**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**KHOA CNTT 1**

-----🙞🙜🕮🙞🙜-----



**BÀI TẬP LỚN**

**XÂY DỰNG HỆ THỐNG NHÚNG**

ĐỀ TÀI: HỆ THỐNG CẢNH BÁO KHÍ GAS

VÀ GIÁM SÁT NHIỆT ĐỘ

|  |  |
| --- | --- |
| **Giảng Viên:** | **Vũ Hoài Nam** |
|  |  |
| **Nhóm BTL:** |  |
|  |
| 1. ***Đặng Trường Giang*** | ***B13DCCN194*** |
| 1. ***Đặng Đình Hùng*** | ***B12DCCN437*** |
|  |
|  |

Hà Nội, tháng 05 năm 2017

MỤC LỤC

[1. Bảng phân chia công việc 2](#_Toc482387089)

[2. Giới thiệu chung 2](#_Toc482387090)

[2.1. Tổng quan về IoT (Internet of Things) 2](#_Toc482387091)

[2.2. Đặt vấn đề 2](#_Toc482387092)

[3. Các thành phần của hệ thốn 3](#_Toc482387093)

[3.1. Firebase 3](#_Toc482387094)

[3.2. Chip ESP8266 3](#_Toc482387095)

[3.3. Cảm biến DHT11 4](#_Toc482387096)

[3.4. Cảm biến MQ2 5](#_Toc482387097)

[3.5. Các thành phần khác 6](#_Toc482387098)

[4. Hoạt động của hệ thống 6](#_Toc482387099)

[4.1. Sơ đồ tổng quát 6](#_Toc482387100)

[4.2. Mô tả hoạt động 6](#_Toc482387101)

[5. Thiết kế phần cứng 7](#_Toc482387102)

[5.1. Mạch giám sát nhiệt độ với DHT11 7](#_Toc482387103)

[5.2. Mạch giám sát nồng độ khí gas với MQ2 8](#_Toc482387104)

[5.3. Mạch lắp đặt thực tế 8](#_Toc482387105)

[6. Thiết kế phần mềm 10](#_Toc482387106)

[6.1. Cài đặt Firebase. 10](#_Toc482387107)

[6.2. Lập trình Arduino 14](#_Toc482387108)

[6.3. Lập trình Android 14](#_Toc482387109)

[7. Tổng kết 18](#_Toc482387110)

# Bảng phân chia công việc

|  |  |  |
| --- | --- | --- |
|  | Đặng Trường Giang (nhóm trưởng) | Đặng Đình Hùng |
| Phân tích thiết kế hệ thống | **√** |  |
| Lắp mạch | **√** |  |
| Cài đặt Server | **√** |  |
| Code Arduino |  | **√** |
| Code Android | **√** |  |
| Tích hợp | **√** |  |
| Kiểm thử |  | **√** |
| Viết báo cáo |  | **√** |

# Giới thiệu chung

## Tổng quan về IoT (Internet of Things)

**Internet of Things (IoT)** là một kịch bản của thế giới, khi mà mỗi đồ vật được cung cấp một định danh của riêng mình, và tất cả có khả năng truyền tải, trao đổi thông tin, dữ liệu qua một mạng duy nhất mà không cần đến sự tương tác trực tiếp giữa người với người, hay người với máy tính. IoT đã phát triển từ sự hội tụ của công nghệ không dây, công nghệ vi cơ điện tử và Internet.

Như vậy, Internet of Things là khi tất cả mọi thứ đều được kết nối với nhau qua mạng Internet, người dùng (chủ) có thể kiểm soát mọi đồ vật của mình qua mạng mà chỉ bằng một thiết bị thông minh, chẳng hạn như smartphone, tablet, PC hay thậm chí chỉ bằng một chiếc smartwatch nhỏ bé trên tay.

Rõ ràng Internet of Things có thể thay đổi hoàn toàn cách sống của loài người. Đây chính là một xu thế công nghệ, tương lai của thế giới.

## Đặt vấn đề

Ngày nay, với sự phát triển của khoa học kỹ thuật, đời sống của con người ngày càng được nâng cao, các thiết bị sử dụng hàng ngày cũng ngày càng trở nên hiện đại. Tuy nhiên đi cùng với đó là việc ẩn chứa nhiều nguy cơ về cháy nổ. Phần lớn các gia đình hiện nay sử dụng bếp gas để phục vụ cho nhu cầu nấu nướng, do đó có khả năng xảy ra tình trạng rò rỉ khí gas, nếu không gây ra cháy cũng gây ảnh hưởng xấu đến sức khỏe, thậm chí là nguy hiểm tới tính mạng. Việc có một hệ thống giám sát và cảnh báo là cần thiết để tránh những hậu quả đáng tiếc có thể xảy ra.

Từ những kiến thức đã có và đã học được qua môn Xây dựng các hệ thống nhúng, nhóm đề xuất ý tưởng xây dựng hệ thống phát hiện rò rỉ khí gas và giám sát nhiệt độ, có thể áp dụng trong gia đình. Hệ thống sẽ ghi lại liên tục thông số về nồng độ chất dễ gây cháy trong không khí (thường là khí gas), cùng với đó là nhiệt độ hiện tại và hiển thị trên máy tính và thiết bị Android của người dùng nếu được kết nối wifi. Như vậy kể cả khi không ở nhà, chủ nhà cũng có thể biết được tình trạng tại nhà ra sao. Nếu nồng độ khí gas và nhiệt độ vượt qua một ngưỡng cho phép, sẽ có cảnh báo (đèn và chuông) được bật để những người ở nhà cũng như hàng xóm xung quanh có thể lưu ý.

# Các thành phần của hệ thốn

## Kết quả hình ảnh cho firebase là gìFirebase

**Firebase** là một nền tảng di động giúp bạn phát triển các ứng dụng chất lượng cao, phát triển ứng dụng quy mô lớn. Firebase cung cấp một số lượng lớn các dịch vụ hỗ trợ lập trình viên xây dựng ứng dụng của mình. Android là một trong những nền tảng được hỗ trợ mạnh nhất.

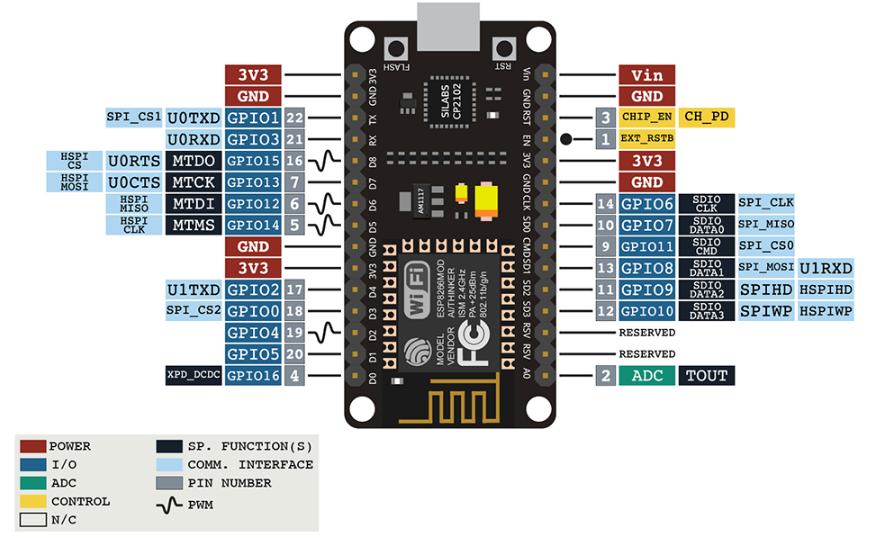
Nhóm đã sử dụng Firebase đóng vai trò như 1 máy chủ để chuyển và nhận dữ liệu giữa thiết bị vi điều khiển và ứng dụng trên Android.(ESP8266 và Android).

Dữ liệu trong cơ sở dữ liệu Firebase được lưu trữ dưới dạng JSON và đồng bộ realtime đến mọi kết nối client. Khi xây dựng những ứng dụng đa nền tảng như Android, iOS và JavaScrip SDKs, tất cả các client sẽ chia sẻ trên một cơ sở dữ liệu Firebase và tự động cập nhật với dữ liệu mới nhất.

Nguyên lý hoạt động trong hệ thống của nhóm khá đơn giản: ESP8266 đọc và xử lý dữ liệu từ cảm biến và gửi lên Firebase lưu trữ và cập nhật sau mỗi 1 giây, dữ liệu này sẽ được gửi đến ứng dụng trên thiết bị Android để hiển thị và ngược lại.

## Chip ESP8266

**ESP8266** là một dòng chip tích hợp Wi-Fi 2.4Ghz có thể lập trình được với đầy đủ các tính năng về Internet, kích thước nhỏ gọn, mức giá khiêm tốn, sử dụng cho giao tiếp qua sóng wifi tiện lợi, đơn giản, gọn nhẹ. Đây là sự lựa chọn hợp lý cho những dự án IoT.



*Sơ đồ chân ESP8266*

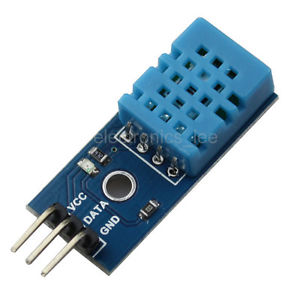
***Thông số kỹ thuật:***

* Hỗ trợ chuẩn 802.11 b/g/n.
* Wi-Fi 2.4 GHz, hỗ trợ WPA/WPA2.
* Chuẩn điện áp hoạt động: 3.3V.
* Chuẩn giao tiếp nối tiếp UART với tốc độ Baud lên đến115200.
* Có 3 chế độ hoạt động: Client, Access Point và kết hợp cả 2.
* Hỗ trợ các chuẩn bảo mật như: OPEN, WEP, WPA\_PSK, WPA2\_PSK, WPA\_WPA2\_PSK.
* Hỗ trợ cả 2 giao tiếp TCP và UDP.
* Khả năng hoạt động trên dải nhiệt rộng từ -40oC tới 125oC.

Trong mạch, ESP8266 đóng vai trò điều khiển các thiết bị cảm biến và cảnh báo. Cụ thể nó sẽ nhận dữ liệu được đọc từ cảm biến, xử lý và gửi lên server, đồng thời điều khiển bật tắt cảnh báo đèn LED và chuông theo mức độ và yêu cầu. Các lệnh được lập trình và nạp cho chip bằng phần mềm Arduino.

## Cảm biến DHT11

DHT11 là cảm biến đo nhiệt độ với kích thước nhỏ gọn nhưng có độ chính xác cao và rất nhạy với sự thay đổi nhiệt độ.

*Thông số kỹ thuật:*

* Điện áp đầu vào: 3.3V-5V
* Độ phân giải điện áp đầu ra 10mV/oC
* Độ chính xác cao ở khoảng 25oC là 0.5oC
* Trở kháng đầu ra thấp 0.1 cho 1mA tải
* Dải nhiệt độ đo được của DHT11 là từ 0 oC tới 50 oC với các mức điện áp ra khác nhau.

DHT11 cho dữ liệu đầu ra dạng analog (tín hiệu tương tự) tại chân thứ 2 (chân AO ở giữa). Điện áp này tỉ lệ tuyến tính với nhiệt độ theo thang Celsious. Công thức đổi từ điện áp đầu ra (đơn vị mV) sang nhiệt độ (oC) như sau:

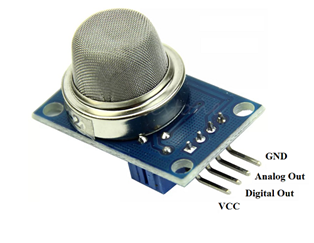
**Nhiệt độ = ( [Điện áp đầu ra] \* 3.3 / 1024.0 ) \*100.0**

Trong đó: điện áp đầu ra là giá trị đọc được từ chân AO ở giữa; 3.3 là điện áp đầu vào - giá trị nguồn được nối với chân VCC (đơn vị Volt).

Cứ 10mV ứng với 1oC nên kết quả nhân với 100.

Lưu ý: Tất cả để ở dạng số thực để nhiệt độ trả về 1 số thực.

## Cảm biến MQ2

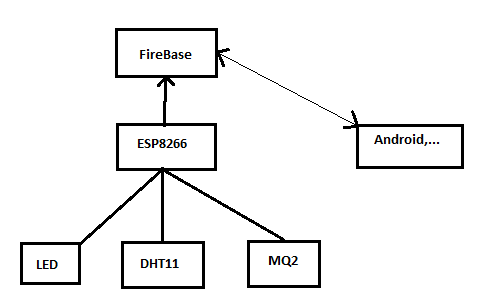
* MQ2 là cảm biến khí, dùng để phát hiện các khí có thể gây cháy. Nó được cấu tạo từ chất bán dẫn SnO2. Chất này có độ nhạy cảm thấp với không khí sạch. Nhưng khi trong môi trường có chất ngây cháy (khí gas), độ dẫn của nó thay đổi ngay. Chính nhờ đặc điểm này người ta thêm vào mạch đơn giản để biến đổi từ độ nhạy này sang điện áp.
* Khi môi trường sạch điện áp đầu ra của cảm biến thấp, giá trị điện áp càng tăng khi nồng độ khí gây cháy xung quanh MQ2 càng cao.
* MQ2 có 4 chân giao tiếp: 1 chân GND nối đất, 1 chân VCC nối nguồn, 1 chân Digital Out xuất ra tín hiệu số và chân Analog Out xuất ra tín hiệu tương tự (chân này được sử dụng để đo nồng độ khí gas)
* Trong mạch được lắp đặt, vì không có công thức chuyển đổi trực tiếp từ điện áp đầu ra chân AO sang độ ppm (đơn vị đo nồng độ gas trong không khí) nên nhóm thực hiện đặt ngưỡng cảnh báo như sau:
* Tiến hành đo trong các môi trường không có khí gas (tạm gọi là môi trường “sạch”) ở các điều kiện phòng ngủ, phòng làm việc, ngoài trời,…
* Lấy khí gas từ bật lửa gas cho cảm biến đọc và ghi lại giá trị điện áp đầu ra, thực hiện nhiều lần với mức độ khí gas khác nhau.
* Kết hợp với kiến thức thực tế: độ ppm lớn hơn 5000 được coi là nguy hiểm, từ đó rút ra công thức để sử dụng: **Độ ppm = Điện áp đầu ra \* 3** (3 là điện áp đầu vào – giá trị nguồn nối chân VCC).

## Kết quả hình ảnh cho led thườngCác thành phần khác

* Đèn LED:
* Loại 5mm.
* Sáng màu đỏ - thể hiện cảnh báo.
* Điện áp đầu vào 1.9V – 3.5V.
* Chân ngắn là chân dương nối với 1 trong các chân D1-D8 của ESP8266 để cấp nguồn cho LED. Việc bật tắt hay nháy đèn cũng sẽ được điều khiển qua chân này.
* Chân dài là chân âm nối đất (GND).

# Hoạt động của hệ thống

## Sơ đồ tổng quát

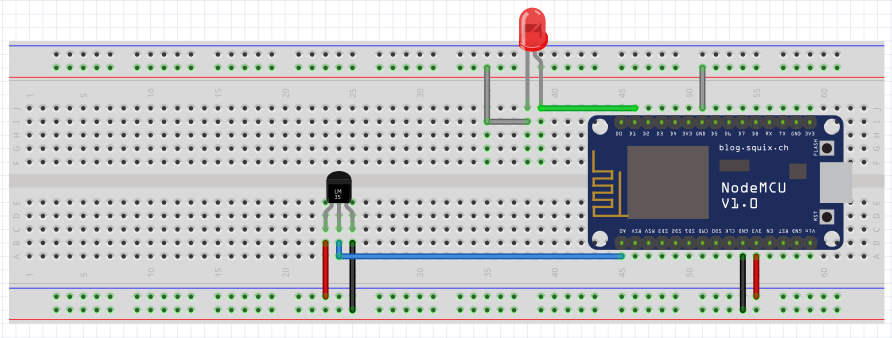


## Mô tả hoạt động

* Firebase đóng vai trò trung gian lưu trữ dữ liệu được các ESP8266 và Android gửi lên, gồm dữ liệu về nhiệt độ, nồng độ khí co, lpg, smoke và một biến kiểm tra việc tắt/bật cảnh báo, gửi dữ liệu đó đến các Client khi được yêu cầu.
* Hệ thống sử dụng 1 chip ESP8266, 1 cảm biến DHT11, 1 cảm biến MQ2 và 1 LED đỏ.
* Chip thứ nhất (điện áp ra 3V) có nhiệm vụ giám sát nồng độ khí gas trong không khí, được kết nối với cảm biến MQ2, 1 đèn LED. Khi được cấp nguồn, dữ liệu đọc từ cảm biến MQ2 sẽ được gửi qua chip dưới dạng analog, tại đây tiến hành xử lý đổi qua đơn vị ppm (đơn vị đo nồng độ khí gas) và gửi lên Server liên tục 1 giây 1 lần. Ở điều kiện thường đèn LED tắt, còn khi nồng độ khí gas vượt quá 5000ppm được coi là nguy hiểm, đèn LED sẽ sáng và chuông ở điện thoại kêu liên tục để cảnh báo.
* Cảm biến DHT11 có nhiệm vụ giám sát nhiệt độ tại phòng, được kết nối với ESP8266. Khi được cấp nguồn, dữ liệu đọc từ cảm biến DHT11 sẽ được gửi qua chip dưới dạng digital, tại đây tiến hành xử lý đổi qua đơn vị độ C và gửi lên Server liên tục 1 giây 1 lần.
* Giao diện Android: dữ liệu bắt đầu được gửi khi người dùng khởi động ứng dụng, khi đó người dùng có thể theo dõi nồng độ khí gas, nhiệt độ và trạng thái bật/tắt của cảnh báo nói chung trên giao diện. Các dữ liệu được cập nhật liên tục 1 giây 1 lần. Ngoài ra khi cảnh báo được bật, vì 1 lí do nào đó người dùng muốn tắt cảnh báo thì cũng có thể thực hiện được, sau khi tắt có thể đặt lại cảnh báo nếu muốn.

# Thiết kế phần cứng

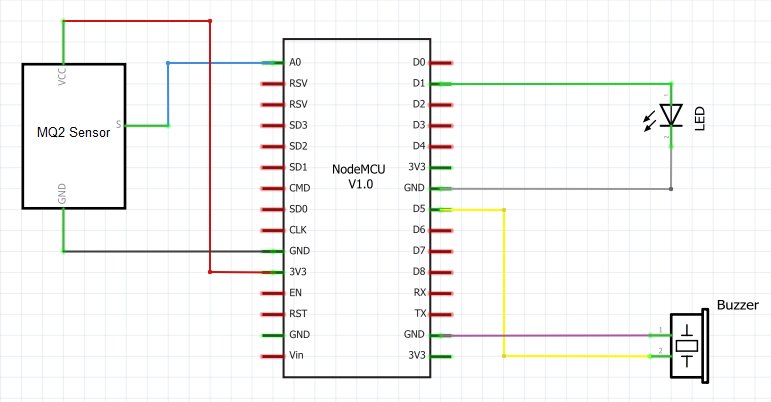
## Mạch giám sát nhiệt độ với DHT11

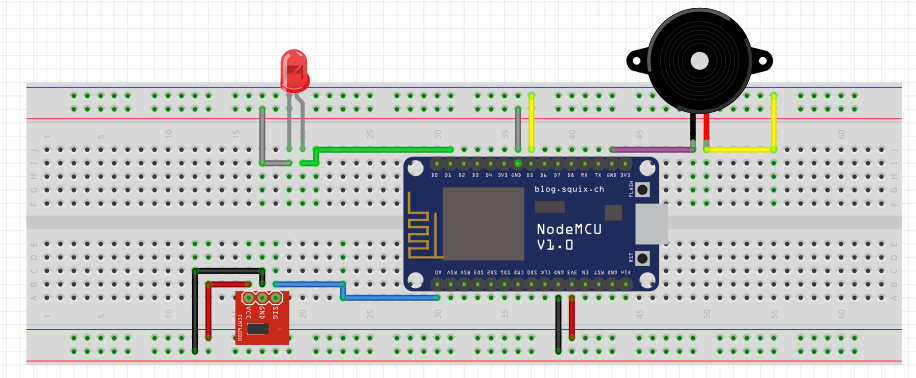


Nối chân và quy ước màu dây điện:

|  |  |  |  |
| --- | --- | --- | --- |
| **DHT11** | **ESP8266** | **LED** | **Màu dây** |
| VCC | 3V3 |  | Đỏ |
| Do | D5 |  | Xanh nước biển |
| GND | GND |  | Đen |
|  | D7 | Dương (dài) | Xanh lá cây |
|  | GND | Âm (ngắn) | Xám |

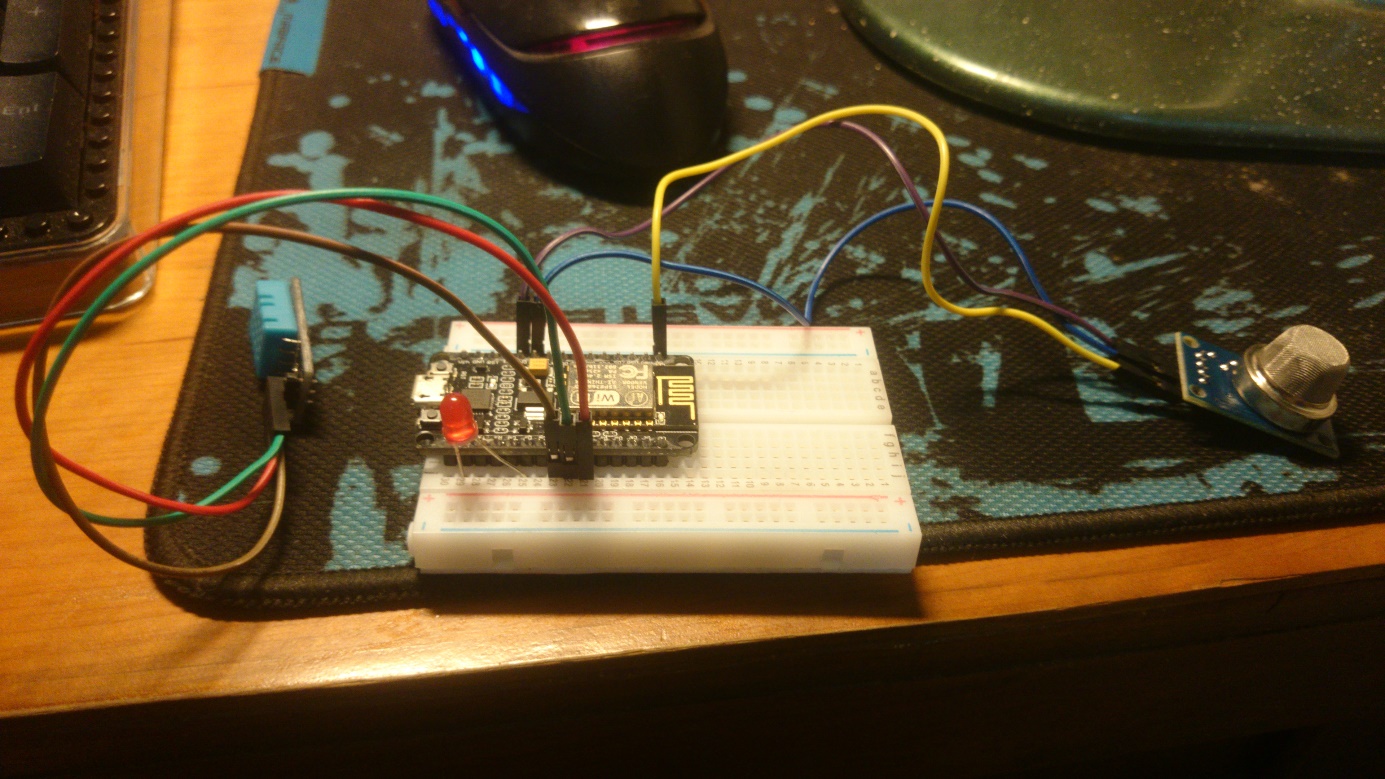
## Mạch giám sát nồng độ khí gas với MQ2

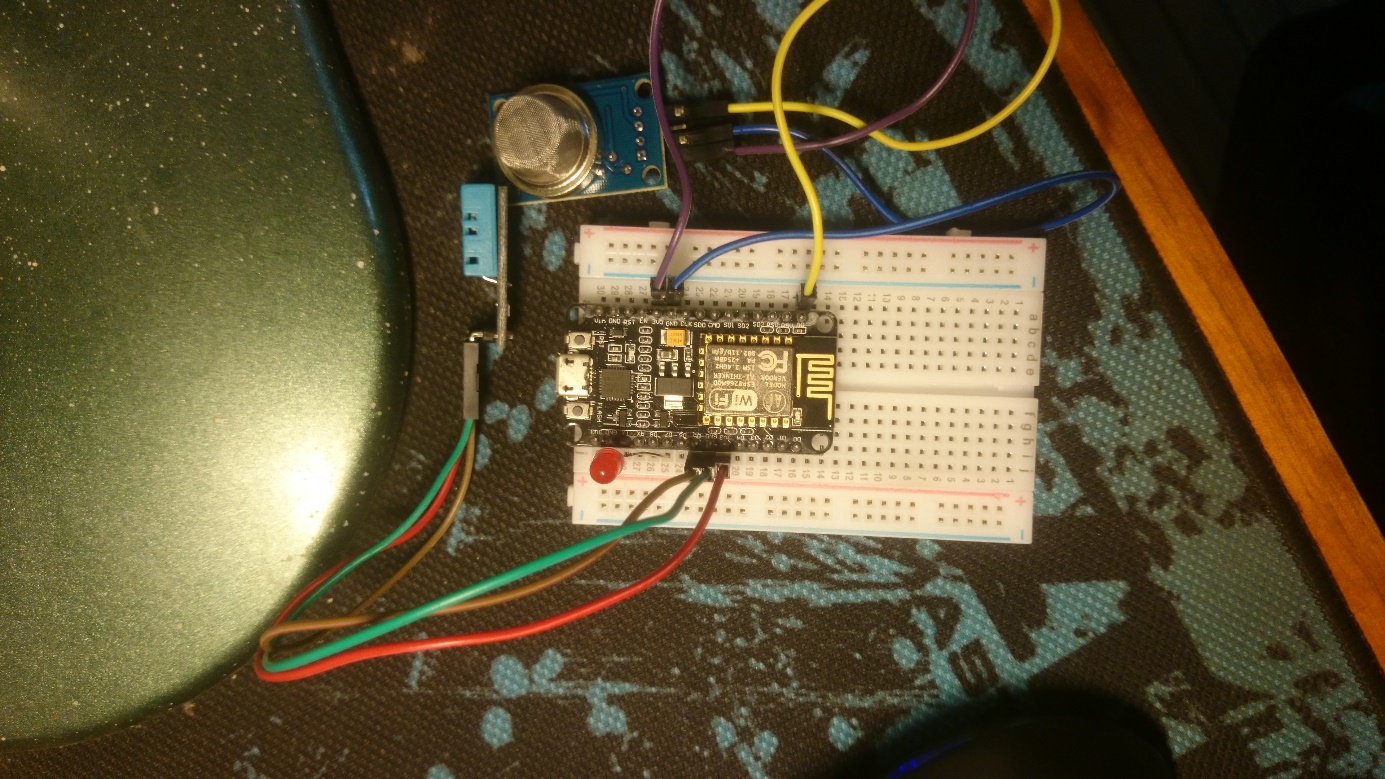




|  |  |  |
| --- | --- | --- |
| **MQ2** | **ESP8266** | **Màu dây** |
| VCC | 3V3 | Đỏ |
| A0 | A0 | Xanh nước biển |
| GND | GND | Đen |

## Mạch lắp đặt thực tế



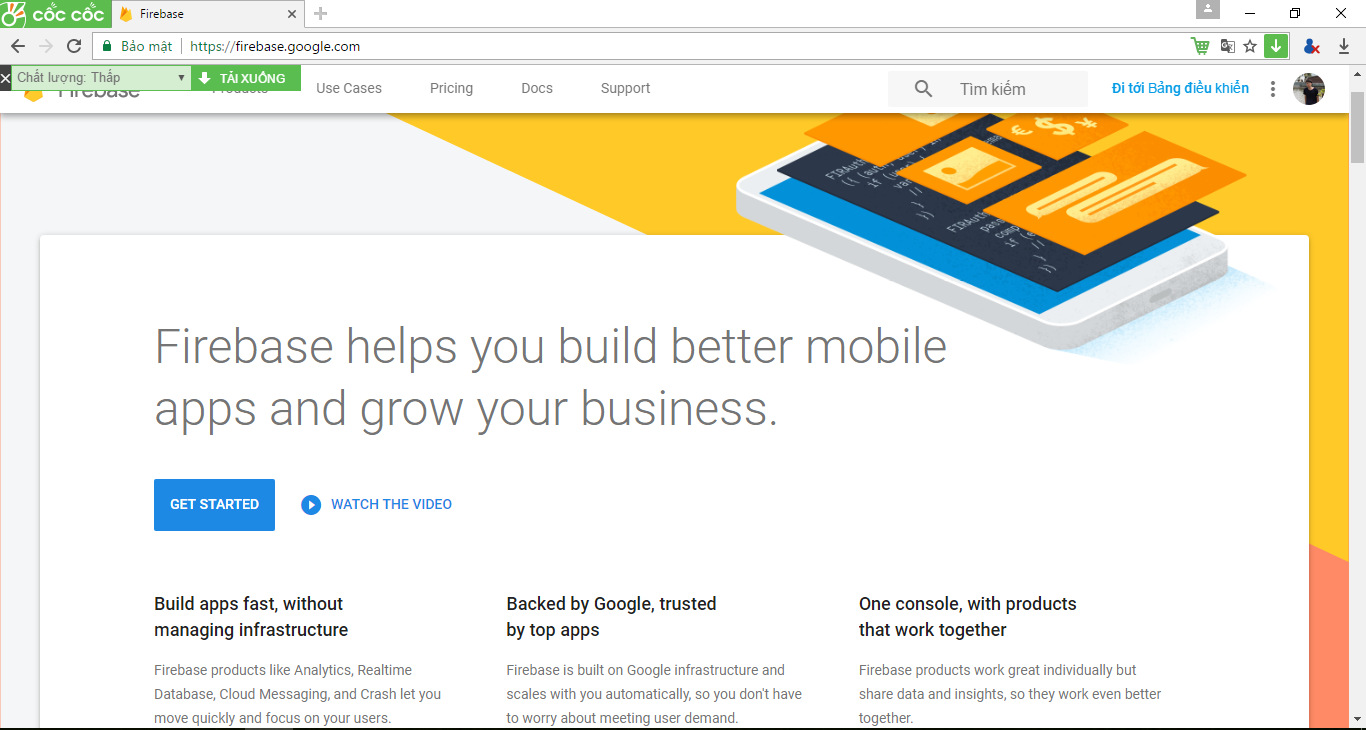


Vì điện áp của các thiết bị đều ở mức không quá cao nên việc cấp nguồn có thể lấy trực tiếp thông qua máy tính và chip ESP8266 mà không cần nguồn riêng. Trên thực tế để việc sử dụng được thuận tiện có thể dùng 2 cục pin 1.5V để cấp nguồn (pin Con Thỏ,…)

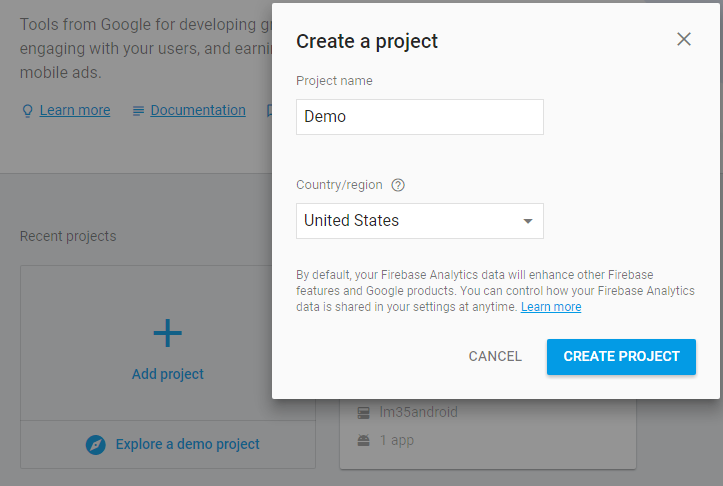
# Thiết kế phần mềm

## Cài đặt Firebase.

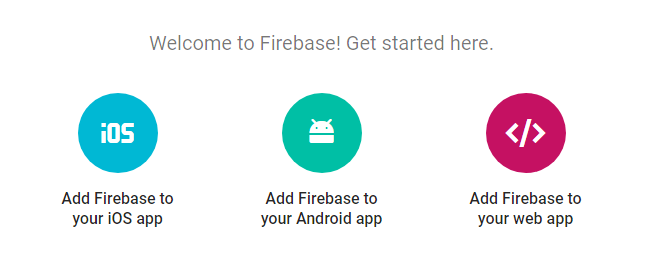
**Bước 1:** Vào trang <https://firebase.google.com/> và tạo 1 project mới: Chọn “Get Started”



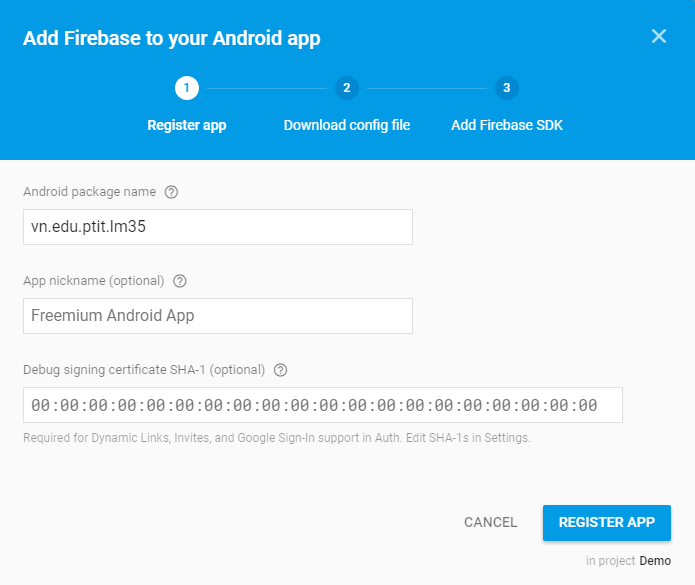
**Bước 2:** Chọn “Add project” và đặt tên cho Project, Region để mặc định United States



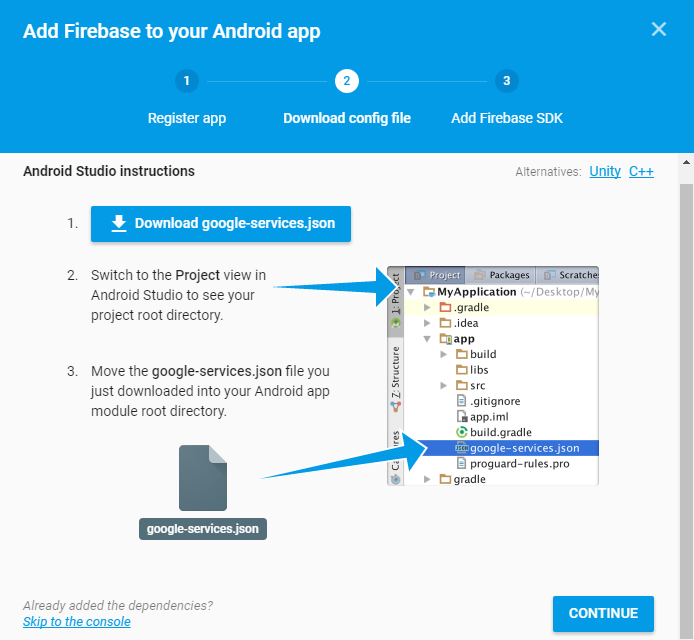
**Bước 3:** Chọn “Add Firebase to your Android app”



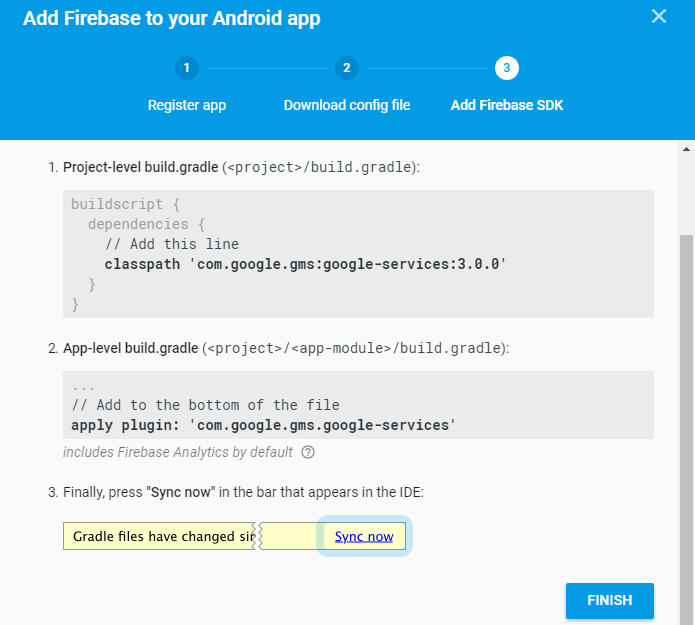
**Bước 4:** Điền tên package ứng dụng android muốn kết nối đến Firebase và chọn “Register App”



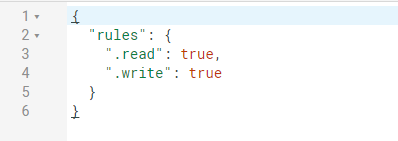
**Bước 5:** Click “Download google-services.json” để tải file (nếu chưa có) sau đó click “Continue”



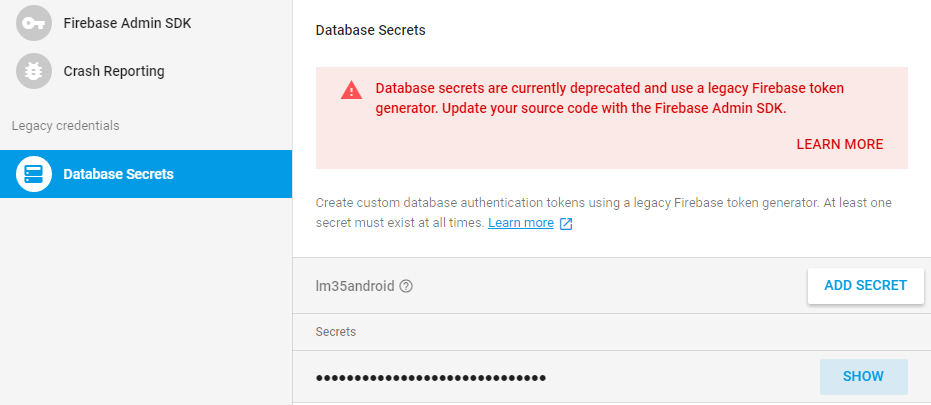
**Bước 6:** Copy các dòng theo chỉ dẫn vào các file và click “Finish”



**Bước 7:** Vào Database **🡪** Rules và đổi “null” thành “true” như sau



**Bước 8:** Chọn Projects Settings **🡪** Service Accounts **🡪** Database Secrets và click nút “SHOW” để lấy API key xác thực sẽ dùng cho Arduino.



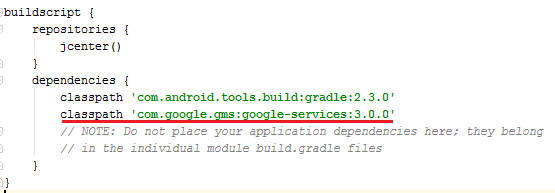
## Lập trình Arduino

* Thư viện cho Arduino: FirebaseArduino Library
* Thêm thư viện: từ Arduino chọn **Sketch 🡪 Include Library 🡪 PubSubClient**, **Sketch 🡪 Include Library 🡪FireBaseArduino**
* Tải thư viện FirebaseArduino: truy cập địa chỉ <https://github.com/firebase/firebase-arduino/tree/master/examples/FirebaseRoom_ESP8266> và tải file ZIP. Sau đó chọn **Sketch 🡪 Include Library 🡪 Add .ZIP**, chọn file vừa tải về.
* Ý tưởng:
* Kết nối đến mạng Wifi.
* Set chân đầu ra cho đèn LED và chuông.
* Đọc dữ liệu từ cảm biến và xử lý, in ra Serial Monitor để kiểm tra.
* Gọi hàm Firebase.set() để gửi dữ liệu lên Firebase.
* Dùng 1 biến boolean ledON để xử lý yêu cầu tắt cảnh báo và đặt lại cảnh báo của người dùng.
* Code chi tiết trong các file .ino đính kèm. Chip ESP8266 được nạp chương trình đo nồng độ khí gas với MQ2 và đo nhiệt độ với DHT11.

## Lập trình Android

* Thêm các Services cho Project
* Mở file build.gradle(Project: Demonhung) – file build ở bên trên và thêm dòng:

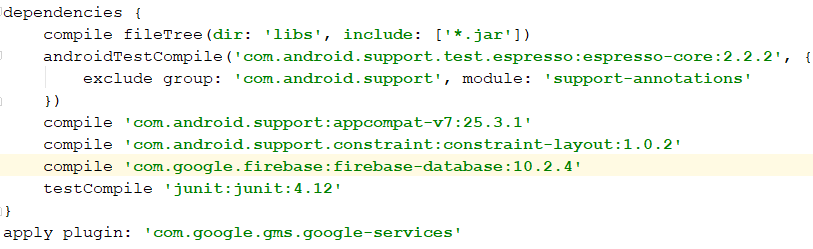
*classpath 'com.google.gms:google-services:3.0.0'*



* Mở file build.gradle(Module: app) – file build ở bên dưới và thêm các dòng:

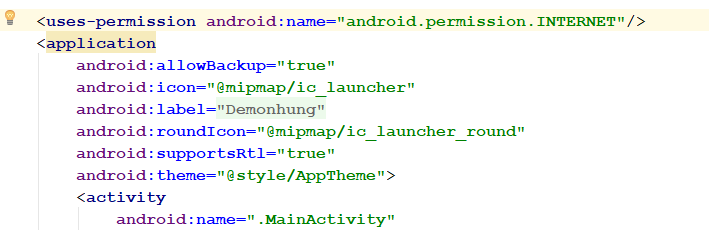
*compile 'com.google.firebase:firebase-database:10.2.0'*

*apply plugin: 'com.google.gms.google-services'*

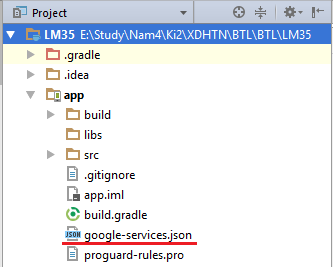


* Mở file AndroidManifest.xml và thêm các dòng:

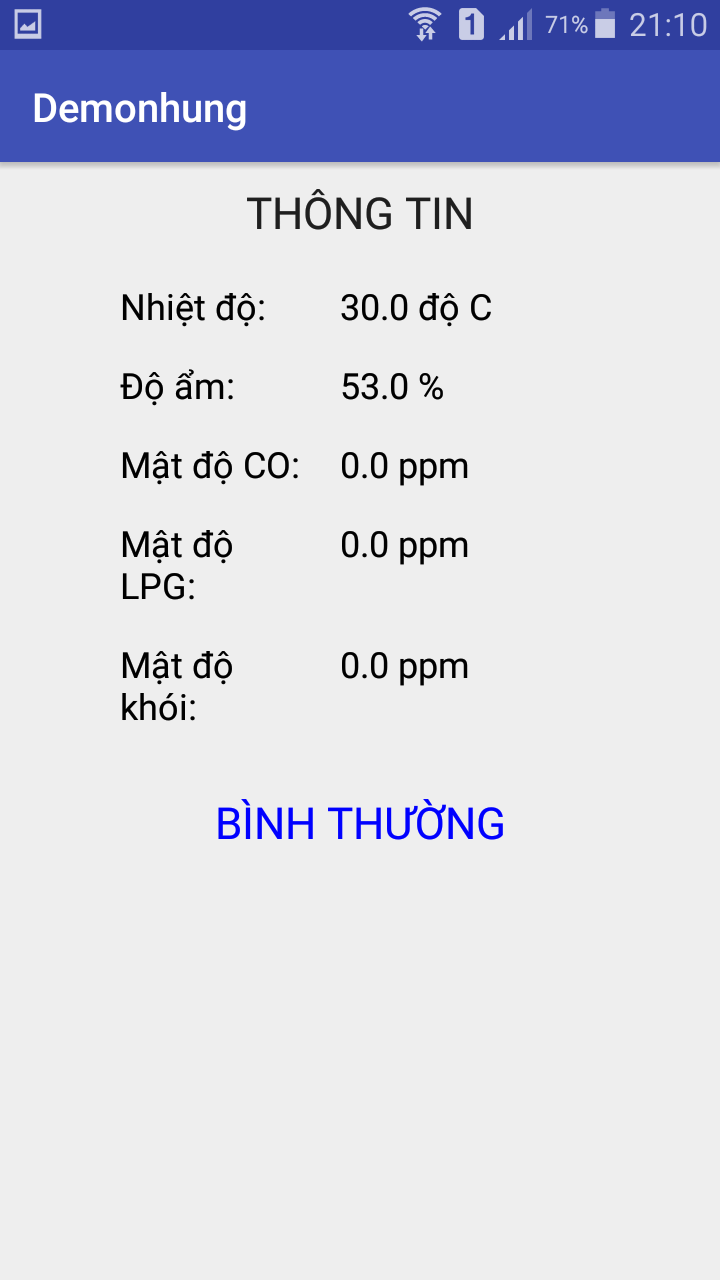
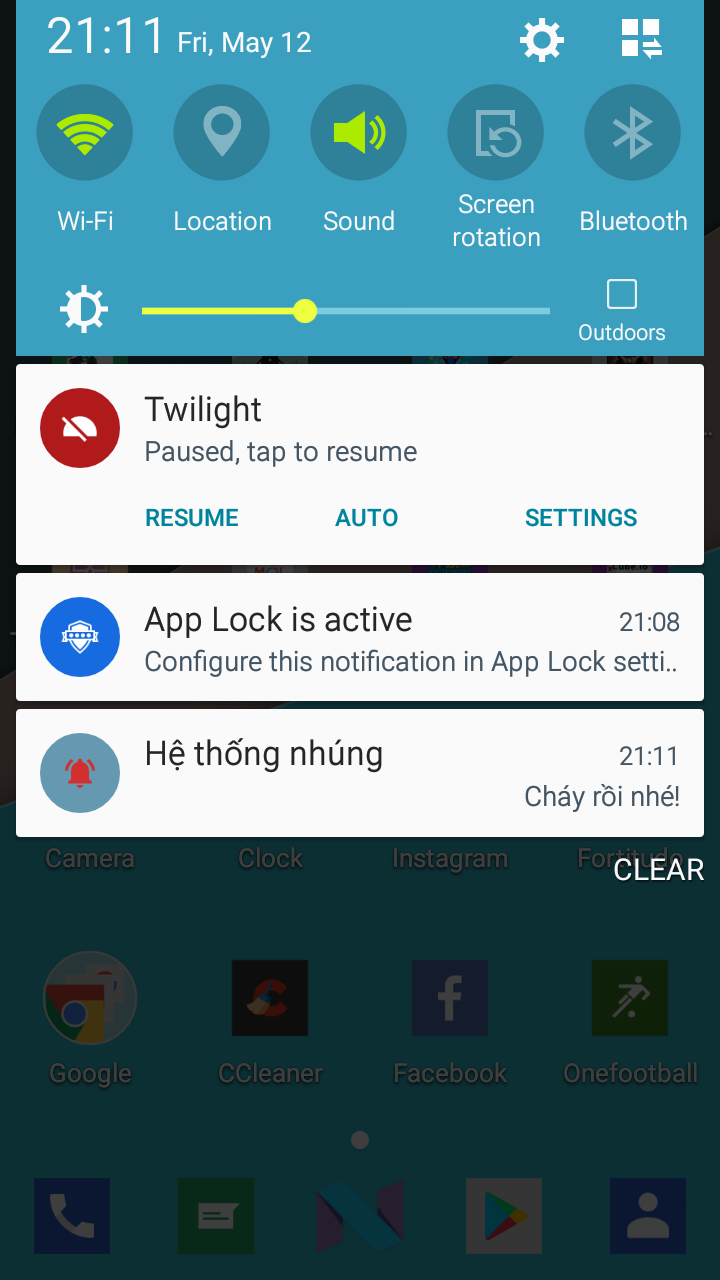
*<uses-permission android:name="android.permission.INTERNET" />*



* Tải file *google-services.json* về và đặt vào folder app



* Ý tưởng:
* Giao diện đơn giản.
* Có các dòng hiển thị nhiệt độ, độ ẩm, mật độ khí, trạng thái cảnh báo.
* Các giá đều được cập nhật 1 giây 1 lần.
* Có 1 lựa chọn tắt cảnh báo.
* Cho phép người dùng tắt cảnh báo khi có mật độ khí cao dễ gây cháy hay làm chết người, sau khi tắt cũng có thể đặt lại cảnh báo nếu muốn.
* Khi ứng dụng chạy ngầm sẽ gửi thông báo ra ngoài màn hình đến người dùng nếu nhiệt độ vượt quá các ngưỡng.
* Mỗi khi ứng dụng khởi động, trạng thái bật cảnh báo sẽ được tự động đặt lại.
* Giao diện:



# Tổng kết

Như vậy nhóm đã xây dựng hệ thống giám sát nhiệt độ, nồng độ khí gas. Vì thời gian có hạn, bên cạnh đó là kiến thức chưa được chuyên sâu và có những khó khăn nhất định chưa thể giải quyết, vì vậy quá trình thiết kế và cài đặt hệ thống không thể tránh được những sai sót. Nhóm đã cố gắng đưa ra những giải pháp thay thế và hoàn thành một cách tốt nhất trong khả năng có thể. Rất mong nhận được sự góp ý của thầy và các bạn để có thể xây dựng hệ thống một cách hoàn thiện hơn.

Xin chân thành cảm ơn thầy và các bạn!