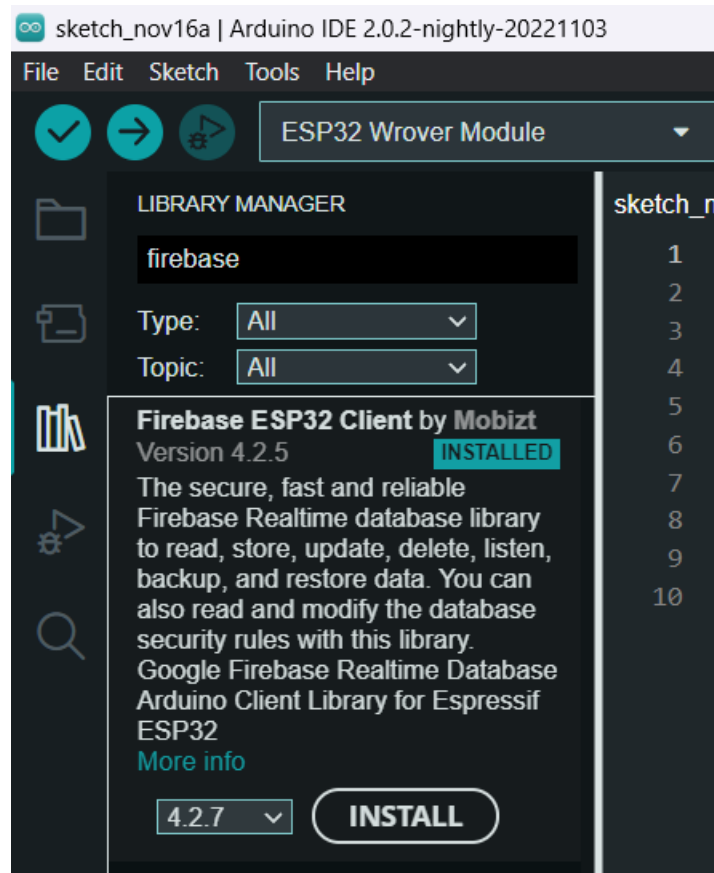


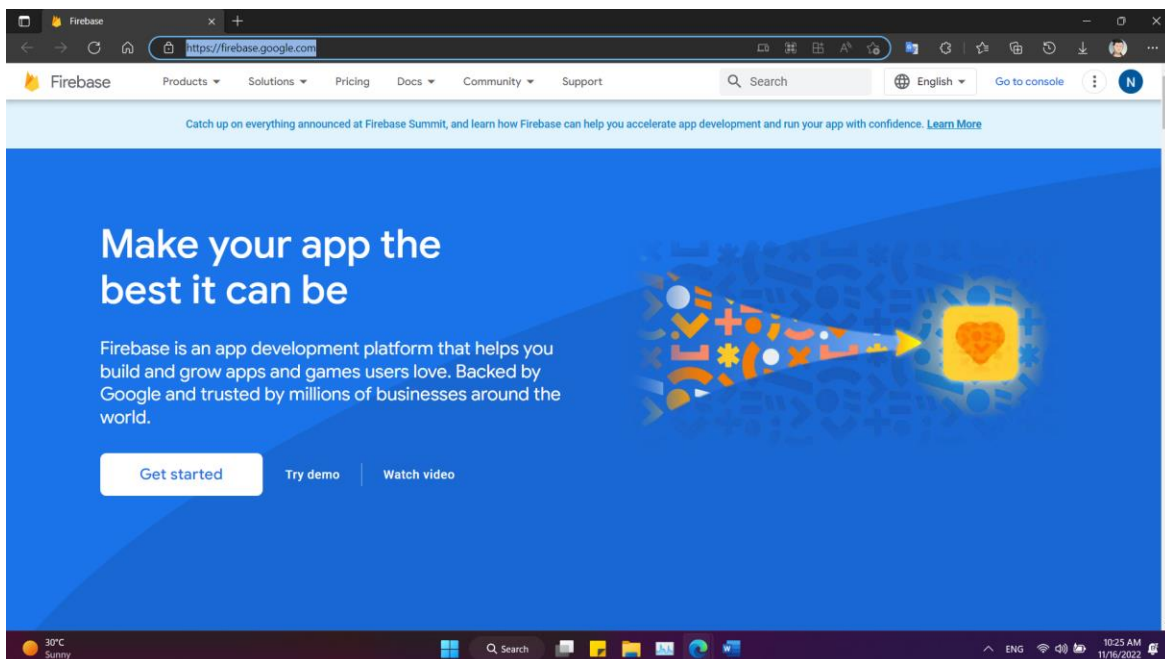
1. Các bước thực hiện, giải thích code quá trình cập nhật dữ liệu lên Google Firebase và hình ảnh kết quả thực hiện (video clip demo nếu có).

## Bước 1: Cài thư viện Firebase cho ESP32 trên Arduino IDE

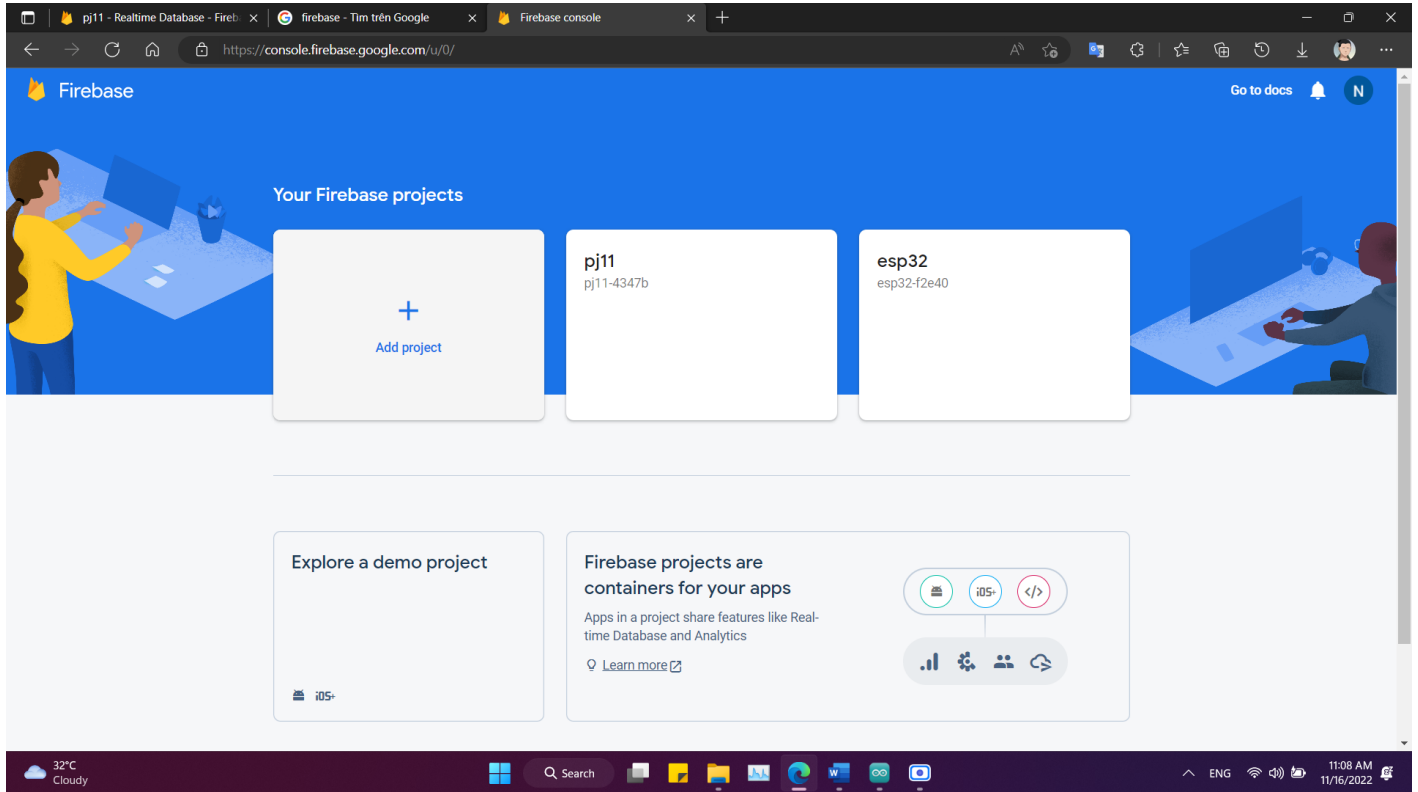


## Bước 2: Thiết lập trên Firebase

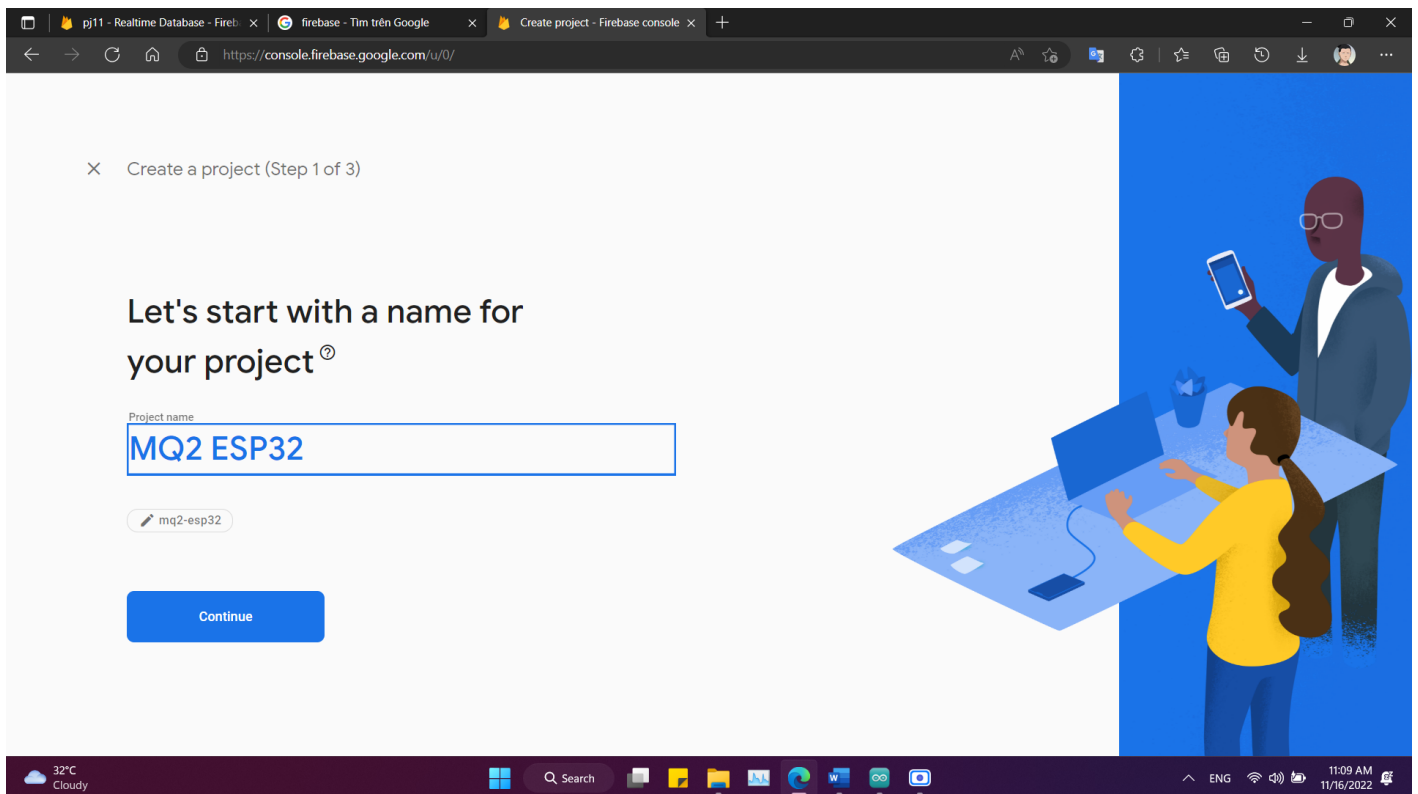
Vào [Firebase \(google.com\)](https://firebase.google.com). Chọn **Get Started**.



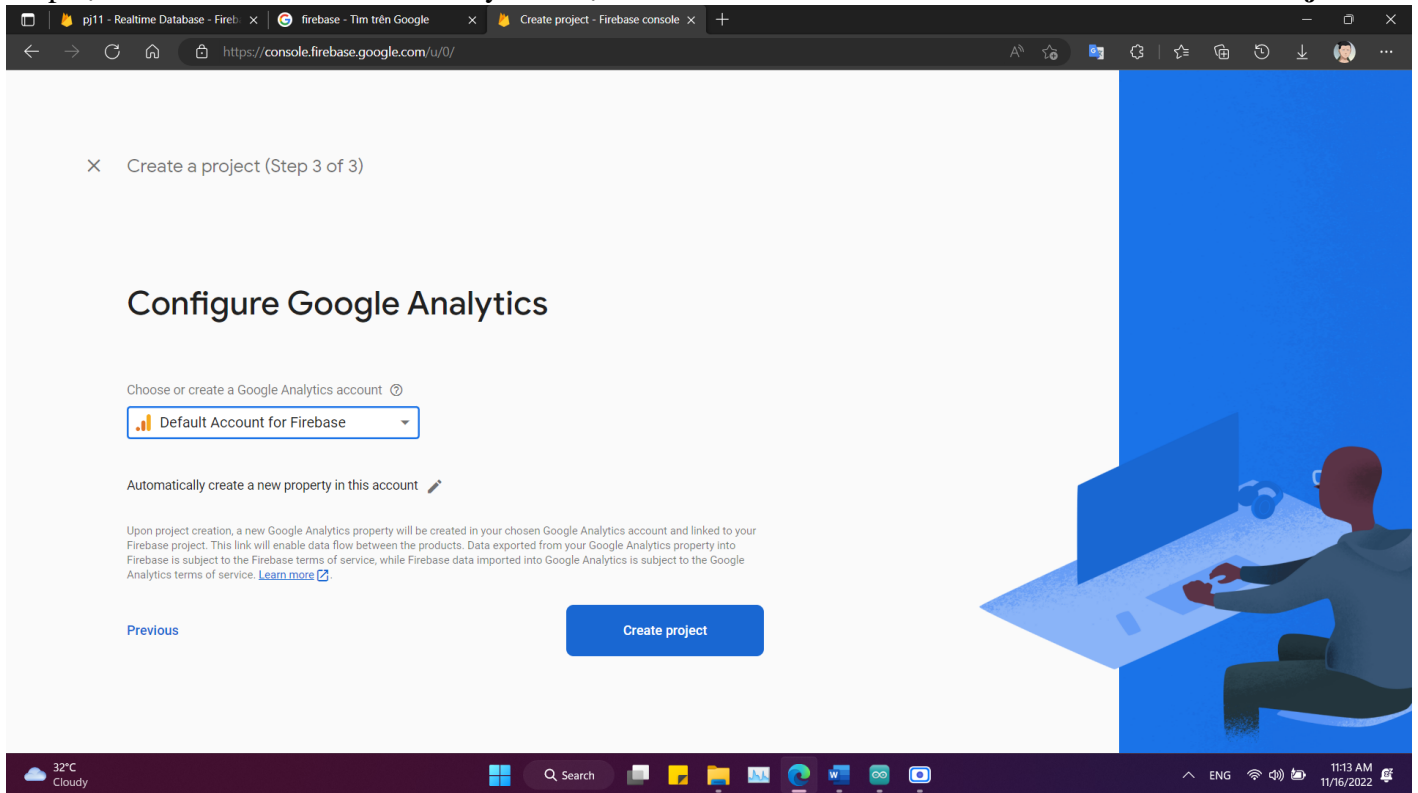
Chọn **Add project** để tạo 1 project mới



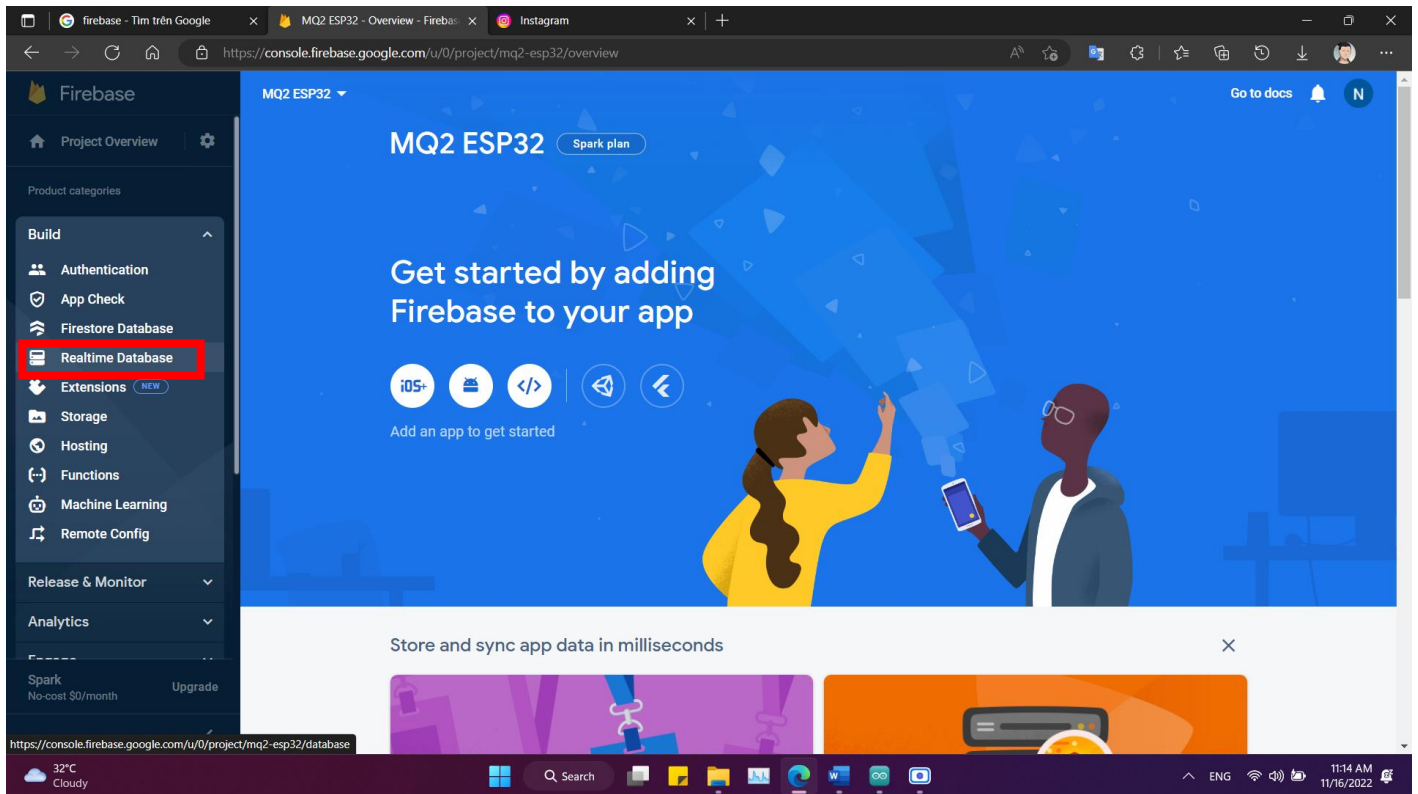
Đặt tên cho Project. Lưu ý: Tên project phải trên 4 ký tự và không được chứa ký tự đặc biệt. Và nhấn **Continue**.



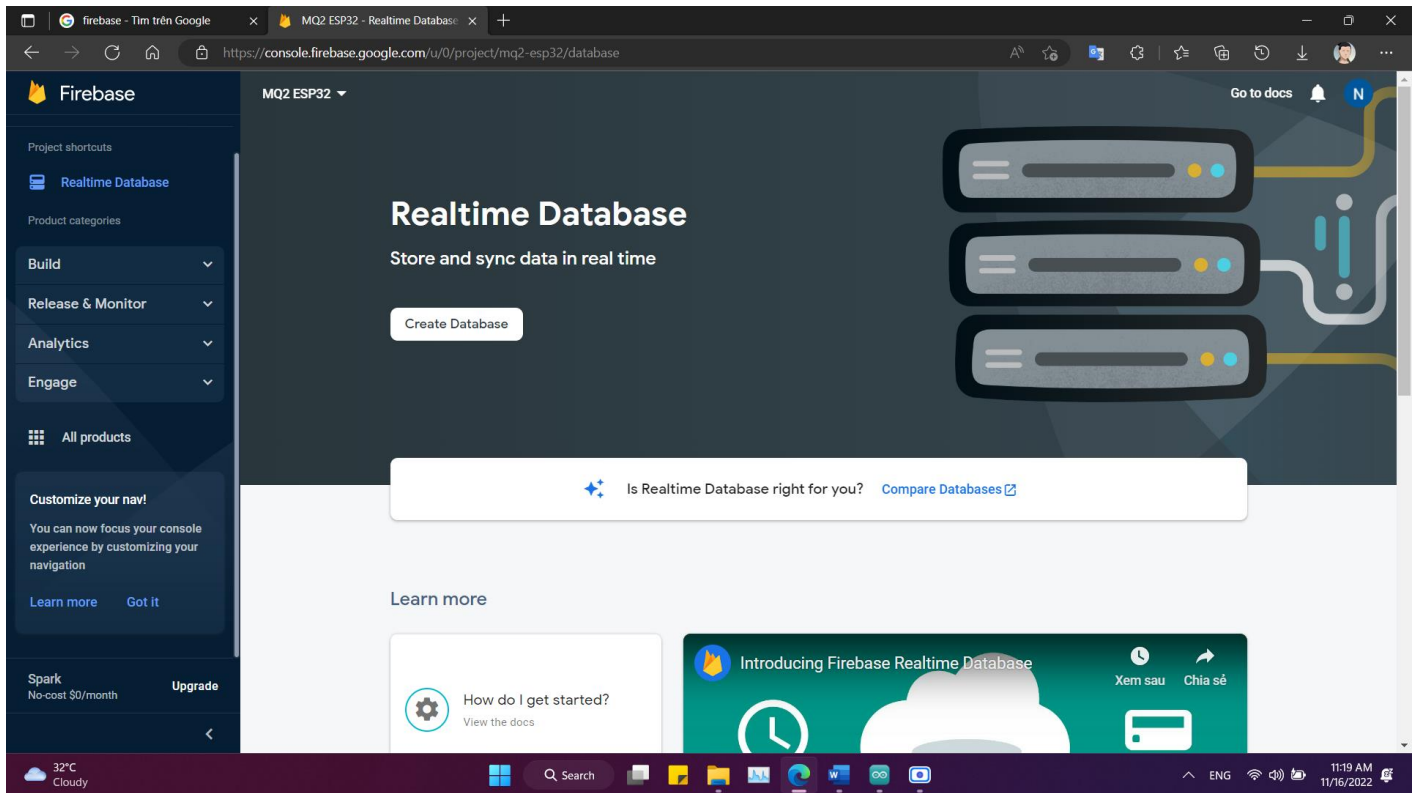
Tiếp tục nhấn Continue đến bước này thì chọn **Default Account for Firebase** và click **Create Project** để tạo



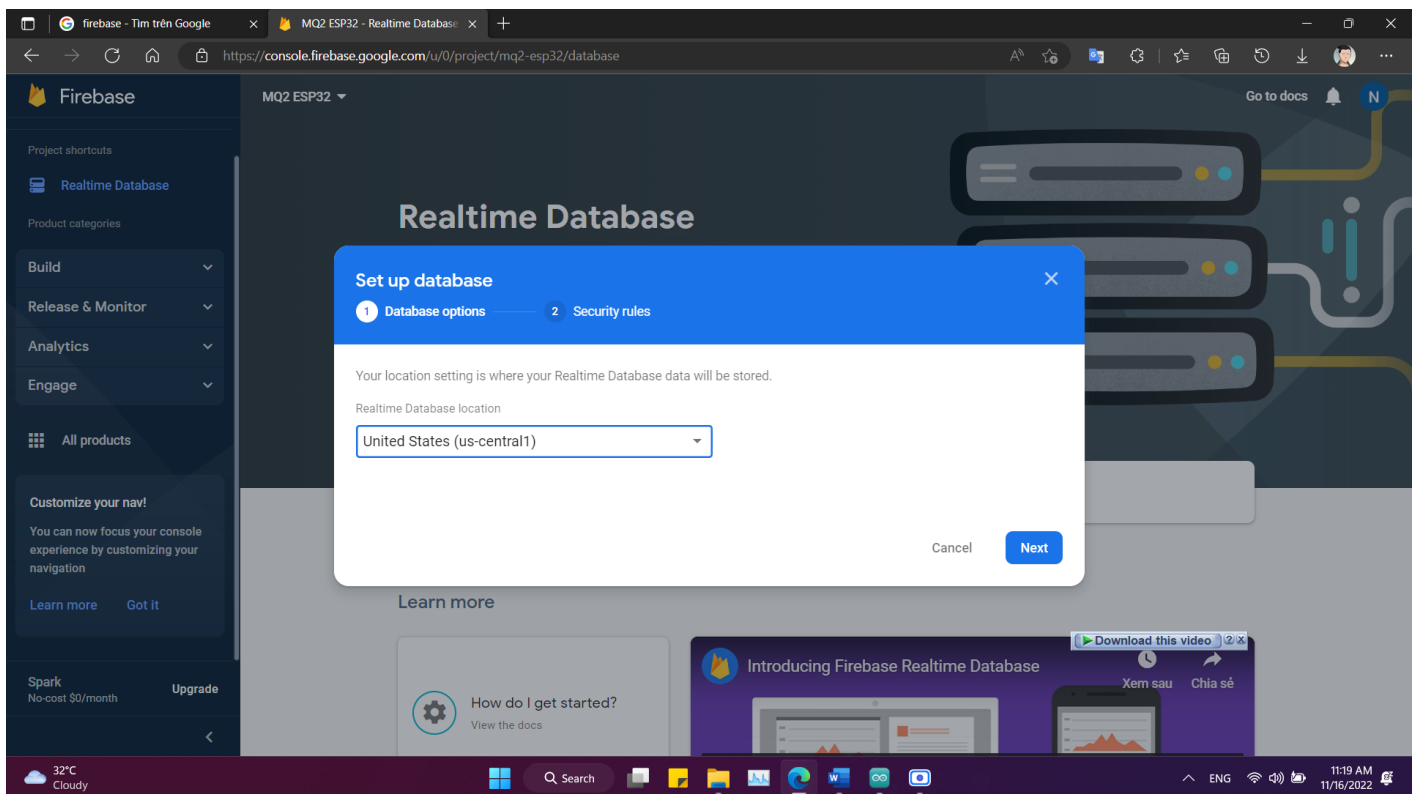
Sau khi tạo được project. Chọn **Realtime Database**



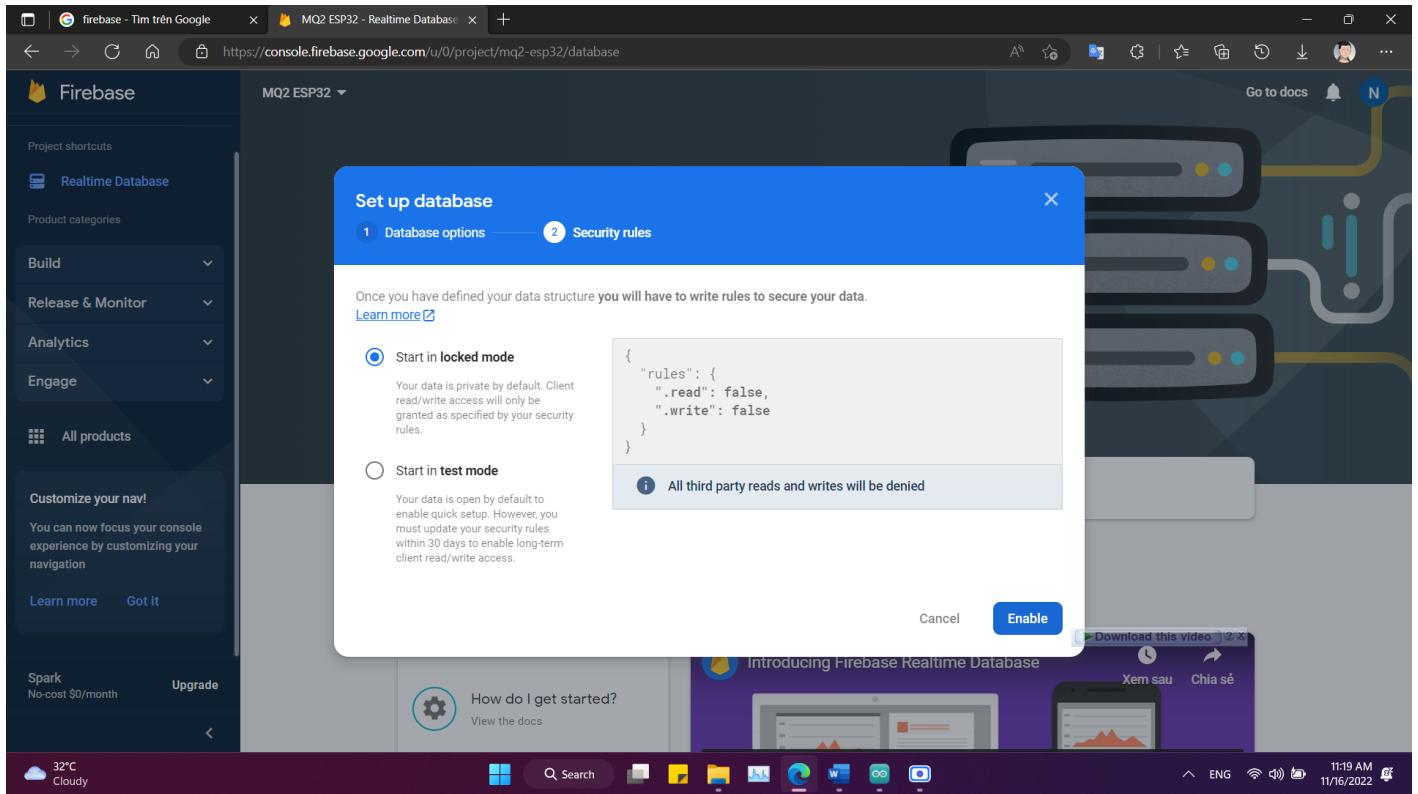
## Chọn Create Database



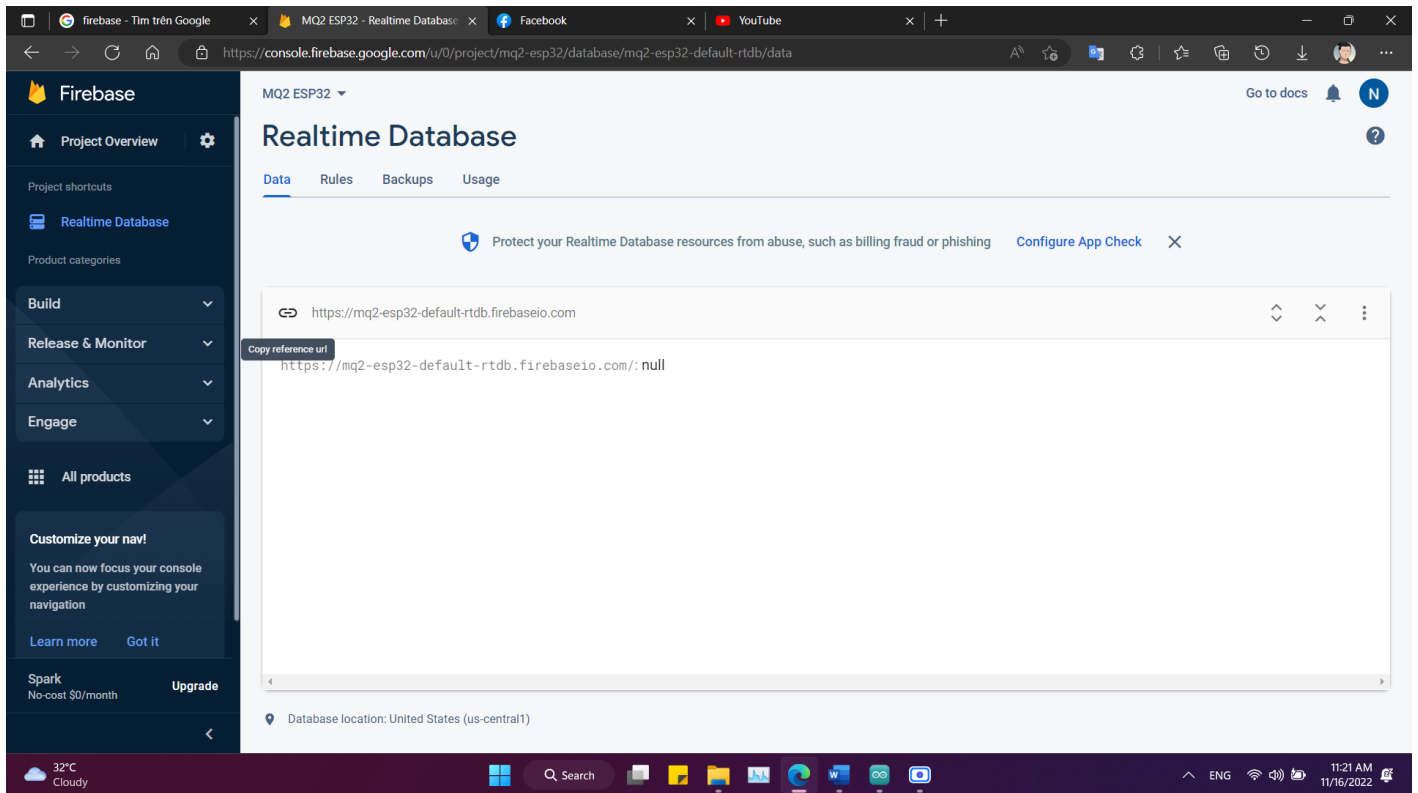
## Nhấn Next



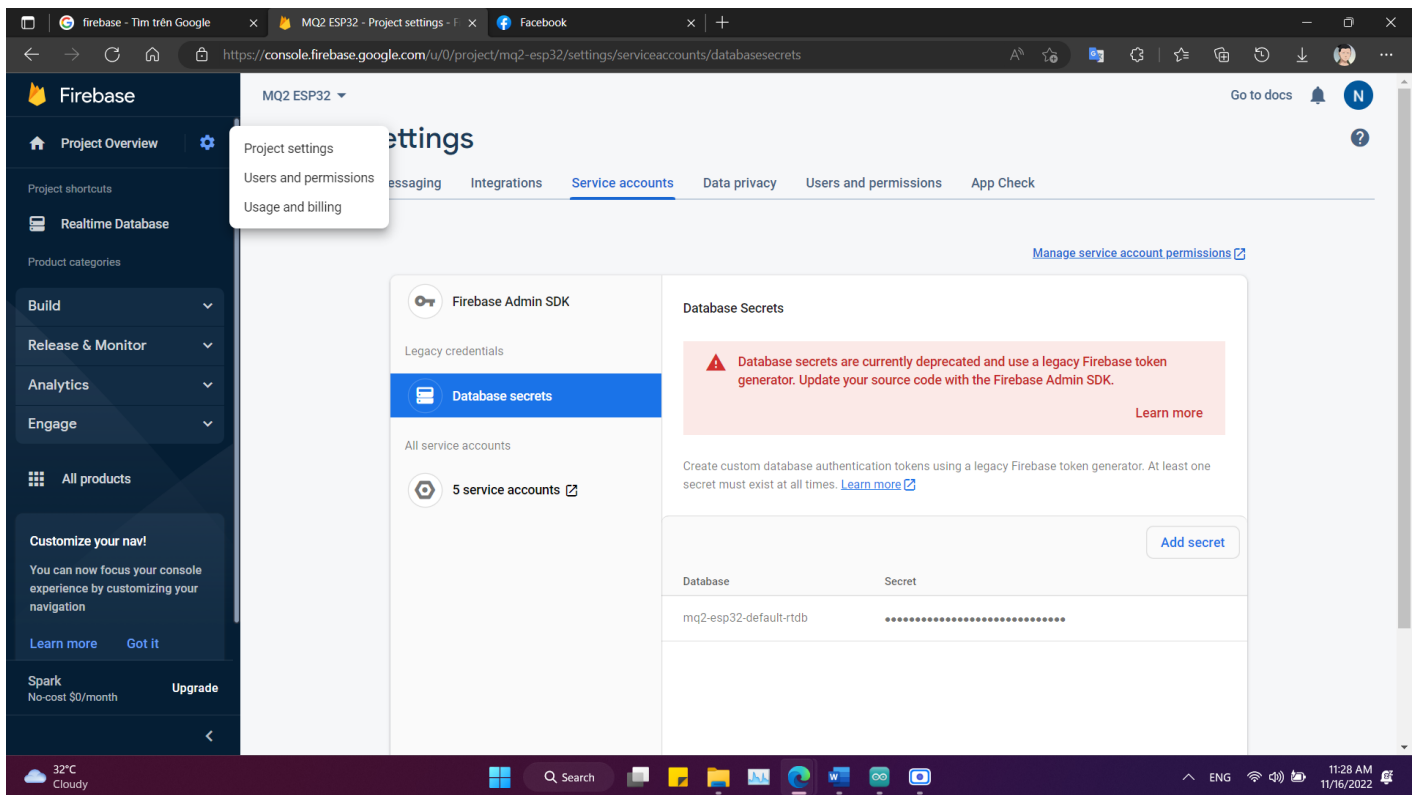
## Chọn Enable



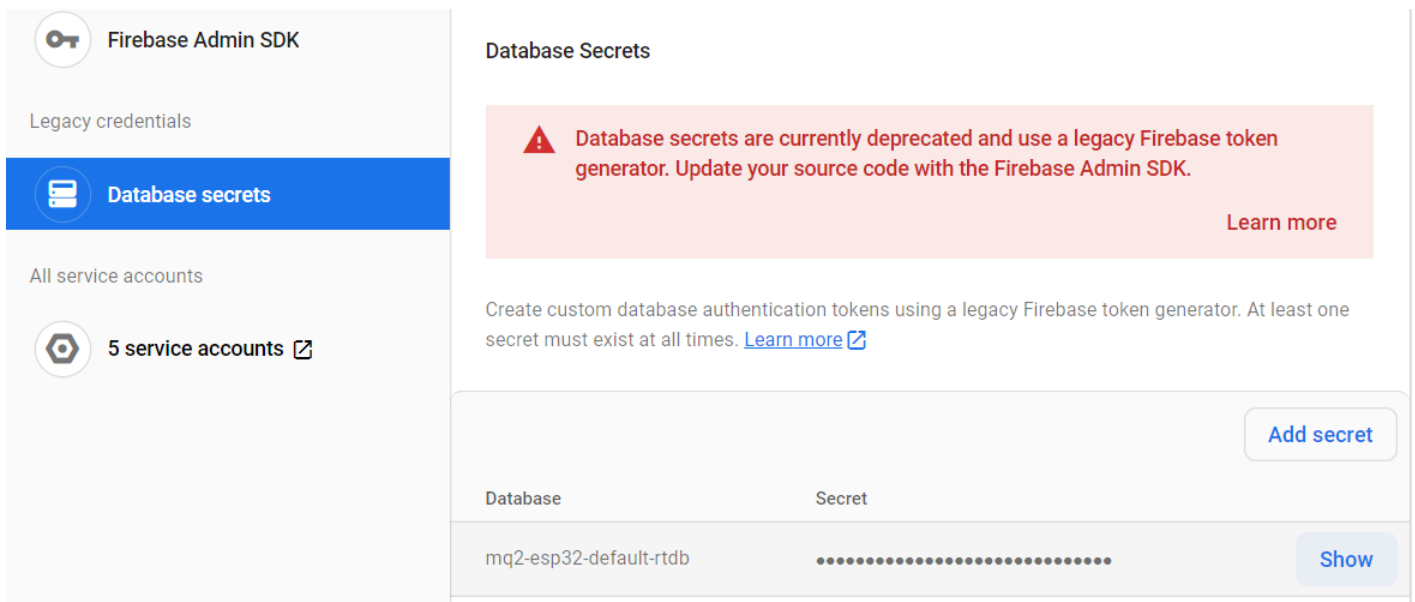
## Copy phân URL để dán vào code



## Chọn Project Overview > Project Settings > Service accounts



## Chọn Show để hiện mã xác thực và copy để dán vào code



## Sau khi dán vào code

```
4 #define FIREBASE_HOST "https://mq2-esp32-default-rtdb.firebaseio.com/"
5 #define FIREBASE_AUTH "U70RPM1YULidokyXd9VyTFI2erfeUtXdg0XgCEca"
6
```

### Bước 3: Thực hiện viết code và nạp code

```
#include <FirebaseESP32.h> //Khai báo các thư viện Firebase cho ESP32
#include <WiFi.h> //Khai báo các thư viện Wifi

#define FIREBASE_HOST "https://mq2-esp32-default-rtdb.firebaseio.com/" //URL Realtime database
#define FIREBASE_AUTH "U7ORPM1YULidokyXd9VyTFI2erfeUtXdg0XgCEca" //Mã xác thực

#define WIFI_SSID "Dinh Thanh"
#define WIFI_PASSWORD "thanhdat16"

FirebaseData fbdo; //PUT dữ liệu vào Firebase

#define PinDigital 15 //gắn chân Digital của MQ2 vào chân 15 của ESP32
#define led 5 //gắn Led
int digital;

void setup() { //Chương trình cài đặt
    Serial.begin(9600);
    pinMode(PinDigital, INPUT);
    pinMode(led, OUTPUT);
    WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
    Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
}

void loop() {

    digital = digitalRead(PinDigital); //gán lệnh analog là giá trị đọc được từ Pin Digital

    Serial.print("Gas Sensor: "); // in ra màn hình giá trị đọc được
    Serial.println(digital);
    Firebase.setFloat(fbdo, "Gas Sensor: ", digital); //hiển thị trên hệ thống Firebase
    delay(100);

    if (digital == 0) //nếu có khí gas digital = 0 thì đèn led sẽ sáng
    {
        digitalWrite(led, HIGH);
    }
    else
    {
        digitalWrite(led, LOW); // nếu không có thì đèn led sẽ tắt
    }
}
```



Khi cảm biến không phát hiện ra khí GAS, dữ liệu từ Firebase sẽ hiện là 1.

The screenshot shows a dual-monitor setup. The left monitor displays the Firebase Realtime Database console for the project 'MQ2 ESP32'. The 'Data' tab is active, showing a single data point: 'Gas Sensor: 1'. The right monitor displays the Arduino IDE with the sketch 'ESP32\_dan\_MQ2\_Firebase.ino'. The code is for an ESP32 Wrover Module and includes logic to read the MQ2 sensor and control an LED. The serial monitor at the bottom of the IDE shows the output: 'Gas Sensor: 0'.

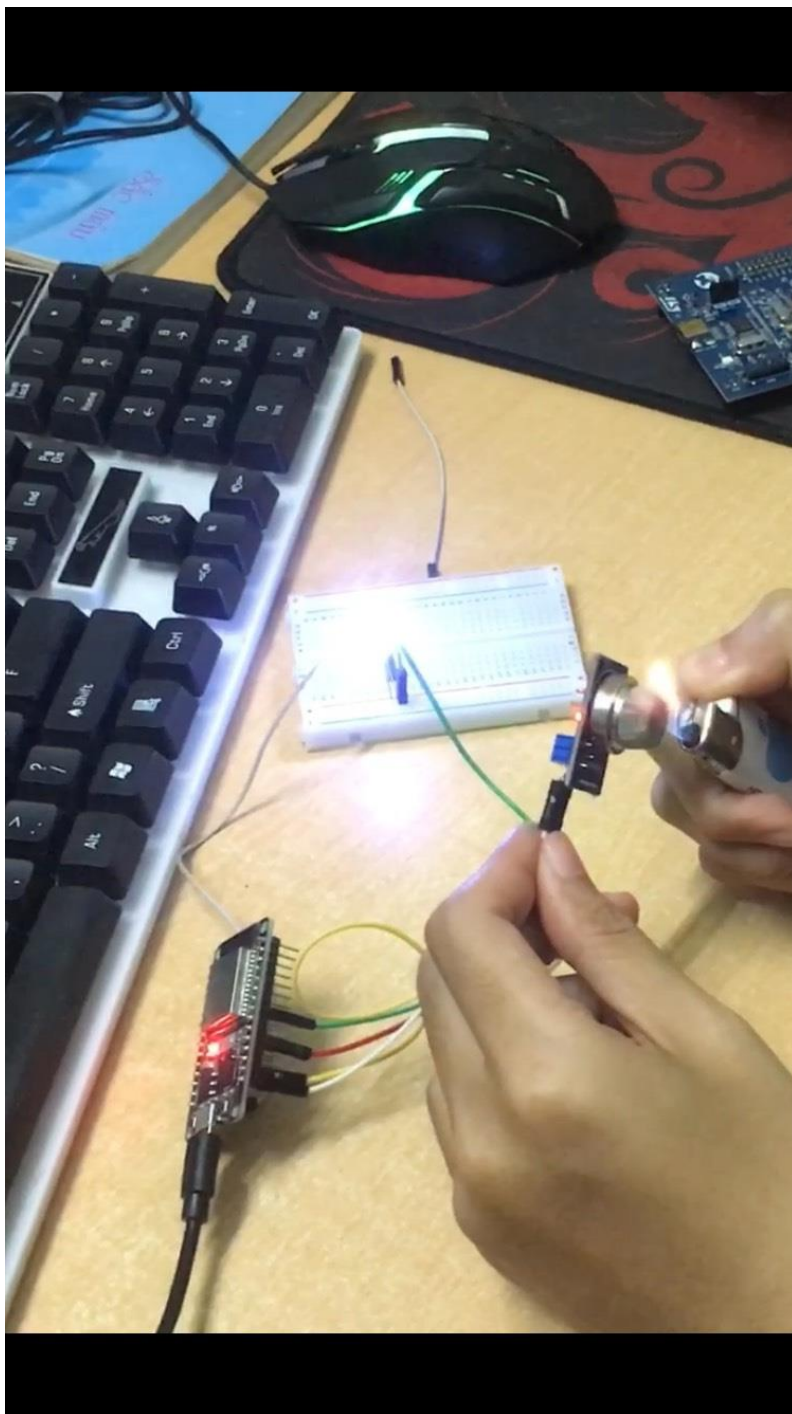
Khi có khí GAS sẽ hiện về 0.

The screenshot shows a dual-monitor setup. The left monitor displays the Firebase Realtime Database console for the project 'MQ2 ESP32'. The 'Data' tab is active, showing a single data point: 'Gas Sensor: 0'. The right monitor displays the Arduino IDE with the sketch 'ESP32\_dan\_MQ2\_Firebase.ino'. The code is for an ESP32 Wrover Module and includes logic to read the MQ2 sensor and control an LED. The serial monitor at the bottom of the IDE shows the output: 'Gas Sensor: 1'.



Hình ảnh kết quả thực hiện

Khi có khí GAS thì đèn sẽ sáng



Khi không có khí GAS thì đèn sẽ tắt

