



## Smart Air Purifier System Documentation

จัดทำโดย

ເພິ່ງພິຈາ ປີຍາວຣານທ໌ 6733185821

ວລັຍພຣຣນ ຕຣີທີພົມສກຸລ 6733236121

ອືສສຍາພຣຣນ ລິ້ມໍມ່ວງນິລ 6733293921

ສຸຈິນນົມ ຮອດເປັນ 6733277921

ໜຸ້ມມ ທັກນວິຫຼຽກຖະ 6733043321

ນໍາເສນອ

ຜ.ສ.ດຣ. ພຣົງຄົດເດັ່ນ ກິຮຕິພຣານທ໌

ຮ.ສ.ດຣ. ເສຣະຫຼາ ປານງາມ

ຜ.ສ.ດຣ. ພິຈູນະ ສີທີ່ອມຮ

รายวิชา 2110356 Embedded System ระบบฝังตัว

ภาคการศึกษาต้น ปีการศึกษา 2568

คณะวิศวกรรมศาสตร์ สาขาวิชาวิศวกรรมคอมพิวเตอร์และเทคโนโลยีดิจิทัล

จุฬาลงกรณ์มหาวิทยาลัย

## คำนำ

โครงการนี้จัดทำขึ้นเป็นส่วนหนึ่งของรายวิชา Embedded System ซึ่งมีวัตถุประสงค์เพื่อพัฒนาระบบอุปกรณ์อัจฉริยะ (Smart Device) เพื่อแก้ปัญหาต่าง ๆ ในชีวิตประจำวัน โดยการประยุกต์ใช้งานไมโครคอนโทรลเลอร์ เช่น ESP32, เซนเซอร์หลายประเภท รวมถึงระบบสื่อสารไร้สายและการเชื่อมต่อกับ Cloud Platform

แรงบันดาลใจของโครงการเกิดจากปัญหามลภาวะทางอากาศ โดยเฉพาะค่าฝุ่น PM2.5 ที่ส่งผลกระทบต่อสุขภาพของประชาชนในชีวิตประจำวัน จึงเป็นที่มาของการออกแบบระบบที่สามารถเฝ้าระวังคุณภาพอากาศ ตรวจจับค่าฝุ่นได้แบบเรียลไทม์ พร้อมทั้งสั่งงานเครื่องกรองอากาศอัตโนมัติเมื่อค่าฝุ่นเกินมาตรฐาน อีกทั้งยังมีฟังก์ชันแสดงผลผ่าน Dashboard และรองรับคำสั่งเสียงเพื่อเพิ่มความสะดวกในการใช้งาน

รายงานฉบับนี้อธิบายถึงแนวคิด การออกแบบระบบ วิธีการพัฒนา ผลการทดสอบ ตลอดจนข้อสรุปจากการดำเนินงานของผู้จัดทำ หวังเป็นอย่างยิ่งว่ารายงานนี้จะเป็นประโยชน์ต่อการเรียนรู้ และประยุกต์ใช้งานระบบสมองกลฝังตัว และสามารถต่อยอดเพื่อใช้งานจริงในอนาคตได้

ผู้จัดทำขอขอบคุณอาจารย์ผู้สอนและผู้เกี่ยวข้องทุกท่านที่ให้คำแนะนำและสนับสนุนให้โครงการนี้สำเร็จลุล่วงไปได้ด้วยดี

# สารบัญ

| เนื้อหา                                     | หน้า |
|---|------|
| คำนำ  | 2    |
| สารบัญ                                      | 3    |
| Introduction and Problem Statement          | 4    |
| Related Works                               | 5    |
| System Design                               | 7    |
| Overview Diagram                            | 7    |
| Sensor Node (DOIT ESP32 DEVKIT V1)          | 8    |
| Gateway (DOIT ESP32 DEVKIT V1)              | 8    |
| Cloud / Storage                             | 9    |
| Dashboard + Source code                     | 10   |
| Test Results                                | 11   |
| Discussion and conclusion                   | 12   |
| Role and responsibility of each team member | 13   |

## Introduction and Problem Statement

ปัญหามลภาวะทางอากาศ โดยเฉพาะค่าฝุ่นละออง PM2.5 ถือเป็นหนึ่งในปัญหาด้านสิ่งแวดล้อมและสุขภาพที่ส่งผลกระทบต่อร่างกาย ทั้งในเขตเมืองใหญ่ พื้นที่จราจรหนาแน่น รวมถึงช่วงฤดูกาลที่มีการเผาในที่โล่ง การสัมผัสฝุ่นละอองขนาดเล็กเป็นเวลานานแม้ภายในบ้าน สามารถเพิ่มความเสี่ยงในการเกิดโรคระบบทางเดินหายใจ ภูมิแพ้ และโรคหัวใจได้ ทำให้ความสามารถในการเผาติดตามคุณภาพอากาศภายในอาคารและการจัดการคุณภาพอากาศอย่างทันท่วงทันนั้นเป็นสิ่งสำคัญในชีวิตประจำวัน

จากปัญหาดังกล่าว ผู้จัดทำจึงพัฒนา ระบบเครื่องกรองอากาศอัจฉริยะ (Smart Air Purifier System) ที่สามารถตรวจวัดคุณภาพอากาศด้วยเซนเซอร์หลักประเภท เช่น เซนเซอร์วัดค่าฝุ่น PM2.5, อุณหภูมิ-ความชื้น, ก้าชคุณภาพอากาศ รวมถึงไมโครโฟนสำหรับใช้รับคำสั่งเสียง ระบบจะส่งข้อมูลจาก Sensor Node ไปยัง Gateway Node เพื่อประมวลผล ควบคุมการทำงานของพัดลมผ่านรีเลย์ และส่งข้อมูลขึ้น Cloud แสดงผลแบบเรียลไทม์ผ่าน Dashboard

นอกจากนี้ระบบยังสามารถเปิด-ปิดเครื่องกรองอากาศอัตโนมัติเมื่อค่าฝุ่นเกินระดับที่กำหนด พร้อมทั้งรองรับการแจ้งเตือนผู้ใช้งาน Notification(Blynk) และรองรับการสั่งงานด้วยเสียง เพื่อเพิ่มความสะดวกในการใช้งาน โดยทั้งหมดนี้ถูกพัฒนาภายใต้แนวคิดของการสร้างอุปกรณ์ IoT แบบ Smart Device ที่ผสานการทำงานของไมโครคอนโทรลเลอร์ การประมวลผลข้อมูล การสื่อสารไร้สาย และระบบ Cloud เข้าด้วยกัน

โครงการนี้มีเป้าหมายเพื่อพัฒนาต้นแบบอุปกรณ์ที่ช่วยลดความเสี่ยงจากการลพิษทางอากาศ ใช้งานง่าย อีกทั้งยังเป็นการประยุกต์ความรู้ด้าน Embedded Systems, IoT, Sensor Integration และ Cloud Computing ตามวัตถุประสงค์ของรายวิชาได้อย่างครบถ้วน

## Related Works

### ระบบตรวจคุณภาพอากาศ

งานวิจัยในหัวข้อ Smart Air Quality Guardian (2023) โดย Rifki Fajar Nugraha และคณะ ได้นำเสนอระบบตรวจคุณภาพอากาศที่ใช้ ESP32 ร่วมกับเซนเซอร์ MQ-2 และ MQ-135 เพื่อรวบรวมข้อมูลเกี่ยวกับมลพิษในอากาศ เช่น ควัน แก๊สไวไฟ และสารประกอบอินทรีย์ระเหย (VOC) โดยข้อมูลทั้งหมดถูกส่งผ่านระบบ IoT เพื่อแสดงผลแบบเรียลไทม์บนหน้าจอและบนแพลตฟอร์มออนไลน์ ทำให้ผู้ใช้สามารถตรวจสอบคุณภาพอากาศได้อย่างต่อเนื่อง ระบบดังกล่าวเน้นการผ่อนผันงานระหว่างไมโครคอนโทรลเลอร์ เซนเซอร์ตรวจคุณภาพอากาศ และการส่งข้อมูลผ่านเครือข่ายเพื่อให้เหมาะสมกับการใช้งานในพื้นที่อาคารหรือตามจุดต่างๆ ที่ต้องการเฝ้าระวังด้านคุณภาพอากาศ

สำหรับโปรเจกต์นี้จะใช้ ESP32 ร่วมกับเซนเซอร์ MQ-135 เช่นเดียวกับงานวิจัยดังกล่าว แต่จะเน้นการนำข้อมูลที่ตรวจวัดได้ไปประยุกต์ใช้ร่วมกับระบบ Smart Device ที่มีเซนเซอร์หลายชนิด และเชื่อมต่อกับ Dashboard หรือระบบแจ้งเตือนออนไลน์ เพื่อให้ผู้ใช้ทราบค่าต่างๆ และทราบสถานะการเปิด-ปิดเครื่องกรองอากาศ

### ระบบตรวจคุณภาพอากาศ PM2.5

งานในวิดีโอ Making Embedded Systems – PM2.5 Monitor โดย Elicia White ได้นำเสนอการสร้างอุปกรณ์ตรวจคุณภาพอากาศ PM2.5 โดยใช้เซนเซอร์วัดฝุ่น GP2Y1014AUOF ซึ่งเป็น Optical Dust Sensor ที่ได้รับความนิยมในการประเมินคุณภาพอากาศในร่ม อุปกรณ์ดังกล่าวถูกพัฒนาให้สามารถอ่านค่าฝุ่นแบบเรียลไทม์ผ่านไมโครคอนโทรลเลอร์ โดยมีการแสดงผลข้อมูลบนจอและสามารถบันทึกข้อมูลเพื่อการวิเคราะห์เพิ่มเติมได้ จุดสำคัญของงานนี้คือการออกแบบระบบ Embedded ให้มีความเสถียร ประหยัดพลังงาน และรองรับการใช้งานต่อเนื่อง เหมาะสำหรับจะเป็นต้นแบบของระบบตรวจคุณภาพอากาศที่ใช้งบไม่สูงมาก

(อ้างอิง: <https://www.youtube.com/watch?v=uus0jh8PfVg>)

สำหรับโปรเจกต์จากวิดีโอนี้ ทำให้ได้อิเดียในการเลือกใช้เซนเซอร์วัดฝุ่น และนำมาต่อยอดกับระบบเครื่องกรองอากาศ โดยทดลองใช้ GP2Y1014AUOF แล้ว แต่ได้เห็นถึงความไม่เสถียร จึงได้ลองใช้ PMS5003 แทน ดังนั้นจึงถือว่าแหล่งอ้างอิงนี้เป็นไอเดียเริ่มต้น ให้ทีมได้ลองพัฒนาต่อยอดให้ดียิ่งขึ้น

## การสั่งเปิด-ปิดด้วยเสียง

โปรเจกต์ในวิดีโอ We can create a voice command system! ได้นำเสนอการสร้างระบบควบคุมอุปกรณ์ด้วยเสียง โดยใช้ ESP32-S3 ร่วมกับโมดูลไมโครโฟน INMP441 และ Espressif Speech Recognition framework ในการตรวจจับคำสั่งเสียง ระบบสามารถจดจำ wake word เช่น “Hey Siri” หรือ “Ok Google” เพื่อเปิดใช้งานการฟังคำสั่ง และตามด้วยคำสั่งเสียงที่ถูกกำหนด เช่น การควบคุมไฟ LED หรืออุปกรณ์อื่น ๆ ให้ทำงานตามที่สั่งได้ทันที จุดเด่นคือไม่จำเป็นต้องเชื่อมต่อ กับ Cloud หรือ Access Point ทำให้ระบบมีความเร็วในการตอบสนองสูงและสามารถนำไปประยุกต์ ใช้กับงาน IoT และ Smart Home ได้อย่างมีประสิทธิภาพ

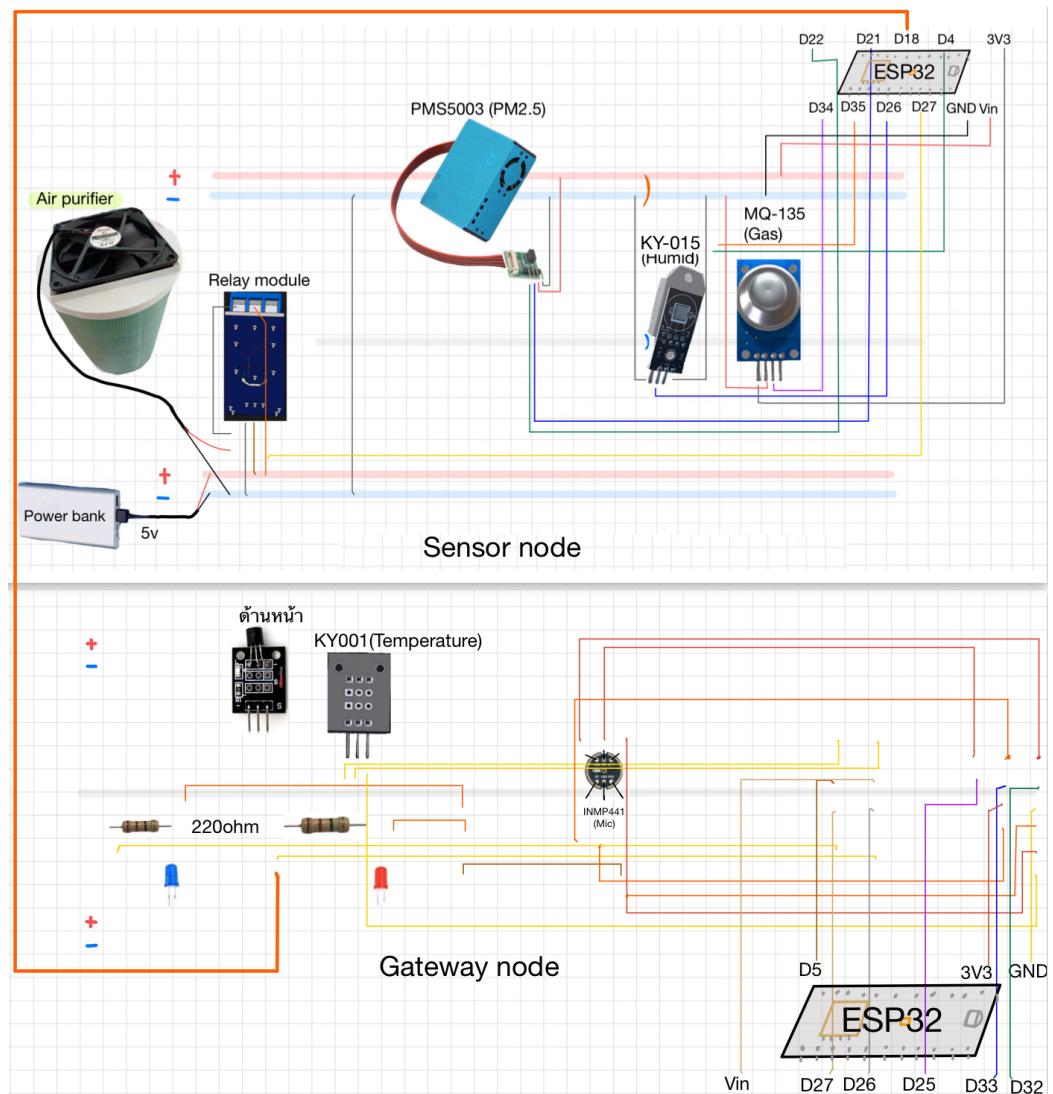
(อ้างอิง: <https://www.youtube.com/watch?si=mvfna4cnh4>)

สำหรับโปรเจกต์จากวิดีโอนี้ ทำให้ได้ไอเดียในทำ speech recognition โดยทีมของเราจะใช้คำว่า “Hey open”, “Hey close” ในการควบคุม LED และนำมาต่อยอดในการควบคุมการเปิด-ปิดของเครื่องกรองอากาศ

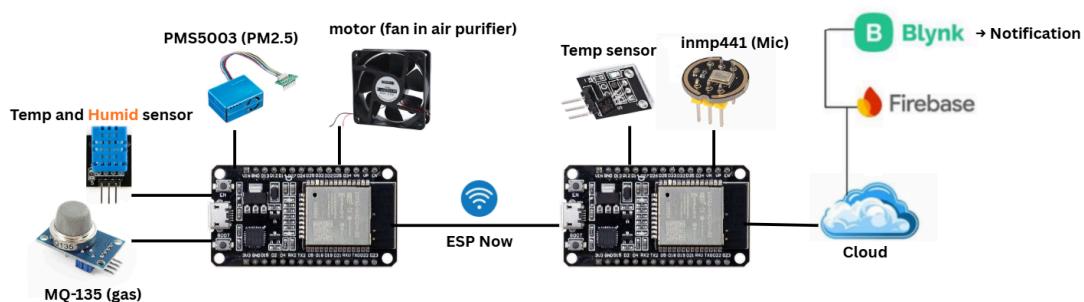
# System Design

## Overview Diagram

### 1. Wiring diagram: Sensor node and Gateway node

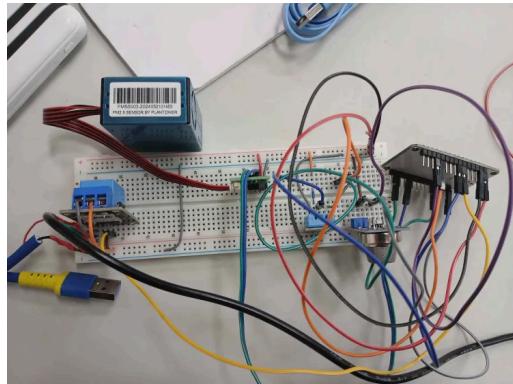


### 2. Architecture diagram



## Sensor Node (DOIT ESP32 DEVKIT V1)

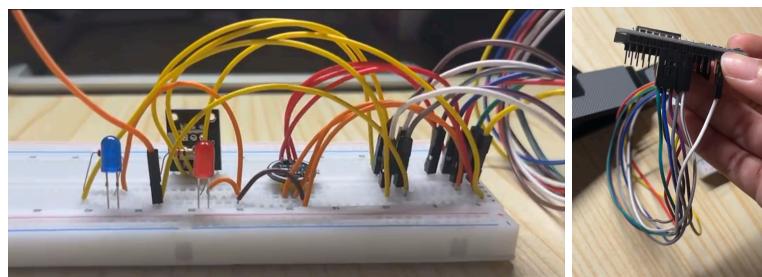
ทำหน้าที่เป็นตัวรับข้อมูลจาก PMS5003 (วัดฝุ่น PM2.5), MQ-135 (วัดแก๊ส) และ KY-015 (วัด Humidity) โดยจะนำข้อมูลที่ได้ส่งต่อไปยัง Gateway ด้วย ESP-Now อีกทั้งยังเป็นบอร์ดที่มีการรับข้อมูลและเชื่อม relay module เพื่อเป็น switch ในการควบคุมการทำงานของเครื่องกรองอากาศด้วย



## Gateway (DOIT ESP32 DEVKIT V1)

ทำหน้าที่เป็นตัวรับข้อมูลจากเซนเซอร์ KY-001 (วัด Temperature) และ INMP441 (Mic for voice command) และรับข้อมูลจาก Sensor Node ผ่าน ESP-Now โดยระบบจะนำข้อมูลที่ได้รับมาประมวลผล ได้แก่

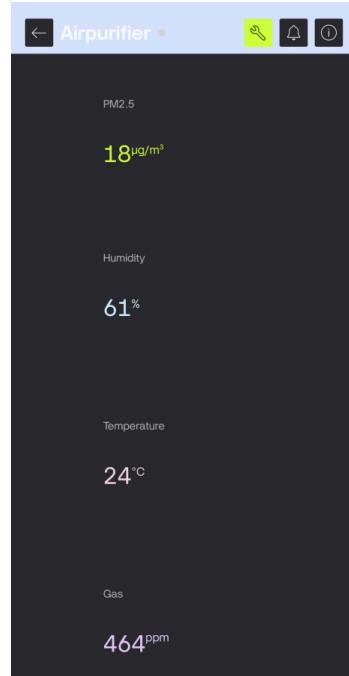
- หากมีคำสั่งเสียง “Hey Open” ระบบจะทำการเปิดไฟ เพื่อให้เห็นว่ามีการรับ voice command แล้ว แต่เครื่องกรองจะยังไม่เปิด จนกว่าค่าฝุ่นจะถึง  $50 \mu\text{g}/\text{m}^3$
- หากมีคำสั่งเสียง “Hey Close” ระบบจะทำการปิดไฟและเครื่องกรองทันที ไม่ว่าค่าฝุ่นจะอยู่ที่ค่าเท่าใด (ตัวอย่างสถานการณ์: หากผู้ใช้ต้องออกจากห้องแล้ว แต่เครื่องกรองอากาศยังไม่ปิด เนื่องจากค่าฝุ่นยังลดลงไม่ถึงระดับ  $40 \mu\text{g}/\text{m}^3$  ก็จะต้องใช้คำสั่งเสียง เพื่อให้เครื่องกรองอากาศปิดแทน)
- มีการแจ้งเตือนผ่าน Blynk (Email) ไปยังผู้ใช้งาน และระบบจะทำการส่งข้อมูลทั้งหมดที่ได้ไปยัง Cloud Storage ได้แก่ Blynk และ Firebase ผ่านทาง WiFi เพื่อแสดงค่าต่างๆ ใน Dashboard



## Cloud / Storage

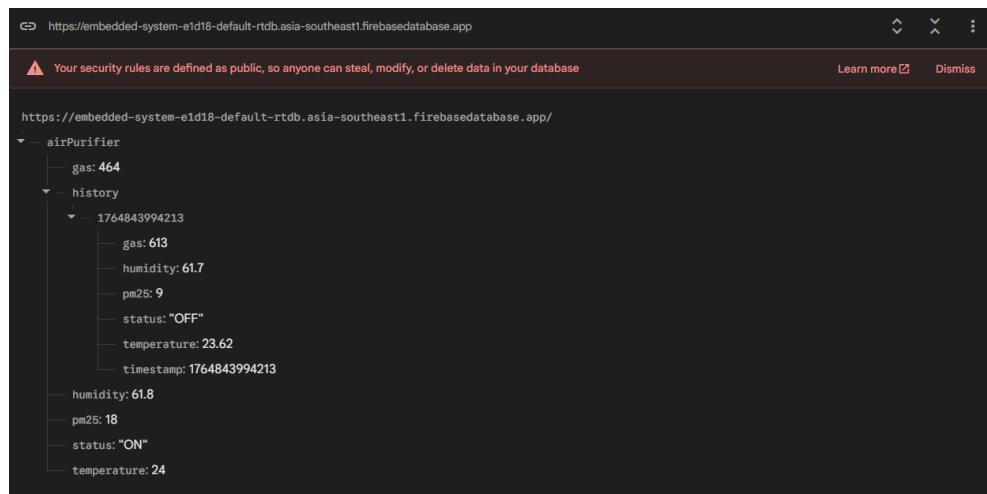
### 1. Blynk

- เก็บสถานะการเปิด-ปิดเครื่องกรองอากาศ, ค่าฝุ่น PM2.5, อุณหภูมิ, ความชื้นและแก๊ส



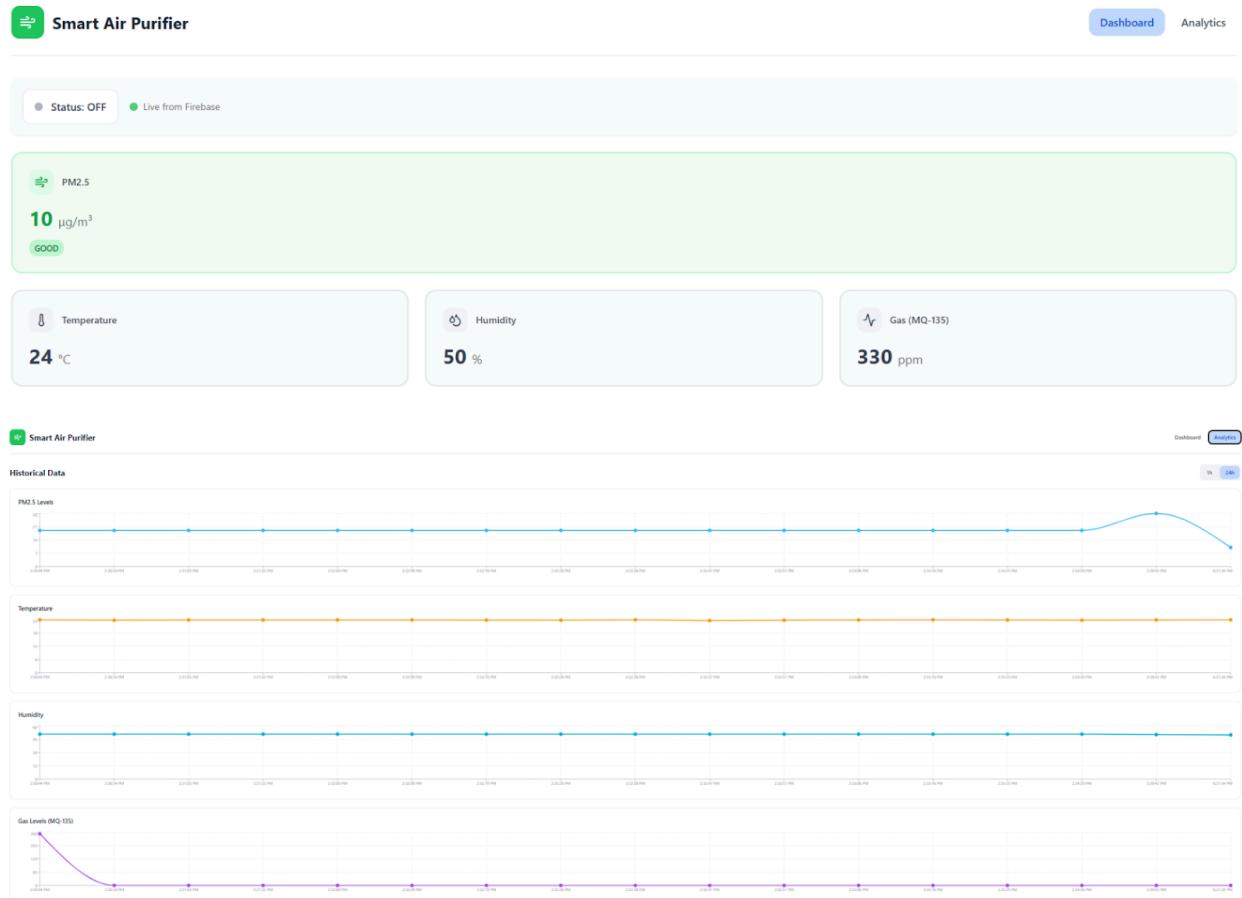
### 2. Firebase

- เก็บสถานะการเปิด-ปิดเครื่องกรองอากาศ, ค่าฝุ่น PM2.5, อุณหภูมิ, ความชื้นและแก๊ส



## Dashboard

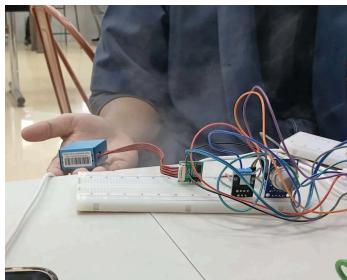
แสดงสถานะการเปิด-ปิดเครื่องกรองอากาศ, ค่าฝุ่น PM2.5, อุณหภูมิ, ความชื้นและแก๊ส รวมทั้งมีกราฟที่แสดงให้เห็นการเปลี่ยนแปลงแต่ละค่าในช่วงเวลาต่าง ๆ



Source code: [S-Gout/Embedded-Project](#)

## Test Results

| Test No. | Test Metric                     | Actions/Inputs                            | Expected Result  | Actual Result  |
|----------|---------------------------------|---|--|--|
| 1        | Detection of PM2.5 sensor       | ค่าฝุ่น $\geq 50 \mu\text{g}/\text{m}^3$  | เครื่องกรองเปิดทำงาน   | เครื่องกรองเปิดทำงาน   |
| 2        |                                 | ค่า $\leq 40 \mu\text{g}/\text{m}^3$      | เครื่องกรองปิด / ยังไม่เปิดทำงาน                                     | เครื่องกรองปิด / ยังไม่เปิดทำงาน                                     |
| 3        | Detection of humidity sensor    | เซ็นเซอร์ตรวจจับความชื้น                  | แสดงผลใน Dashboard   | แสดงผลใน Dashboard   |
| 4        | Detection of temperature sensor | เซ็นเซอร์ตรวจจับอุณหภูมิ                  | แสดงผลใน Dashboard   | แสดงผลใน Dashboard   |
| 5        | Detection of gas sensor         | เซ็นเซอร์ตรวจจับแก๊ส                      | แสดงผลใน Dashboard   | แสดงผลใน Dashboard   |
| 6        | Voice command                   | รอหlodotไฟสีน้ำเงินติดเพื่อรับสัญญาณเสียง | ไฟติด และพร้อมรับสัญญาณเสียง   | ไฟติด และพร้อมรับสัญญาณเสียง   |
| 7        |                                 | พูด “Hey Open”                            | ไฟสีแดงติด   | ไฟสีแดงติด   |
|          |                                 |   | เครื่องกรองไม่เปิดทำงาน หากค่าฝุ่นไม่ถึง $50 \mu\text{g}/\text{m}^3$ | เครื่องกรองไม่เปิดทำงาน หากค่าฝุ่นไม่ถึง $50 \mu\text{g}/\text{m}^3$ |
| 8        |                                 | พูด “Hey Close”                           | เครื่องกรองปิดทำงาน  | เครื่องกรองปิดทำงาน  |



Sensor รับฝุ่น



พัดลมเปิดทำงาน



พัดลมปิดการทำงาน

## **Discussion and conclusion**

โครงการนี้ได้นำเสนอระบบเครื่องกรองอากาศอัจฉริยะ (Smart Air Purifier System) ที่สามารถตรวจคุณภาพอากาศได้อย่างแม่นยำด้วยเซ็นเซอร์หลากหลายชนิด เช่นเซ็นเซอร์วัดค่าฝุ่น PM2.5, อุณหภูมิ, ความชื้น, ก้าชคุณภาพอากาศ พร้อมส่งข้อมูลไปยัง Gateway เพื่อประมวลผลควบคุมพัดลมอัตโนมัติ และอัปโหลดข้อมูลเข้า Cloud ให้ผู้ใช้ติดตามแบบเรียลไทม์ผ่าน Dashboard นอกจากนี้ ระบบยังรองรับการแจ้งเตือนผ่าน Blynk และการสั่งงานด้วยเสียง เพื่อเพิ่มความสะดวกในการใช้งาน ซึ่งระบบต้นแบบนี้ยังสะท้อนให้เห็นถึงการบูรณาการความรู้ด้าน Embedded Systems, IoT, Sensor Integration และ Cloud Computing เพื่อนำไปสู่การพัฒนานวัตกรรมที่ช่วยลดความเสี่ยงจากมลพิษทางอากาศและเพิ่มประสิทธิภาพการใช้งานอุปกรณ์อัจฉริยะในชีวิตประจำวัน

## **Role and responsibility of each team member**

### **ເພື່ອພິຈານ ປີຍາວານນທ໌**

- ຮັບຜົດຂອບໃນສ່ວນກາຮອກແບບ System Architecture ແລະ ກາຮຕ່ອງຈາກ
- ປະກອບຕັວເຄື່ອງການອາກາສ
- Team management
- ຈັດທໍາຮາຍງານ

### **ວລັຍພຣຣນ ຕຣີທີພິຍໍສຸກຸລ**

- ຮັບຜົດຂອບໃນສ່ວນຂອງໂຄດທີ່ຮັບຂໍ້ມູນຈາກເຊັນເຊື່ອງ
- ທຳ Dashboard ເພື່ອແສດງຜລຄ່າຈາກເຊັນເຊື່ອງຕ່າງໆ
- ທຳ notification ຈາກ Blynk ແລະ ກາຮຈັດເກີບຂໍ້ມູນໃນ Firebase

### **ອີສສຍາພຣຣນ ລຶ່ມມ່ວງນິລ**

- ຮັບຜົດຂອບສ່ວນຂອງກາຮຮັບຄໍາສັ່ງເສີຍຈາກ Microphone ແລະ ກາຮນຳ AI ມາປະມວລຜລຄໍາສັ່ງ
- ທຳ Dashboard ເພື່ອແສດງຂໍ້ມູນຈາກເຊັນເຊື່ອງຕ່າງໆ

### **ສຸຈິນິນ ຮອດເປັນ**

- ຮັບຜົດຂອບໃນສ່ວນຂອງໂຄດທີ່ຮັບຂໍ້ມູນຈາກເຊັນເຊື່ອງ ແລະ ກາຮຄວບຄຸມ Relay
- ພັນນາໂຄດສໍາຮັບ Gateway ແລະ ກາຮຄ່າໄອນຂໍ້ມູນກັນຂອງບອർດ esp-32 ທີ່ສອງບອർດ

### **ໜຸດມ ທັກວິຫຼຽກຖະ**

- ຮັບຜົດຂອບໃນສ່ວນກາຮອກແບບ System Architecture ແລະ ກາຮຕ່ອງຈາກ
- ປະກອບຕັວເຄື່ອງການອາກາສ
- ຈັດທໍາຮາຍງານ ແລະ ຕັດຕ່ອງ Video documentation