

ระบบปรับปรุงสภาพอากาศภายในบ้านแบบพลวัต
โดยใช้เทคโนโลยีอินเทอร์เน็ตของสรรพสิ่ง

ภูมิพัฒน์ นิยมธรรม

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์และโทรคมนาคม
วิทยาลัยนวัตกรรมด้านเทคโนโลยีและวิศวกรรมศาสตร์
มหาวิทยาลัยธุรกิจบัณฑิตย์
ปีการศึกษา 2564

**AUTO ACTIVE AIRFLOW AND DYNAMIC AIR CONDITIONING
SYSTEMS USING INTERNET OF THINGS TECHNOLOGY**

PHUMIPAT NIYOMTHAM

**A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering
Department of Computer and Telecommunication Engineering
College of Innovative Technology And Engineering,
Dhurakij Pundit University
Academic Year 2021**



ใบรับรองวิทยานิพนธ์

วิทยาลัยนวัตกรรมการด้านเทคโนโลยีและวิศวกรรมศาสตร์ มหาวิทยาลัยราชภัฏบึงฉลวย
ปริญญา วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์และโทรคมนาคม

หัวข้อวิทยานิพนธ์ ระบบปรับปรุงสภาพอากาศภายในบ้านแบบพลวัต โดยใช้เทคโนโลยี
อินเทอร์เน็ตของสรรพสิ่ง
เสนอโดย นายภูมิพัฒน์ นิยมธรรม
สาขาวิชา วิศวกรรมคอมพิวเตอร์และโทรคมนาคม
อาจารย์ที่ปรึกษาวิทยานิพนธ์ ผู้ช่วยศาสตราจารย์ ดร.ณรงค์เดช กิรติพรานนท์

ได้พิจารณาเห็นชอบโดยคณะกรรมการสอบวิทยานิพนธ์แล้ว

.....ประธานกรรมการ
(รองศาสตราจารย์ ดร.สถัญจน วุฒิสัทติกุลกิจ)

.....กรรมการและอาจารย์ที่ปรึกษาวิทยานิพนธ์
(ผู้ช่วยศาสตราจารย์ ดร.ณรงค์เดช กิรติพรานนท์)

.....กรรมการ
(อาจารย์ ดร.ชัยพร เขมะภาคะพันธ์)

.....กรรมการ
(อาจารย์ ดร.รัฐศิลป์ รานอกกานูวัชร)

วิทยาลัยนวัตกรรมการด้านเทคโนโลยีและวิศวกรรมศาสตร์ รับรองแล้ว

.....คณบดีวิทยาลัยนวัตกรรมการด้านเทคโนโลยีและวิศวกรรมศาสตร์
(ดร.ชัยพร เขมะภาคะพันธ์)

วันที่ 30 เดือน ..ปี: ..พ.ศ. 2565

หัวข้อวิทยานิพนธ์	ระบบปรับปรุงสภาพอากาศภายในบ้านแบบพลวัต โดยใช้เทคโนโลยีอินเทอร์เน็ตของสรรพสิ่ง
ชื่อผู้เขียน	ภูมิพัฒน์ นิยมธรรม
อาจารย์ที่ปรึกษา	ผศ.ดร. ณรงค์เดช กิริติพรานนท์
สาขาวิชา	วิศวกรรมคอมพิวเตอร์และโทรคมนาคม
ปีการศึกษา	2564

บทคัดย่อ

การประมวลผลคลาวด์แบบฟัซซีลอจิก มีการศึกษาการจัดสรรทรัพยากรในระบบปรับปรุงสภาพอากาศภายในบ้านแบบพลวัต โดยใช้เทคโนโลยีอินเทอร์เน็ตของสรรพสิ่ง เป็นทรัพยากรหนึ่งที่สามารถนำมาจัดสรรได้ ดังนั้น ในวิทยานิพนธ์นี้ จึงประยุกต์วิธีการฟัซซีลอจิกเพื่อจัดสรร ควบคุมการทำงานของระบบพัดลมชนิดเดิม และ ระบายอากาศ ให้สอดคล้องกับสภาพอุณหภูมิของอากาศภายในอาคาร โดยมุ่งให้ผลลัพธ์ของการปรับปรุงสภาพอากาศ มีค่าการประหยัดพลังงานไฟฟ้า โดยเน้นเชื่อมต่อไปยังเครื่องปรับอากาศเพื่อให้ทำงานต่อเนื่อง เป็นการลดค่าใช้จ่ายพลังงานด้วยระบบอัจฉริยะแบบฟัซซีลอจิก

การทดลองจะประกอบไปด้วย 4 ส่วน เช่น ภาคอินพุต เอาท์พุท ระบบควบคุม และประมวลผลคลาวด์ โดยได้พัฒนาระบบจากต้นแบบจำลองก่อนนี้ มาเป็นแบบทดสอบภายในอาคาร โดยมีระบบควบคุมเป็นหัวใจสำคัญหลัก และ ระบบประมวลผลคลาวด์โดยใช้ Fuzzy Logic ที่คอยทำหน้าที่ประมวลผลสถานะอุณหภูมิของอากาศ เพื่อปรับสภาพอากาศให้เป็นไปตามวัตถุประสงค์ โดยมุ่งเน้นผลลัพธ์การประหยัดพลังงานไฟฟ้า ซึ่งคิดเป็นผลประหยัดพลังงานได้ที่ 36.9% เมื่อเทียบระหว่างระบบแบบฟัซซี (Fuzzy System) กับแบบธรรมดา (Normal System)

Thesis Title	AUTO ACTIVE AIRFLOW AND DYNAMIC AIR CONDITIONING SYSTEMS USING INTERNET OF THINGS TECHNOLOGY
Author	Phumipat Niyomtham
Thesis Advisor	Asst. Prof. Dr. Narongdech Keeratipranon
Department	Computer and Telecommunication Engineering
Academic Year	2021

ABSTRACT

Fuzzy Logic Cloud Computing resource allocation in a dynamic home climate improvement system was studied using the Internet of Things Technology. It is a resource that can be allocated. Therefore, we applied the fuzzy logic method to allocate and control the operation of the filling and ventilating fan system according to the temperature of the indoor air, focusing on the results of climate improvement. There is a value to save electricity, by focusing on connecting to the air conditioner to keep working. It reduces energy costs with intelligent fuzzy logic systems.

The experiment will consist of 4 parts, i.e. input, output, control system and cloud computing, which has developed a system from the original model before. Come as a test to use inside the real building. There is a control system as the heart and a cloud computing system using Fuzzy Logic to process the weather conditions. To adjust the weather according to the purpose by aiming to create results of saving electricity which accounted for an energy saving effect of 36.9% when compared to a fuzzy system with a normal system.

กิตติกรรมประกาศ

การจัดทำวิทยานิพนธ์ฉบับนี้ สำเร็จได้ด้วยความช่วยเหลือจากบุคคลหลายฝ่าย ผู้วิจัยขอขอบพระคุณ ผศ.ดร.ณรงค์เดช กิริติพรานนท์ และ ดร.รัฐศิลป์ รานอกภานุวัชร อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่สละเวลาอันมีค่า คอยให้ข้อคิดเห็น และคำแนะนำที่เป็นประโยชน์ต่องานวิจัย โดยสม่ำเสมอ และขอขอบคุณอาจารย์ ดร.ชัยพร เขมะภาคะพันธ์ ที่ให้ข้อคิดเห็น พร้อมให้คำชี้แนะต่างๆ ในการปรับปรุงแก้ไขข้อบกพร่องของวิทยานิพนธ์ฉบับนี้จนเสร็จสมบูรณ์ด้วยดี

ส่วนหนึ่งของความสำเร็จในครั้งนี้ก็มาจากเพื่อนๆ ที่คอยแนะนำ และเป็นกำลังใจ รวมถึงครอบครัวที่ให้โอกาส คอยให้การสนับสนุนและเป็นกำลังใจให้ตลอดระยะเวลาที่จัดทำวิทยานิพนธ์ฉบับนี้

ท้ายสุดนี้ประโยชน์และความดีของวิทยานิพนธ์ฉบับนี้ ขอมอบแด่ผู้มีพระคุณทุกๆ ท่าน ที่ได้ให้ความช่วยเหลือและเสริมสร้างกำลังใจให้ จนการจัดทำวิทยานิพนธ์ฉบับนี้สำเร็จลงได้ด้วยดี

ภูมิพัฒน์ นิยมธรรม



สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ฅ
บทคัดย่อภาษาอังกฤษ.....	ง
กิตติกรรมประกาศ.....	จ
สารบัญตาราง.....	ช
สารบัญภาพ.....	ฉ
บทที่	
1. บทนำ.....	1
1.1 ความสำคัญและที่มา.....	1
1.2 วัตถุประสงค์.....	3
1.3 ขอบเขตของงานวิจัย.....	3
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	4
1.5 ภาพรวมของระบบ.....	4
1.6 เครื่องมือที่ใช้ในการพัฒนา.....	6
1.7 องค์ความรู้สำคัญของงานวิจัย.....	6
1.8 ผลงานตีพิมพ์.....	7
2. ทฤษฎีและผลงานวิจัยที่เกี่ยวข้อง.....	8
2.1 ปัญหาฝุ่นละออง PM2.5 ในอากาศที่เกิดขึ้นรอบตัวเรา.....	8
2.2 เทคนิคการระบายอากาศ Ventilation Technical.....	9
2.3 งานวิจัยด้านการระบายอากาศภายในอาคาร.....	14
2.4 เทคนิคการใช้พัดลมเพื่อเติมอากาศ ช่องผ่านกรองฝุ่น และ ช่องระบายอากาศ...	16
2.5 เทคนิคการตรวจวัดด้วยเซ็นเซอร์และคัดกรองฝุ่น PM2.5 ด้วยฟิลเตอร์.....	19
2.6 องค์ประกอบของระบบ IoT.....	23
2.7 การนำระบบ Fuzzy Logic เข้ามาช่วยจัดการ.....	39
2.8 ระบบปรับปรุงสภาพอากาศภายในอาคารแบบพลวัต โดยใช้ IoT เข้ามาช่วย....	43
2.9 สรุปเนื้อหา.....	44

สารบัญ (ต่อ)

บทที่	หน้า
3. การวางแผนและการออกแบบ.....	45
3.1 แนวทางการวิจัยและพัฒนา.....	45
3.2 แผนการดำเนินงาน.....	46
3.3 ภาพรวมของระบบ.....	46
3.4 การออกแบบการทำงาน.....	48
3.5 สรุปเนื้อหา.....	58
4. ผลการทดสอบระบบ.....	59
4.1 สถานะการตรวจวัดด้วยเซ็นเซอร์วัดอุณหภูมิ.....	59
4.2 สถานะที่ควบคุมการหมุนใบพัดลม คุณ และระบายอากาศด้วยรอบสูง ปานกลาง ปกติ หยุดการทำงาน.....	63
4.3 ผลจากการจำลองการทำงาน พัดลมดูด-ระบายอากาศ ด้วยระบบควบคุมแบบฟัซซีลอจิก.....	67
4.4 ผลการทดสอบการควบคุมฟัซซีลอจิกและตัวควบคุมอินทิกรัลแบบปรับตัวได้.....	67
4.5 ผลการแสดงความสัมพันธ์ระหว่างรอบการหมุนของพัดลมและตัววัดอุณหภูมิ.....	70
4.6 ผลกราฟแสดงเปรียบเทียบระดับการประหยัดพลังงานของระบบ Fuzzy และ Normal.....	71
5. สรุปผลการทดลองและข้อเสนอแนะ.....	73
5.1 สรุปผลการวิจัย.....	73
5.2 ปัญหาที่พบและข้อเสนอแนะ.....	74
บรรณานุกรม.....	75
ภาคผนวก.....	79
ประวัติผู้เขียน.....	84

สารบัญตาราง

ตารางที่	หน้า
2.1 เปรียบเทียบข้อดีข้อเสียระหว่างระบบระบายอากาศแบบธรรมชาติ ระบบกล ระบายอากาศและระบบระบายอากาศแบบผสม.....	14
2.2 ตารางความสัมพันธ์ระหว่างค่าการแลกเปลี่ยนอากาศและในการกำจัดละออง ฝุ่น.....	16
3.1 แผนการดำเนินงาน.....	46
3.2 ตัวแปรที่ใช้ในการรับส่งข้อมูลระหว่าง Sensor Node และ Node MCU Firebase.....	52
3.3 ตัวแปรที่จัดเก็บใน EEPROM.....	54
3.4 แสดงค่าที่ Node MCU LINE รับมาจาก Sensor Node.....	55
4.1 แสดงข้อมูลสรุปการทดสอบหัวข้อที่ 4.1.....	63
4.2 ข้อมูลสรุปการทดสอบหัวข้อที่ 4.2.....	66
4.3 การเปรียบเทียบค่าความผิดพลาดระหว่างตัวควบคุมพีซีซีลอจิกและตัว ควบคุมพีซีซีลอจิกร่วมกับตัวควบคุมอินทิเกรตแบบปรับตัวได้.....	70

สารบัญภาพ

ภาพที่	หน้า
1.1 แผนผังแสดงภาพรวมของระบบ.....	5
2.1 การระบายอากาศแบบข้ามฝาก (Cross ventilation) จาก Automatic Natural by Ventilation Andrew Dyke 2019	10
2.2 การระบายอากาศทางเดียว (Single-sided ventilation).....	11
2.3 การระบายอากาศด้วยระบบกล (Mechanical ventilation).....	11
2.4 การเปรียบเทียบความแตกต่างระหว่างการระบายอากาศธรรมชาติและการระบายอากาศแบบผสม จาก Natural ventilation for infection control in Health-care setting โดย Martin Liddament 2018, World Health organization.....	13
2.5 วิธีการระบายอากาศแบบธรรมชาติและวิธีกล จากปัญหาาระบบระบายอากาศภายในอาคาร และแนวทางแก้ไข โดย อนุภพละอง 2560 นครสวรรค์:ศูนย์วิศวกรรมทางการแพทย์ที่ 3.....	15
2.6 ภาพรวมทั้งหมดของระบบระบายอากาศและการกรองฝุ่นละอองภายในอาคาร	17
2.7 วิธีการติดตั้ง ช่องเติมอากาศด้วยพัดลมติดผนัง เพื่อช่วยเติมอากาศเข้าภายในอาคาร.....	17
2.8 ลักษณะการติดช่องเติมอากาศติดผนัง ให้เห็นทั้งภายในและภายนอกอาคาร.....	18
2.9 ลักษณะการติดช่องเติมอากาศติดผนัง ให้เห็นทั้งภายในและภายนอกอาคาร.....	18
2.10 ชุดพัดระบายความร้อนห้องโถงหลังคา ช่วยให้เกิดการไหลเวียน และ อากาศมีการไหลเวียนอยู่ตลอดเวลา.....	19
2.11 ผลของการวัดปริมาณฝุ่นวันที่ 22 มกราคม 2019 บริเวณที่พักอาศัยโดยผ่าน App AirVisual.....	20
2.12 การทำงานใน 4 ขั้นตอนของเครื่องตรวจวัดปริมาณฝุ่น จากยี่ห้อ Honeywell HPRM Series.....	21
2.13 การทำงานของแผ่นกรอง HEPA Filters จาก Source : Dyson.com.....	22
2.14 Network Layers ของ Internet of Things โดย Console Firebase Google Cloud Website Service Platform.....	23
2.15 อุปกรณ์ Microcontroller Arduino Uno Nano.....	24

สารบัญภาพ (ต่อ)

ภาพที่	หน้า
2.16 เซ็นเซอร์ตรวจวัดอุณหภูมิ DHT11 Temperature And Humidity Sensors.....	25
2.17 การทำงานของ DHT11.....	25
2.18 อุปกรณ์เซ็นเซอร์ตรวจวัดปริมาณฝุ่น Plan tower Laser PM 2.5 Dust Sensor PMS7003.....	27
2.19 วงจรการทำงานของ Relay Switch 4 Channel.....	27
2.20 แสดง MQTT on Local Network Gateway.....	28
2.21 อุปกรณ์ Microcontroller NodeMCU V2 ESP8266.....	29
2.22 โครงสร้างของ Console Firebase Google Cloud Messaging.....	30
2.23 โครงสร้างของ Console Firebase Google Cloud Functions.....	31
2.24 การทำงานและองค์ประกอบของ Console Firebase Google Cloud Website Service IoT.....	31
2.25 การทำงานและองค์ประกอบของ Console Firebase Google Cloud Website Service IoT.....	32
2.26 โครงสร้างการทำหน้าที่ Back-End และ Front-End ของ Console Firebase Cloud Website Service.....	33
2.27 ตารางระบุความสามารถที่ทำงานแบบ Offline (On-device) และ Online (Cloud).....	34
2.28 รูปแบบของการผิดพลาด.....	35
2.29 รูปแบบของการรายงานประสิทธิภาพ.....	35
2.30 การทดสอบอุปกรณ์ต่างๆ.....	36
2.31 รูปแบบผลการทดสอบ.....	36
2.32 การแสดงผล In-App Messaging.....	37
2.33 การเชื่อมต่อพัลลัม กับ Arduino Uno R3.....	39
2.34 การรับส่งข้อมูลแบบ I2C BUS.....	39
2.35 แสดงบูลีนลอจิก และ ฟังก์ชันลอจิก.....	40
2.36 ตัวอย่างอุณหภูมิปานกลางของเซตวินัยและเซตของพีซีซี.....	41

สารบัญภาพ (ต่อ)

ภาพที่	หน้า
2.37 การกำหนดค่าความเป็นสมาชิกของเซตทวินัยและพีชชีเซต.....	42
3.1 แผนผังแสดงภาพรวมของระบบ.....	46
3.2 ภาพรวมของระบบ Firebase on IoT.....	48
3.3 ผังการทำงานของระบบ Sensor Node.....	49
3.4 ผังการทำงานของระบบ Gateway Firebase.....	50
3.5 ผังการทำงานของระบบเปิดปิดพัดลมระบายอากาศ.....	50
3.6 ผังการทำงานของระบบดูดอากาศ.....	51
3.7 การเชื่อมต่อ Arduino Nano กับ Node MCU.....	52
3.8 การทำงานของ Google Firebase Cloud ในส่วนของ IoT.....	56
3.9 ส่วนโครงสร้าง Realtime Database ใน Storage.....	56
3.10 การกำหนดค่าอุณหภูมิโดยใช้ทฤษฎีของพีชชีเซต.....	57
3.11 การกำหนดรอบความเร็วของพัดลมและอุณหภูมิโดยใช้ทฤษฎีของพีชชีเซต	58
4.1 รายงานผลกราฟตามเวลาจริง แสดงอุณหภูมิสูง และ ความชื้นผ่าน HTT:Thing Speak.....	60
4.2 รายงานผลกราฟตามเวลาจริง แสดงอุณหภูมิต่ำปานกลาง และ ความชื้นผ่าน HTT:Thing Speak.....	60
4.3 รายงานผลกราฟตามเวลาจริง แสดงอุณหภูมิปกติ และ ความชื้น ผ่าน HTT:Thing Speak.....	60
4.4 รายงานผลกราฟตามเวลาจริง แสดงอุณหภูมิต่ำกว่า 25 องศาเซลเซียส ผ่าน HTT:Thing Speak.....	61
4.5 รายงานผลแสดงอุณหภูมิ และ ความชื้น ภายในอาคาร เพื่อเปรียบเทียบและ ยืนยันผ่านสมาร์ตโฟน โดยใช้ MQTT : Blynk Application.....	62
4.6 กราฟแสดงรอบการหมุนพัดลม ดูด และระบายอากาศ รอบสูง 6,000 รอบ ต่อนาที.....	64
4.7 กราฟแสดงรอบการหมุนพัดลม ดูด และระบายอากาศ รอบสูง 5,000 รอบ ต่อนาที.....	64

สารบัญภาพ (ต่อ)

ภาพที่	หน้า
4.8 กราฟแสดงรอบการหมุนพัดลม ดูด และระบายอากาศ รอบสูง 4,000 รอบ ต่อนาที.....	64
4.9 กราฟแสดงรอบการหมุนพัดลม ดูด และระบายอากาศ รอบเป็นศูนย์.....	65
4.10 กราฟแสดงช่วงเวลาและรอบการทำงานของพัดลมดูดอากาศ (Fresh Air) และระบายอากาศ (Exhaust Air).....	67
4.11 ผลการทดสอบระบบด้วยตัวควบคุมพีซีซีลอจิก.....	68
4.12 ผลการทดสอบระบบด้วยตัวควบคุมพีซีซีลอจิกเมื่อระบบถูกรบกวน.....	68
4.13 ผลการทดสอบระบบด้วยตัวควบคุมพีซีซีลอจิกร่วมกับตัวควบคุมอินทิกรัล แบบปรับตัวได้.....	69
4.14 กราฟประมวลผลรวม fuzzy logic แสดงความสัมพันธ์ของอุณหภูมิที่ลดลง และความเร็วรอบของพัดลมที่ช้าลง.....	71
4.15 กราฟเปรียบเทียบระดับการประหยัดพลังงานระหว่างระบบ Fuzzy และ Normal.....	72

บทที่ 1

บทนำ

1.1 ความสำคัญและที่มา

จากสถิติในปัจจุบันพบว่าประเทศไทยติดหนึ่งในสิบอันดับต้นๆ ของโลก ของสภาพภูมิอากาศที่ร้อนและการเกิดฝุ่นละอองมากขึ้น เนื่องจากภูมิประเทศอยู่ในบริเวณใกล้เส้นศูนย์สูตร โดยสภาพอากาศในฤดูร้อนช่วงเดือนเมษายนถึงเดือนกรกฎาคมของทุกๆ ปี จะมีอุณหภูมิสูงไปถึง 40 องศาเซลเซียส โดยเฉพาะในภาคกลาง ภาคเหนือ และภาคตะวันออกเฉียงเหนือ ส่วนภาคใต้ ภาคตะวันออกและภาคตะวันตก อุณหภูมิจะแปรสภาพสูงสุดไม่เกิน 38 องศาเซลเซียส ในช่วงของเวลากลางวัน อาจเนื่องมาจากประเทศไทยอยู่ใกล้บริเวณทะเลฝั่งอ่าวไทย จึงได้รับอิทธิพลลมมรสุม ลมทะเลพัดผ่าน เป็นเหตุให้มีการถ่ายเทความร้อนอย่างต่อเนื่องของในแต่ละวัน จึงเป็นผลดีแต่ไม่มากนัก ของการปรับสภาพอากาศ โดยร้อยละ 80 จะพบว่าอากาศจะร้อนทั้งปี และจะมีร้อนขึ้นในฤดูฝนและท้ายสุดในฤดูหนาว ซึ่งจะมีระยะเวลาของอากาศหนาว โดยประมาณไม่เกินสองเดือนของในทุก ๆ ปี จึงเป็นเหตุให้พบว่าร้อยละ 80 ของสภาพภูมิอากาศร้อนจะยาวนานมากกว่าทุกฤดู และฝุ่นละอองที่เกิดขึ้นไม่เว้นแต่ละวัน อันเนื่องมาจากการเผาป่า ป่าถูกเผา การก่อสร้าง สิ่งปลูกสร้าง ซึ่งส่งผลต่อสภาพร่างกาย อารมณ์และจิตใจ ส่งผลให้มีค่าใช้จ่ายที่จะต้องจ่ายเพิ่มขึ้น โดยเฉพาะค่ากระแสไฟฟ้าที่ส่งผลโดยตรง โดยเฉพาะอย่างยิ่งเครื่องปรับอากาศและเครื่องฟอกอากาศที่ต้องเปิดใช้ในระยะเวลาที่นานขึ้น

ปัจจุบันความร้อนและฝุ่นละอองที่สะสมไม่เพียงในสภาพภูมิอากาศ แต่ยังส่งผลในตัวอาคาร บ้านเรือน สำนักงาน สถานที่ราชการ โรงงานอุตสาหกรรม ซึ่งได้รับผลกระทบโดยตรงจากความร้อนแสงอาทิตย์ ความร้อนจะค่อย ๆ สะสมเข้าไปภายในตัวอาคารจากอุณหภูมิ 32-35 องศาเซลเซียส และจะสะสมความร้อนสูงสุดหรือที่เรียกว่า อุณหภูมิเพดานสูงสุดได้ถึง 50 องศาเซลเซียส ในกรณีอาคารเหล่านั้นไม่มีช่องระบายอากาศหมุนเวียน ซึ่งถือเป็นวัฒนธรรมของคนเอเชีย ที่ต้องปิดอาคารบ้านเรือนให้มีฉนวนเพื่อป้องกันการโจรกรรมทรัพย์สินที่อาจจะเกิดขึ้นได้ตลอดเวลา และนั่นจึงเป็นเหตุให้เกิดการสะสมของอุณหภูมิภายในอาคาร และยังเป็นแหล่งสะสมแบคทีเรีย ที่ส่งผลต่อสุขภาพร่างกาย แต่ในขณะเดียวกัน เมื่อผู้อาศัยได้กลับจากภารกิจประจำวัน และเข้าพักอาศัยพบว่า ร้อยละ 80 ต้องใช้ความเย็นจากแอร์คอนดิชัน ซึ่งจะต้องใช้พลังงานสูงมากในการเปิดและทำ

ให้อากาศเย็น โดยพบว่ากว่าจะเย็นจำเป็นต้องใช้เวลาไม่ต่ำกว่า 1-2 ชม. ในการทำความเย็นเนื่องจาก ความร้อนที่สะสม เป็นการเพิ่มภาระค่าใช้จ่ายของกระแสไฟฟ้าที่สูงมาก ในช่วงของการปรับค่า FT จากทางการไฟฟ้าส่วนภูมิภาคค่า FT (Float Time) หมายถึง การปล่อยลอยตัวค่าของต้นทุนการผลิต เหลือพลังงานเพื่อนำไปใช้ในส่วนอื่นๆ ได้อีก ต้นทุนที่ไม่สามารถควบคุมได้ เช่น ราคาเชื้อเพลิง อัตราเงินเฟ้อ อัตราแลกเปลี่ยนเงินตราต่างประเทศ เป็นต้น ซึ่งปัจจัยเหล่านี้จะถูกใช้เป็นการรอบ ในการคำนวณ

ดังนั้นจึงมีงานวิจัยและเทคโนโลยีมากมายเกิดขึ้น เพื่อนำมาช่วยแก้ไขปัญหา โดยหยิบยกเอาตัวอย่างงานวิจัยของระบบ Active AIR flow TM System จากบริษัท SCG ที่เป็นนวัตกรรมระบบระบายอากาศภายในบ้าน โดยมีเงื่อนไขการติดตั้งเฉพาะบ้านเดี่ยวชนิดสองชั้นเท่านั้น ซึ่งจะมุ่งเน้นเฉพาะบ้านหลังใหม่ ซึ่งง่ายต่อการติดตั้งและจัดการกับระบบหรือนวัตกรรมบ้านเย็น โดยกำหนดการใช้งานกับวัสดุพิเศษเท่านั้น ซึ่งเป็นฉนวนกันความร้อนชนิดบุผนังที่เรียกว่า Q-CON และชนิดบุหลังคา ที่เรียกว่า STAY COOL โดยทั้งหมดเน้นการป้องกันความร้อนจากนอกบ้านเข้าสู่ภายในบ้าน เนื่องจากผลกระทบจากแสงยูวีของ ดวงอาทิตย์โดยเพิ่มการเร่งระบายอากาศตรงบริเวณห้อง โถงหลังคาด้วยการติดตั้งระบบพัดลมดูดอากาศออกจากตัวอาคารตรงจุดด้านบนของหลังคา มุ่งเน้นการถ่ายเทความร้อนออกจากตัวบ้านผ่านโถงของหลังคา โดยมีขีดความสามารถในการทำงานได้ตั้งแต่ 7-10 รอบต่อชั่วโมง ซึ่งจะช่วยให้บ้านไม่ร้อนอบอ้าว เนื่องจากมีการระบายอากาศร้อนสะสมออกสู่ภายนอกอยู่ตลอดเวลา จึงช่วยประหยัดค่าไฟและเพิ่มประสิทธิภาพการทำงานของเครื่องปรับอากาศ ทำให้เย็นเร็วขึ้น แต่ข้อจำกัดในโครงการนี้ยังมีอุปสรรคในเรื่องราคาจำหน่าย ซึ่งส่งผลในแง่ของความคุ้มค่ากับการลงทุนที่สูง โดยราคาเริ่มต้นสำหรับรุ่น SMART ราคาจำหน่ายที่ 72,000บาท วัสดุพ่นสีแบบสีรถยนต์ และไม่มีระบบการเชื่อมต่อกับ Application ใน Smartphone หรือ iPad แต่จะมี Display ขนาด 2 นิ้ว ติดอยู่บริเวณใกล้ๆกล่อง Control Box ของระบบและ Monitor Sensor สามารถจับอุณหภูมิได้เพียง 1 จุด คือบริเวณโถงหลังคาบ้าน สำหรับรุ่น PREMIUM ราคาจำหน่ายที่ 108,000บาท โดยวัสดุจะเน้นเป็นสแตนเลส ระบบมีการเชื่อมต่อกับ Application ใน Smartphone หรือ iPad และ Monitor Sensor สามารถจับอุณหภูมิได้ 3 จุด ขึ้นอยู่กับสภาพแวดล้อมและขนาดพื้นที่ โดยทั้งสองรุ่นยังไม่รวมค่าติดตั้งและค่าสำรวจก่อนก่อนติดตั้ง ซึ่งจากการศึกษาค้นคว้าพบว่า มีความเป็นไปได้ส่วนน้อยมากที่จะนำระบบนี้มาใช้งานจริงๆ เนื่องจากมีข้อจำกัดในแง่ของการลงทุนที่สูง และพื้นที่กับสภาพแวดล้อมของการติดตั้ง โดยกำหนดเพียงบ้านเดี่ยวสองชั้นเท่านั้น พร้อมกำหนดเงื่อนไขให้ใช้วัสดุที่เป็นกระเบื้องหลังคา ฉนวนกันความร้อนบุผนัง บุษหลังคา จากทางบริษัทเพื่อรักษามาตรฐานในการติดตั้งทั้งระบบ ซึ่งส่งผลให้เกิดอุปสรรคต่อความต้องการใช้งาน

ดังนั้นงานวิจัยนี้จึงได้นำเอาเทคโนโลยีที่เรียบง่ายของ IoT หรือที่เรียกว่า Internet of Things เข้ามาช่วยแก้ไขปัญหาและความต้องการจัดการกับระบบไหลเวียนอากาศด้วยพัดลมอัตโนมัติ The Auto Active Airflow-Fan Cooler System using Internet of Things Technology ที่ช่วยเตรียมอุณหภูมิภายในบ้านให้เย็นลงต่ำกว่า 32 องศาเซลเซียส พร้อมประสิทธิภาพช่วยขจัดฝุ่นละออง PM 2.5 ให้มีจำนวนน้อยลงตามค่ามาตรฐานฝุ่น PM 2.5 เพื่ออากาศที่สะอาดก่อนผู้อาศัยจะเข้าสู่ภายในบ้าน ส่งผลดีต่อการใช้แอร์คอนดิชันไม่ให้ทำงานหนัก และลดจำนวนการใช้งานอุปกรณ์ไฟฟ้าต่างๆที่ไม่จำเป็น และค่าไฟฟ้าให้น้อยลง ด้วยต้นทุนที่เหมาะสมและการใช้งานง่าย สะดวกต่อผู้ใช้งานด้วยการควบคุมผ่านแอปพลิเคชันบนสมาร์ตโฟน

1.2 วัตถุประสงค์

1.2.1 เพื่อพัฒนาระบบไหลเวียนอากาศด้วยพัดลมอัตโนมัติโดยการประยุกต์ใช้งานจากเทคโนโลยีของอินเทอร์เน็ตเนตเวิร์ก

1.2.2 พัฒนาระบบตรวจจับและคัดกรองฝุ่น PM2.5 โดยการประยุกต์ใช้งานจากเทคโนโลยีของอินเทอร์เน็ตเนตเวิร์ก

1.2.3 พัฒนาระบบควบคุมแบบฟัซซี่ โดยใช้ Fuzzy logic เพื่อให้เกิดการประหยัดพลังงานในการควบคุมสภาพแวดล้อมของห้อง

1.2.4 พัฒนาส่วนการเชื่อมต่อ Gateway เพื่อจัดส่งค่าวัดอุณหภูมิที่วัดได้ผ่านระบบเครือข่ายที่ใช้โปรโตคอลแตกต่างกัน

1.2.5 พัฒนาส่วนแสดงผลให้กับผู้ใช้งานผ่านแอปพลิเคชันบนสมาร์ตโฟน

1.3 ขอบเขตของงานวิจัย

1.3.1 ใช้อินเทอร์เน็ต WiFi เป็นตัวเชื่อมต่อสู่ระบบคลาวด์ของ console.firebase.google.com

1.3.2 ใช้บอร์ด Arduino ไมโครคอนโทรลเลอร์เป็นตัวกลางในการควบคุมและประมวลผลระบบอัตโนมัติในรูปแบบฮาร์ดแวร์และซอฟต์แวร์

1.3.3 พัฒนาส่วนควบคุมเซนเซอร์โหนด ได้แก่ส่วนอิเล็กทรอนิกส์ เซนเซอร์วัดอุณหภูมิ และวัดขนาดฝุ่น PM 2.5 เพื่อคัดกรองโดยใช้ไมโครคอนโทรลเลอร์เพื่อประมวลผล

1.3.4 เซนเซอร์วัดอุณหภูมิจำนวนสองตัว โดยติดตั้งบนชั้นที่ 1 และชั้นที่ 2 ภายในบ้าน

1.3.5 เซนเซอร์ตรวจวัดฝุ่น PM2.5 บริเวณพิวเตอร์กรองอากาศสองจุด ด้านใน ด้านนอกอาคาร

1.3.6 พัดลมระบายอากาศออกสู่ภายนอกตัวบ้านจำนวน 1 ตัว ติดตั้งบริเวณชั้นที่ 2

1.3.7 พัดลมดูดอากาศเข้าภายในบ้านจำนวน 1 ตัว ติดตั้งบริเวณชั้นที่ 1

1.3.8 พิลเตอร์กรองอากาศ แบบป้องกันและคัดกรองฝุ่น PM 2.5 ก่อนนำเข้าภายในอาคารชั้นที่ 1

1.3.9 ช่วงเวลาอ่านค่าเริ่มต้นการทำงานของระบบที่เวลา 03:00 pm – 12:00 am

1.3.10 พัฒนาเกตเวย์เครือข่ายที่ใช้โปรโตคอลต่างกันและส่งข้อมูลต่างกันได้สื่อสารกันได้

1.3.11 ให้สามารถแสดงข้อมูลการทำงานของระบบโดยรวมได้

1.3.12 สามารถนำข้อมูลมาวิเคราะห์และวางแผนในขั้นตอนต่อไปได้

1.3.13 ให้สามารถสั่งการและควบคุมระบบในรูปแบบแมนวลผ่านแอปพลิเคชัน

1.3.14 ให้สามารถสั่งการและควบคุมระบบได้เองในรูปแบบอัตโนมัติ

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 ผู้ใช้สามารถสังเกตการณ์ในระยะไกล และระยะใกล้ สามารถควบคุมปรับลดอุณหภูมิภายในอาคารเตรียมอุณหภูมิที่เหมาะสมก่อนเข้าพักอาศัย

1.4.2 การถ่ายเทความร้อนสู่ภายนอกนำพาอากาศเย็นสู่ภายใน ช่วยลดการสะสมของฝุ่น PM 2.5 ในอากาศ ส่งผลให้คุณภาพอากาศดีขึ้น

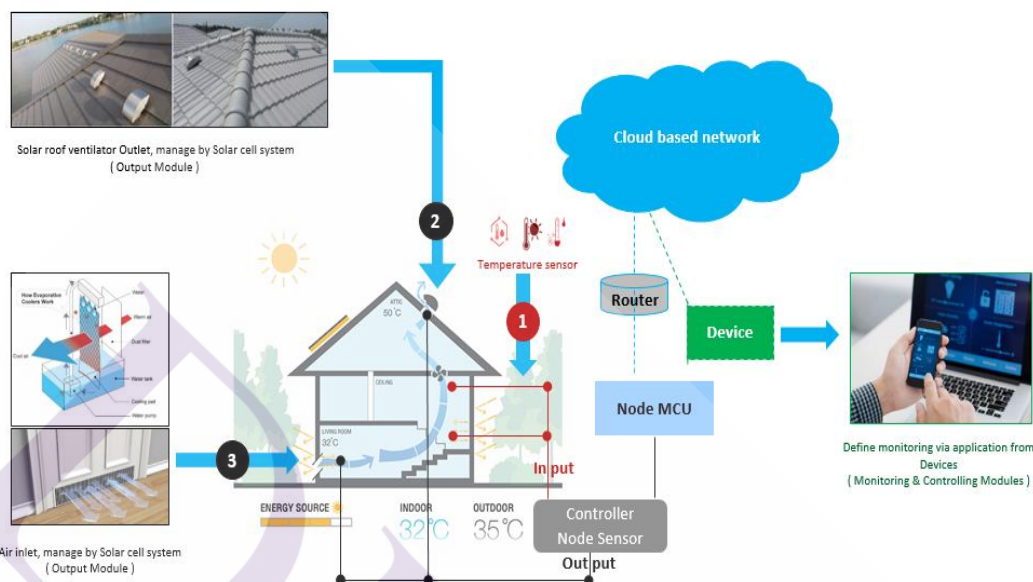
1.4.3 ลดภาระค่าใช้จ่ายของการใช้กระแสไฟฟ้าและเพื่อช่วยประหยัดค่าไฟฟ้าในแต่ละช่วงของการใช้งาน โดยเฉพาะช่วงสภาพอากาศร้อนในช่วงฤดูร้อนโดยมีผลเกี่ยวข้องกับค่าต้นทุนแปรผัน FT (Float Time) จากทางการไฟฟ้าส่วนภูมิภาค

1.4.4 สามารถนำระบบที่ออกแบบไปใช้ในการควบคุมระบบไหลเวียนอย่างอื่น เช่น ระบบไหลเวียนของไหลในท่อ เช่น ในอุตสาหกรรมพลังงาน เชื้อเพลิง น้ำมัน แก๊ส ฯลฯ หรืออุตสาหกรรมยานยนต์ในการไหลเวียนของอากาศในรถยนต์ และ อุตสาหกรรมลอจิสติกส์ โดยการขนถ่ายวัสดุหรือสินค้าลงเรือ รถบรรทุกตู้คอนเทนเนอร์ และการลำเลียงสินค้าจากจุดหนึ่งไปยังจุดหนึ่งที่ต้องการควบคุมอุณหภูมิ และควบคุมปริมาณฝุ่นหรือสารปนเปื้อนต่างๆ ด้วยอุปกรณ์ที่ทันสมัยขึ้น

1.5 ภาพรวมของระบบ

จะมีส่วนของการใช้เว็บแอปพลิเคชัน เพื่อใช้ควบคุมและสั่งการระบบ ผ่านเว็บเบราว์เซอร์ ส่วนอุปกรณ์ Sensor Node จะทำหน้าที่รับและส่งค่าหรือ Message จากตัวเซนเซอร์ต่างๆ ให้ส่งไปยังส่วนGateway หรือเรียกว่าเส้นทางที่กำหนดขึ้นและส่งต่อไปยังส่วนระบบ

ฐานข้อมูล เพื่อดำเนินการจัดเก็บข้อมูลที่ซับซ้อนของระบบ โดยภาพรวมของระบบทั้งหมดสามารถแบ่งส่วนการทำงานออกเป็นภาคส่วนดังนี้ ดังรูปที่ 1



ภาพที่ 1.1 แผนผังแสดงภาพรวมของระบบ

1.5.1 Sensor Node คืออุปกรณ์ตัวหลักคอยทำหน้าที่ป้อนภาคขยายส่วน Input และ Output โดยรับค่าจากอุปกรณ์ที่เป็นเซนเซอร์ต่างๆ ในการส่งข้อมูลและแสดงผลของข้อมูลผ่าน Gateway โดย ON-OFF Switch Relay สำหรับควบคุมสัญญาณไปยังภาค Input คือ Temperature Sensor และภาค Output

1.5.2 Gateway ทำหน้าที่รับและส่งข้อมูลจาก Sensor Node และ Firebase Cloud ผ่านโปรโตคอล MQTT ซึ่งถือเป็นตัวหลักมาตรฐานในการใช้งาน และสามารถใช้ได้ทั่วไปกับอุปกรณ์ต่างๆ

1.5.3 Firebase Cloud Computing ทำหน้าที่เป็น Web Server เพื่อจัดการระบบการทำงานของเทคโนโลยี Internet of Things เพื่อรับและส่งข้อมูลจาก Gateway ผ่านโปรโตคอล MQTT และใช้ Firebase Dynamo-DB สำหรับเป็นฐานข้อมูลในการเก็บข้อมูล โดยจะถูกจัดเก็บในรูปแบบไฟล์ JSON

1.5.4 User Interface เป็นลักษณะ Web Application หมายถึง ส่วนแสดงผลและควบคุมการทำงานของระบบ โดยสามารถสั่งการผ่านแอปพลิเคชันบนสมาร์ตโฟน

1.6 เครื่องมือที่ใช้ในการพัฒนา

1.6.1 ส่วนของฮาร์ดแวร์ (Hardware)

1.6.1.1 ARDUINO UNO R3 Microcontroller ที่ทำหน้าที่ควบคุมหน่วยประมวลผล
กลาง

1.6.1.2 CNDF DC12VDC 0.28A TFS-9225G โมดูลพัดลม 2 ชุด ที่ทำหน้าที่เป็น
พัดลมดูด และระบายอากาศ ภาคเอาต์พุต

1.6.1.3 HC-020 Opto switch, Phototransistor ชุดตรวจจับแสงเซนเซอร์วัดรอบมอเตอร์

1.6.1.4 Motor Driver L298N 2A โมดูลขับมอเตอร์พัดลมดูดและระบายอากาศ

1.6.1.5 NODE MCU ESP32 Tutorial ที่ทำหน้าที่เป็น GATEWAY

1.6.1.6 NRF24L01 โมดูลสื่อสารไร้สายเป็นตัวรับและส่งสัญญาณ

1.6.1.7 DTH11 Temperature And Humidity Sensors ทำหน้าที่ตรวจวัดอุณหภูมิ
ภาคอินพุต

1.6.1.8 Battery Recharge 12V DC 35A สนับสนุนระบบมอเตอร์แบบใบพัดลม

1.6.1.9 PM2.5 เซ็นเซอร์ตรวจจับฝุ่นละออง วัดคุณภาพอากาศ Laser Dust Sensor
PMS3003

1.6.1.10 Filter กรองอากาศสำหรับดักจับฝุ่น PM2.5 ก่อนนำอากาศเข้าสู่ภายในอาคาร

1.6.2 ส่วนของซอฟต์แวร์ (Software)

1.6.2.1 Arduino IDE

1.6.2.2 Android Studio

1.6.2.3 Visual Studio Code

1.6.2.4 Xcode

1.6.2.5 Console Firebase Cloud Google Website Service

1.7 องค์กรความรู้สำคัญของงานวิจัย

นำเสนอแนวทางการประยุกต์ใช้เทคโนโลยีของระบบอินเทอร์เน็ตสรรพสิ่ง IoT (Internet Of Things) เพื่อรองรับเทคโนโลยีอุตสาหกรรมไทยแลนด์ 4.0 เรียนรู้และเข้าใจหลักการ
ทำงานของอุปกรณ์ต่างๆในงานวิจัย

การทดสอบขั้นต้นได้ทำการทดสอบระบบกับบ้านจำลองขนาดสองชั้นพื้นที่กว้าง 1 ฟุต
ยาว 1 ฟุต และสูง 2 ฟุต ที่ทำจากวัสดุจาก PC (Polycarbonate) โดยการแยกการทดสอบออกเป็น
ส่วนๆก่อนตามลำดับ นำชุดขับเคลื่อนมอเตอร์แบบพัดลมดูดและระบายอากาศมาต่อเข้ากับชุด

ไมโครคอนโทรลเลอร์เพื่อทำการทดสอบการทำงานของพัดลมทั้งสองในเรื่องของทิศทางการหมุนของพัดลม และตรวจสอบการรับค่าจาก เซนเซอร์ของมอเตอร์ที่อยู่กับพัดลมว่าใช้งานได้เป็นปกติหรือไม่ พร้อมระบบตรวจวัดค่าฝุ่น PM 2.5

การพัฒนาขั้นที่สอง ได้นำมาประยุกต์ใช้โดยเชื่อมต่อระบบ เข้าภายในอาคาร 2 ชั้น ขนาด 15 ตรม. เพื่อศึกษาข้อมูลที่เซนเซอร์ แต่ละชนิดให้ส่งค่าออกมาเพื่อที่จะเตรียมข้อมูลนำไปใช้ในการประมวลผล บนแดชบอร์ด เพื่อให้ผู้ใช้งานทราบถึงประสิทธิภาพการทำงานและเข้าใจหลักการของอุปกรณ์ต่าง ๆ ที่นำมาใช้ในงานวิจัย

1.8 ผลงานตีพิมพ์

ส่วนหนึ่งของวิทยานิพนธ์นี้ ได้รับการตีพิมพ์เป็นบทความวิจัยเรื่อง “ Auto Active Airflow-Fan Cooler System Using Internet of Things Technology” Phumipat Niyomtham, Narongdech Keeratipranon and Ratthaslip Ranokphanuwat 2017 2nd International Conference on Economics & Management Innovations (ICEMI) 1(1) (2017) 367-369, 15-16, July, 2017 Bangkok Thailand.

อากาศนำไปสู่ความต้องการที่เกินอุปทานในหน้ากากป้องกันใบหน้า N-95 ในขณะที่จำนวนโรงพยาบาลเข้าร่วมด้วยเนื่องจากความเจ็บป่วยทางเดินหายใจเพิ่มขึ้นอย่างมีนัยสำคัญ

เนื่องจากการขาดแคลนข้อมูลที่เกี่ยวข้องกับประชาชนทำให้จุฬาลงกรณ์มหาวิทยาลัยกำลังดำเนินการเพื่อแบ่งปันความรู้และเพิ่มความตระหนักในการจัดการกับสถานการณ์ของ PM2.5 ซึ่งจะเป็นประโยชน์ต่อประชากรในประเทศไทยในการปกป้องสุขภาพของพวกเขา เช่นเดียวกับการสนับสนุน อภิปรายเกี่ยวกับวิธีการอย่างมีประสิทธิภาพแก้ไขปัญหาด้วยคู่มือ วิธีการนี้เป็นความพยายามร่วมกันพัฒนาขึ้น ด้วยการสนับสนุนจากอาจารย์ผู้เชี่ยวชาญจากคณะต่างๆ และสาขาที่จะใช้ความรู้ความเชี่ยวชาญจากคณะต่างๆ และสาขาที่จะใช้ความรู้ความชำนาญ และการวิจัยเชิงวิชาการ เพื่อให้การศึกษาเพื่อส่งต่อไปยังประชาชน โดยเฉพาะด้านสุขภาพเป็นเป้าหมายหลัก

2.2 เทคนิคการระบายอากาศ Ventilation Technical

การระบายอากาศ เป็นการเคลื่อนที่ของอากาศภายนอกเข้ามาในพื้นที่อาคารหรือพื้นที่ห้อง และเข้าแทนที่อากาศภายในอาคารหรือห้อง ซึ่งจุดประสงค์หลักของการระบายอากาศในอาคารคือ เพื่อให้อากาศที่ดีมีพอสำหรับการหายใจ พร้อมทั้งเจือจางมลพิษที่เกิดขึ้นภายในพื้นที่และการกำจัดมลพิษออกจากอาคาร โดยการระบายอากาศในอาคารประกอบด้วย 3 องค์ประกอบพื้นฐาน ดังนี้

(1) อัตราการระบายอากาศ (Ventilation rate) คือ ปริมาตรอากาศภายนอกที่ไหลเข้าสู่พื้นที่ และเป็นอากาศที่มีคุณภาพ

(2) ทิศทางการไหลของอากาศ (Airflow direction) คือ การคำนึงถึงทิศทางการไหลของอากาศโดยรวมภายในอาคาร โดยอากาศควรจะไหลจากพื้นที่ที่สะอาดไปยังพื้นที่ที่สกปรก

(3) การกระจายตัวของอากาศ (Air distribution) คือ อากาศภายนอกควรเข้าถึงทุกพื้นที่และระบายเอามลพิษในอากาศออกไปอย่างมีประสิทธิภาพ

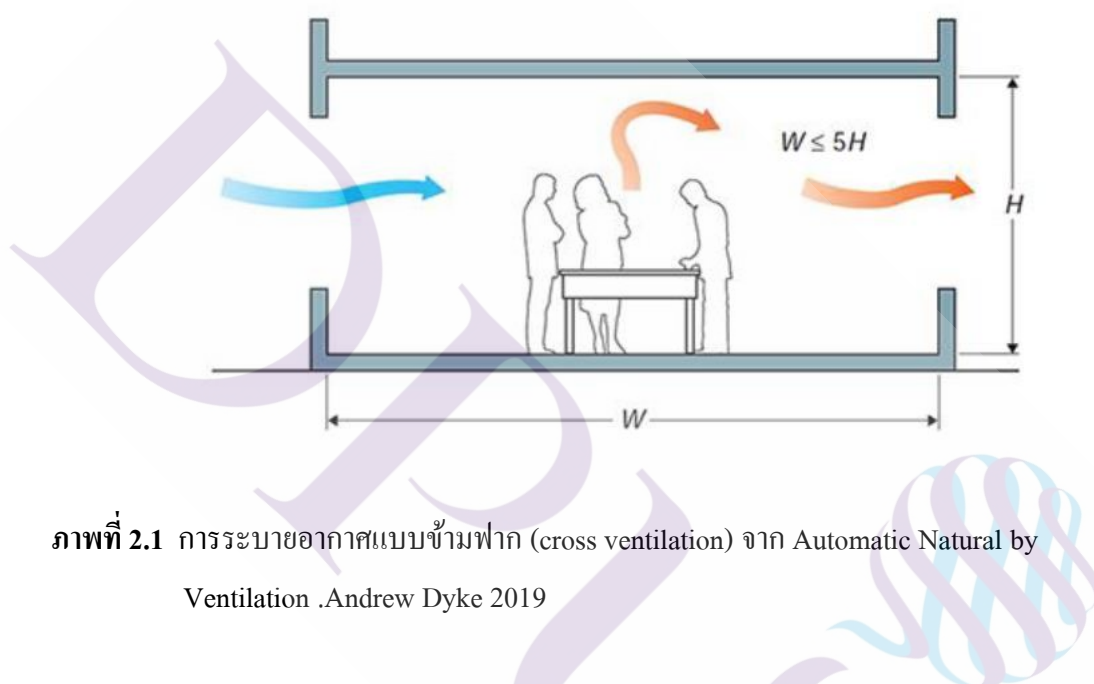
โดยการระบายอากาศภายในอาคารแบ่งออกเป็น 3 ประเภทได้แก่ การระบายอากาศด้วยธรรมชาติ การระบายอากาศด้วยระบบกลและการระบายอากาศแบบผสม

2.2.1 การระบายอากาศด้วยธรรมชาติ (Natural ventilation)

การระบายอากาศธรรมชาติ เป็นการระบายอากาศที่ใช้แรงจากธรรมชาติ เช่น ลม ความแตกต่าง ของอุณหภูมิ และความแตกต่างของความดันอากาศ ซึ่งจะทำให้อากาศไหลผ่านเข้าออก ทางช่องเปิดของอาคาร โดยขึ้นอยู่กับสภาพอากาศ การออกแบบอาคาร และพฤติกรรมในการใช้งานอาคารของมนุษย์ การระบายอากาศธรรมชาติแบ่งออกเป็น 3 ประเภท ดังนี้

2.2.1.1 การระบายอากาศข้ามฟาก (Cross ventilation)

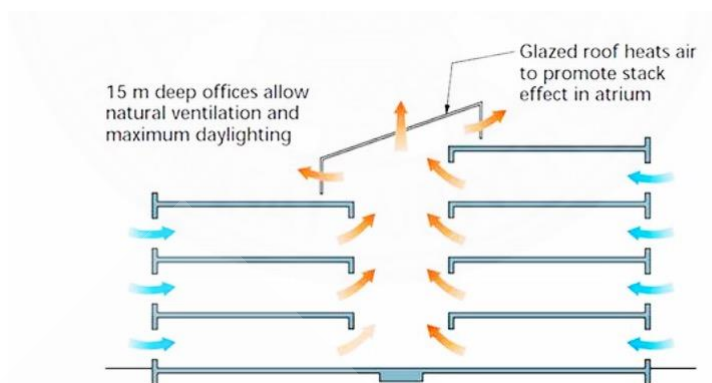
การระบายอากาศแบบข้ามฟากเกิดขึ้น โดยการเปิดช่องเปิดของเปลือกอาคาร สองด้านตรงข้ามกัน ซึ่งทำให้เกิดความแตกต่างของความกดอากาศที่ช่องอากาศเข้าและช่องอากาศออกทำให้อากาศไหลผ่านอาคารแต่ข้อจำกัดของการไหลของอากาศคือ ลักษณะของแปลนและความซับซ้อนซึ่งจะส่งผลกระทบต่อประสิทธิภาพในการกำจัดความร้อนภายในอาคาร และมลพิษที่เกิดขึ้นภายในอาคาร โดยความกว้างของพื้นที่ที่แนะนำจะต้องไม่เกิน 5 เท่า ของความสูงภายในพื้นที่นั้น



ภาพที่ 2.1 การระบายอากาศแบบข้ามฟาก (cross ventilation) จาก Automatic Natural by Ventilation .Andrew Dyke 2019

2.2.1.2 การระบายอากาศด้วยปล่องความร้อน (Stack ventilation)

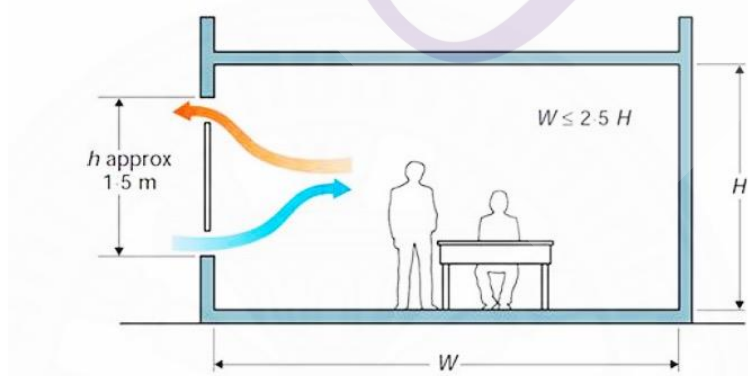
การระบายอากาศด้วยปล่องความร้อนเป็นการระบายอากาศที่อาศัยความแตกต่างของอุณหภูมิ (Buoyancy effect) ซึ่งจะเห็นได้ชัดเมื่อลมภายนอกมีความเร็วต่ำและบริเวณช่องอากาศออก จะมีความร้อนที่สูงกว่าส่วนอื่นๆ โดยอากาศร้อนจะลอยตัวสูงขึ้นและอากาศที่เย็นกว่าจะเข้าไปแทนที่ ทำให้เกิดการไหลของอากาศภายในอาคาร แต่การไหลของอากาศที่เกิดจากการเคลื่อนที่ไหลผ่านจะให้ประสิทธิภาพที่ดีกว่าการใช้หลักการความแตกต่างของอุณหภูมิ



ภาพที่ 2.2 การระบายความร้อนด้วยปล่องความร้อน (stack ventilation) จาก Automatic Natural by Ventilation .Andrew Dyke 2019

2.2.1.3 การระบายอากาศทางเดียว (Single-sided ventilation)

การระบายอากาศทางเดี่ยวมักพบเห็นในห้องเดียว เช่น ห้องพักคอนโดมิเนียม สำนักงาน หรือ ห้องที่มีช่องเปิดด้านเดียว โดยการไหลของอากาศเกิดจากความแตกต่างของอุณหภูมิภายในพื้นที่เป็นส่วนใหญ่และอาศัยหลักความแตกต่างของความดันอากาศเพียงเล็กน้อย โดยอากาศที่มีความร้อนจะลอยสูงขึ้น และไหลออกทางช่องด้านบนของช่องเปิดและอากาศเย็นจะไหลเข้ามาแทนที่อากาศร้อนที่ไหลออกไป โดยความกว้างของพื้นที่ใช้งานที่มีระบบระบายอากาศแบบทางเดียวที่ใช้หลักการความแตกต่างของอุณหภูมิ (Buoyancy effect) ต้องกว้างไม่เกิน 2.5 เท่าของความสูงของพื้นที่และไม่เกิน 2 เท่าของความสูงของพื้นที่ หากเป็นการระบายอากาศทางเดียวที่ใช้หลักความแตกต่างของความดันอากาศในการระบายอากาศ



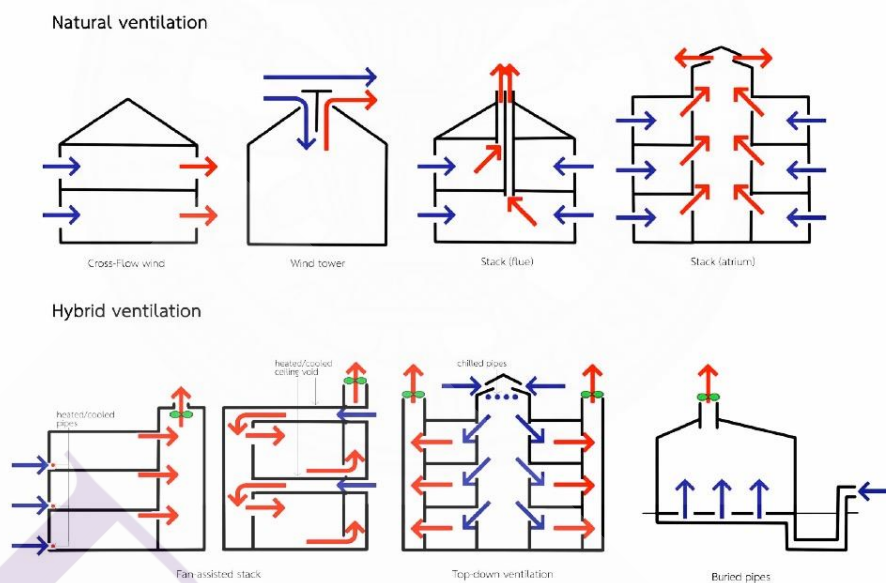
ภาพที่ 2.3 การระบายอากาศทางเดียว (single-sided ventilation) จาก Automatic Natural by Ventilation. Andrew Dyke 2019

2.2.2 การระบายอากาศด้วยระบบกล (Mechanical ventilation)

การนำเอาระบบกล เช่น พัดลมมาติดตั้งบนผนังหรือบริเวณหน้าต่าง หรือ การระบายอากาศผ่านทางระบบท่อ เพื่อให้อากาศมีการไหลเวียนภายในพื้นที่ โดยรูปแบบของระบบกลขึ้นอยู่กับสภาพแวดล้อมของพื้นที่ โดยสามารถติดตั้งระบบกล ขึ้นอยู่กับสภาพแวดล้อมของพื้นที่ โดยสามารถติดตั้งระบบกรองอากาศ เพื่อดักจับเศษฝุ่นละอองที่ลอยอยู่ในอากาศ หรือติดตั้งตัวดักจับความชื้นในอากาศหากในพื้นที่ที่มีความชื้นสูง โดยระบบกลระบายอากาศ สามารถควบคุมทิศทางการไหลของอากาศได้ดีกว่าการระบายอากาศแบบธรรมชาติ และยังสามารถควบคุมความกดอากาศภายในห้องให้เหมาะสมกับการใช้งานได้

2.2.3 การระบายอากาศแบบผสม (Hybrid ventilation)

ด้วยข้อจำกัดของการระบายอากาศแบบธรรมชาติ ที่ขึ้นอยู่กับสภาพอากาศภายนอก เช่น ลมภายนอกที่เบาเกินไปหรืออุณหภูมิภายนอกที่สูงเกินไป จึงทำให้ความในการใช้การระบายอากาศธรรมชาติลดลง การใช้การระบายอากาศแบบผสมคือระหว่างธรรมชาติ และระบบกลจึงเป็นวิธีอย่างง่ายในการจัดการ การระบายอากาศ ซึ่งสามารถปรับเปลี่ยนการใช้งานให้เหมาะสมกับหลายสภาพอากาศ โดยรูปแบบหลักในการใช้ระบบระบายอากาศแบบผสมมี 3 รูปแบบ คือ การปรับสลับใช้ระหว่างการระบายอากาศธรรมชาติ และระบบกล การใช้พัดลมในการระบายอากาศธรรมชาติ และการใช้งานร่วมกันของระบบระบายอากาศธรรมชาติ และระบบกลการระบายอากาศแบบผสม สามารถปรับใช้ได้ด้วยอาคารที่มีทางเดินเดี่ยว (Single corridor) อาคารที่มีทางเดินกลาง (Central corridor) อาคารที่มีโถงสูง (Atrium) และช่องระบายอากาศด้วยความร้อน (Chimney) ซึ่งสามารถนำระบบกลเข้าไปปรับใช้ เพื่อช่วยในการระบายอากาศที่เพิ่มขึ้นและที่สำคัญยังสามารถออกแบบและควบคุมทิศทางได้อีกด้วย



ภาพที่ 2.4 เปรียบเทียบความแตกต่างระหว่างการระบายอากาศธรรมชาติและการระบายอากาศแบบผสม จาก Natural ventilation for infection control in Health-care setting โดย Martin Liddament 2018. World health organization

ระบบระบายอากาศธรรมชาติแต่ละประเภทจะเหมาะสมต่อการใช้งานขึ้นอยู่กับสภาพภูมิอากาศของแต่ละพื้นที่ สภาพพื้นที่ภายในอาคาร รวมทั้งงบประมาณในการติดตั้งและบำรุงรักษาระบบระบายอากาศ โดยได้เปรียบเทียบข้อดีและข้อเสียของระบบระบายอากาศแต่ละประเภทในตารางที่ 2.2 ดังนี้

ตารางที่ 2.1 เปรียบเทียบข้อดีและข้อเสียระหว่างระบบระบายอากาศแบบธรรมชาติ ระบบกล ระบายอากาศ และระบบระบายอากาศแบบผสม

	ระบบระบายอากาศธรรมชาติ	ระบบกลระบายอากาศ	ระบบระบายอากาศแบบผสม
ข้อดี	เหมาะสำหรับพื้นที่ที่อากาศอบอุ่นและเย็น	เหมาะสำหรับทุกสภาพภูมิอากาศและปรับใช้กับระบบปรับอากาศได้	เหมาะสำหรับทุกสภาพภูมิอากาศ
	ใช้เงินทุนในการดำเนินการและค่าใช้จ่ายในการบำรุงรักษาที่ต่ำ	สามารถควบคุมและสร้างสภาวะความสบายในสภาพแวดล้อมได้	ประหยัดพลังงาน
	ให้อัตรการระบายอากาศสูง	สามารถควบคุมได้ในช่วงสั้นเช่นในห้อง	สามารถปรับเปลี่ยนการใช้งานให้เหมาะสมได้ทุกสภาพแวดล้อม
	สามารถควบคุมได้ในช่วงกว้าง		
ข้อเสีย	ได้รับผลกระทบจากสภาพอากาศภายนอกหรือพฤติกรรมจากผู้ใช้งานได้ง่าย	มีค่าใช้จ่ายในการติดตั้งและบำรุงรักษาสูง	อาจมีค่าใช้จ่ายสูง
	ยากในการคาดการณ์ วิเคราะห์ และออกแบบ	อาจเกิดสภาวะที่อัตราแลกเปลี่ยนอากาศไม่เป็นไปตามมาตรฐาน	ยากในการออกแบบ
	ไม่สามารถสร้างความสบายให้ผู้ใช้เมื่ออากาศร้อน ชื้น หรือหนาวเกินไป	อาจเกิดเสียงรบกวนจากอุปกรณ์	

หมายเหตุ: จาก Natural ventilation for infection control in health-care setting โดย Atkinson, James. Yves C., Camen L., Paul L., 2010. World health organization.

2.3 งานวิจัยด้านการระบายอากาศภายในอาคาร

2.3.1 รูปแบบการระบายอากาศที่ไม่ใช้ระบบปรับอากาศ

รูปแบบที่นิยมใช้ในปัจจุบันมี 2 รูปแบบ ที่นิยมใช้ในปัจจุบัน คือ การระบายอากาศโดยวิธีธรรมชาติและการระบายอากาศด้วยวิธีกล

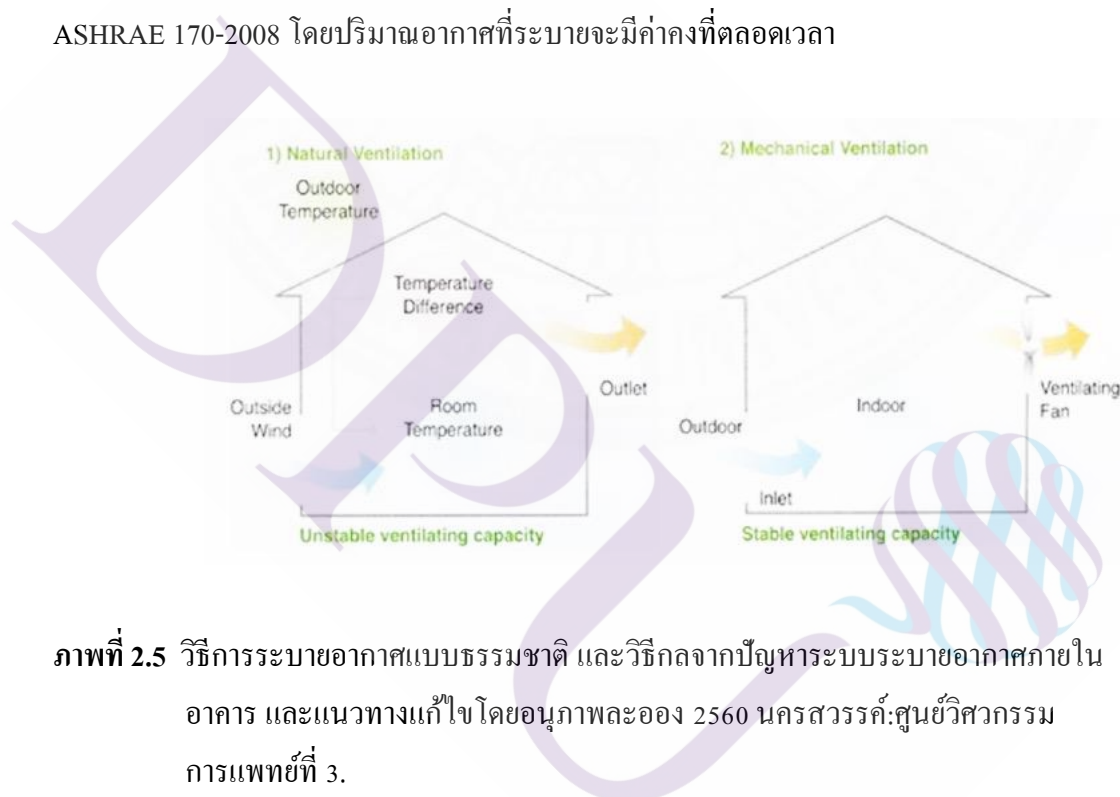
2.3.1.1 การระบายอากาศโดยวิธีธรรมชาติ (Natural ventilation)

เป็นการระบายอากาศด้วยธรรมชาติเพียงอย่างเดียว ซึ่งให้อากาศไหลผ่านอาคาร เพื่อขับดันและระบายอากาศภายในออกสู่ภายนอก โดยขึ้นอยู่กับความเร็วลมและอุณหภูมิ

ซึ่งเป็นสิ่งที่ไม่สามารถควบคุมให้คงที่ตลอดเวลาได้ โดยห้องหรืออาคารที่ใช้การระบายอากาศธรรมชาติต้องมีผลอย่างน้อย 1 ด้าน ที่มีช่อง เปิดออกสู่ภายนอกและต้องมีพื้นที่ๆ ลมสามารถไหลผ่านได้สุทธิไม่น้อยกว่าร้อยละ 10 เมื่อเทียบกับพื้นที่

2.3.1.2 การระบายอากาศโดยวิธีกล (Mechanical ventilation)

เป็นการนำเอาเครื่องกลที่ช่วยขับเคลื่อนอากาศหรือระบายอากาศ ซึ่งสามารถใช้กับพื้นที่ใด ๆ ก็ได้ โดยอุปกรณ์ขับเคลื่อนอากาศเพื่อทำให้เกิดการนำอากาศภายนอกเข้าสู่ภายในอาคารหรือบริเวณนั้น โดยมี อัตราไม่น้อยกว่าที่ระบุไว้ตามข้อกำหนดคุณภาพอากาศและมาตรฐานอากาศภายในอาคาร โดยวิศวกรรมสถานแห่งประเทศไทยและพระราชบัญญัติควบคุมอาคาร หรือ ASHRAE 170-2008 โดยปริมาณอากาศที่ระบายจะมีค่าคงที่ตลอดเวลา



ภาพที่ 2.5 วิธีการระบายอากาศแบบธรรมชาติ และวิธีกลจากปัญหาการระบายอากาศภายในอาคาร และแนวทางแก้ไขโดยอนุภาพละออง 2560 นครสวรรค์:ศูนย์วิศวกรรม การแพทย์ที่ 3.

ความต้องการอัตราไหลเวียนอากาศที่สูงนั้น ส่งผลต่อการใช้พลังงานไฟฟ้าที่สูงตามมาด้วย หากอาคารนั้นใช้ระบบปรับอากาศภายในอาคาร ซึ่งแตกต่างจากการระบายอากาศแบบธรรมชาติ ที่เกิดการไหลของอากาศจากความแตกต่างของอุณหภูมิ ความเร็วลม และ ความกดดันอากาศในบริเวณนั้น ซึ่งมีค่าใช้จ่ายที่น้อยกว่าการใช้พลังงานของระบบปรับอากาศ และได้อัตราการแลกเปลี่ยนอากาศต่อชั่วโมงที่สูงกว่า การใช้ระบบปรับอากาศ ซึ่งทำให้ความเสี่ยงต่อสุขภาพลดลง โดยเมื่อมีการแลกเปลี่ยนอากาศที่สูงขึ้น เวลาที่จะนำเอาอากาศที่แยกออกไปจากพื้นที่นั้นก็มีค่าน้อยลง แต่ข้อเสียของการระบายอากาศแบบธรรมชาติ คือ ความไม่คงที่ของสภาพแวดล้อมโดยรอบ ซึ่งส่งผลต่ออัตราการแลกเปลี่ยนอากาศที่ไม่คงที่ด้วยเช่นกัน

ตารางที่ 2.2 ตารางแสดงความสัมพันธ์ระหว่างค่าการแลกเปลี่ยนอากาศและเวลาในการกำจัดละอองฝุ่น

ACH	Minutes required for a removal efficiency of :		
	90%	99%	99.90%
1	138	276	414
2	69	138	207
3	46	92	198
4	35	69	104
5	28	55	83
6	23	46	69
7	20	39	59
8	17	35	52
9	15	31	46
10	14	28	41
11	13	25	38
12	12	23	35
13	11	21	32
14	10	20	30
15	9	18	28
20	7	14	21
25	6	11	17
30	5	9	14
40	3	7	10
50	3	6	8

หมายเหตุ: จากการออกแบบห้องเดี่ยวโดย สุพจน์ เตชะอำนวยวิทย์ 2551. กรุงเทพฯ : สมาคมวิศวกรรมปรับอากาศแห่งประเทศไทย.

2.4 เทคนิคการใช้พัดลมเพื่อเติมอากาศ ช่องแผ่นกรองฝุ่น และ ช่องระบายอากาศ

2.4.1 รูปแบบการปรับสภาพแวดล้อมด้วยการติดตั้งพัดลมระบาย และ ดูดอากาศรูปแบบที่นิยมใช้ในปัจจุบันมี 3 รูปแบบ ที่นิยมใช้ในปัจจุบันคือ ช่องเติมอากาศด้วยพัดลมติดผนังและชุดระบายอากาศฝ้าเพดานและชุดระบายความร้อน โถงหลังคา



ภาพที่ 2.6 ภาพรวมทั้งหมดของระบบระบายอากาศและการกรองฝุ่นละอองภายในอาคาร

2.4.1.1 ช่องเติมอากาศด้วยพัดลมดูดผนัง (Intake Air Grille by Ventilation)

เป็นชนิดของพัดลมระบายอากาศ แบบติดตั้งบนกำแพงของอาคาร ซึ่งเป็นช่องเล็กๆ ที่สามารถติดตั้งได้เรียบง่าย โดยสามารถติดตั้งได้ในระหว่างการก่อสร้างตัวอาคารหรือขณะที่มีการก่อสร้างเสร็จแล้ว โดยขึ้นอยู่กับลักษณะและแบบแปลนในการออกแบบตามทิศทางของการไหลอากาศที่เหมาะสม ซึ่งทำหน้าที่เติมอากาศใหม่เข้าสู่ตัวอาคาร เพื่อให้มีการถ่ายเทอากาศภายในอาคาร อย่างสม่ำเสมอ จุดเด่นมีแผ่นกรอง (Filter) ป้องกันแมลงและกรองฝุ่นละอองขนาด PM 2.5



ภาพที่ 2.7 วิธีการติดตั้งช่องเติมอากาศด้วยพัดลมดูดผนังเพื่อช่วยเติมอากาศเข้าภายในอาคาร

เป็นชนิดของพัดลมระบายอากาศแบบติดตั้งบนฝ้าเพดาน ซึ่งเป็นช่องเล็ก ๆ ทำหน้าที่
เร่งการระบายอากาศภายในอาคารออกสู่ภายนอกหรือขณะที่มีการก่อสร้างเสร็จแล้ว

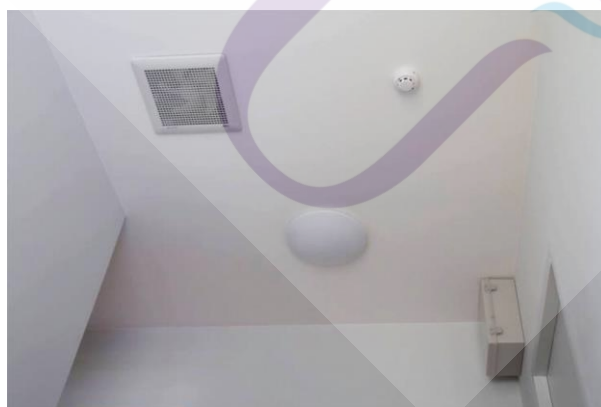


ลักษณะการติดตั้งเดิมอากาศติดผนัง
(Intake Air Grille)

ภาพที่ 2.8 ลักษณะการติดตั้งช่องเดิมอากาศติดผนัง ให้เห็นทั้งภายในและภายนอกอาคาร

2.4.1.2 ชุดระบายอากาศฝ้าเพดาน (Ceiling Ventilation)

เป็นชนิดของพัดลมระบายอากาศแบบติดตั้งบนฝ้าเพดาน ซึ่งเป็นช่องเล็ก ๆ
ซึ่งทำหน้าที่เร่งการระบายอากาศภายในอาคารออกสู่ภายนอก



ชุดระบายอากาศฝ้าเพดาน (Ceiling Ventilator)

ภาพที่ 2.9 ลักษณะการติดตั้งช่องเดิมอากาศติดผนัง ให้เห็นทั้งภายในและภายนอกอาคาร

2.4.1.3 ชุดระบายความร้อนในโถงหลังคา (Solar Roof Tile Ventilation)

ทำหน้าที่ช่วยเร่งการระบายอากาศภายในห้องโถงใต้หลังคาออกสู่ตัวนอกอาคาร ช่วยให้อาคารมีการแลกเปลี่ยนความร้อนของอากาศ ออกสู่ภายนอกของอาคารได้ตลอดเวลา จุดเด่น คือ มีระบบป้องกันน้ำฝนไหลย้อนกลับเข้าภายในหลังคา ข้อเสีย คือ มีต้นทุนสูง



ชุดระบายความร้อนในโถงหลังคา
(Solar Roof Tile Ventilator)

ภาพที่ 2.10 ชุดพัดลมระบายความร้อนห้องโถงหลังคาช่วยให้เกิดการไหลเวียนและอากาศมีการไหลเวียนอยู่ตลอดเวลา

2.5 เทคนิคการตรวจวัดด้วยเซ็นเซอร์และคัดกรองฝุ่น PM2.5 ด้วยฟิลเตอร์

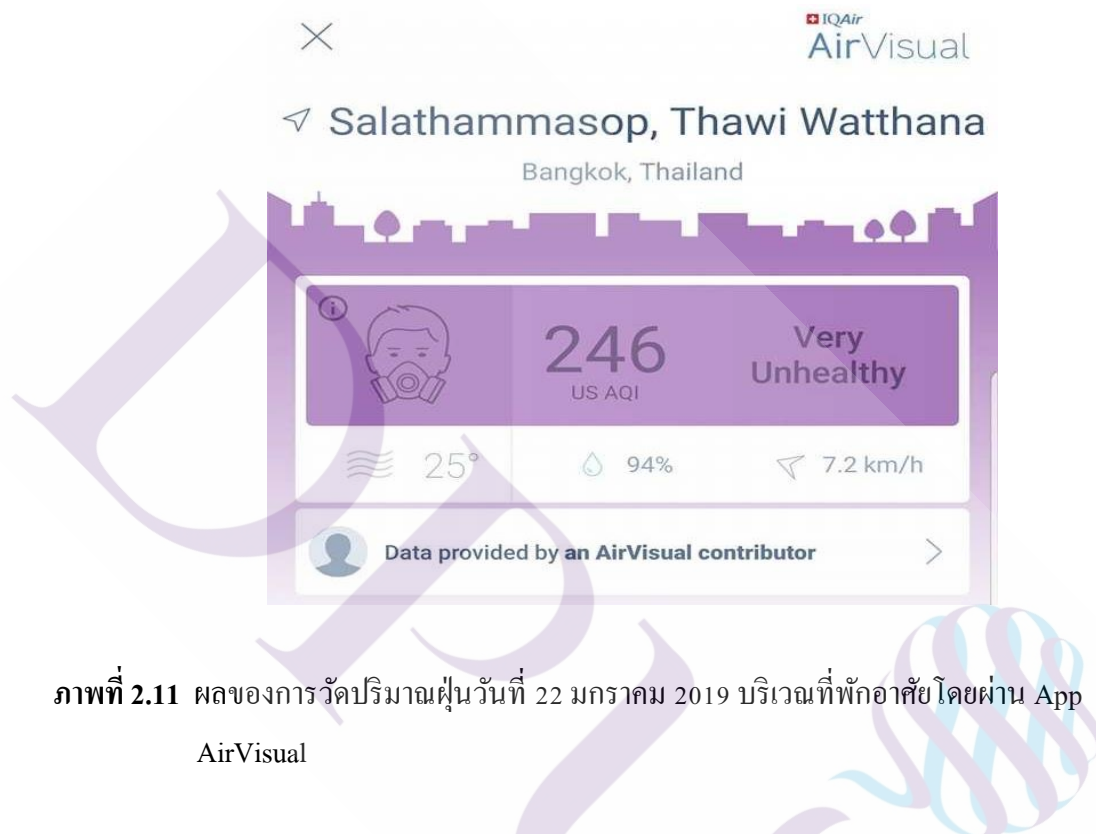
PM หมายถึง Particulate Matter ที่พุดถึงปริมาณของสิ่งในอากาศไม่ว่าจะเป็นของแข็ง เช่น (ฝุ่นและเกสรดอกไม้) และของเหลว (ที่ระเหยได้ เช่น ฟอร์มัลดีไฮด์) โดยของที่อยู่ในอากาศเหล่านี้ก็มีการแบ่งออกอีกเป็น 2 ประเภทด้วยกันตามขนาดของมัน คือ แบบหยาบ (Coarse Particle) นั่นคือ อนุภาคที่มีขนาดใหญ่ 2.5 Micron และแบบละเอียด (Fine Particle) คืออนุภาคที่เล็กกว่า 2.5 Micron (เป็นคำอ่านทับศัพท์จากคำว่า Micron และมาจากคำเต็มๆว่า Micrometer) กล่าวคือ PM หรือ Particulate Matter คือจำนวนของที่ลอยอยู่ในอากาศนั่นเอง

2.5.1 การวัดคุณภาพอากาศ (Air Quality Index : AQI)

การวัดคุณภาพอากาศใช้หลายค่าเข้ามาประกอบเช่น Ozone ส่วนที่เป็น PM จะใช้ตั้งแต่ PM10 จนถึง PM2.5 ในการคำนวณ ซึ่งในบางมาตรฐานก็อาจจะมีการใช้ปริมาณของสารประกอบอื่นๆ ในการคำนวณอีกหลายแบบ ดังนั้นที่เราเคยพบปัญหาว่าทำไมค่า AQI (Air Quality Index) ของฝั่งประเทศไทยกับใน Application ในต่างประเทศที่เราใช้กันทำไมไม่ตรงกันหรือเท่ากัน

ก็เพราะว่า มาตรฐานในการคำนวณไม่เท่ากันนั่นเอง

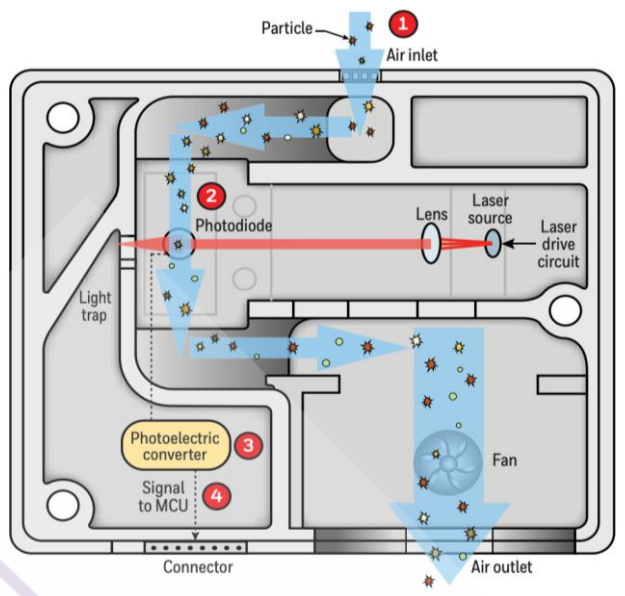
สำหรับการวัดค่าในประเทศไทย เราใช้มาตรฐานแบบ 6 ปัจจัยในการวัดผลในการคำนวณด้วยกัน ตั้งแต่ PM10, PM2.5, Ozone, Carbon Monoxide, Nitrogen Dioxide และ Sulfur Dioxide สามารถเข้าไปดูและคำนวณได้ในเว็บของกรมควบคุมมลพิษ



ภาพที่ 2.11 ผลของการวัดปริมาณฝุ่นวันที่ 22 มกราคม 2019 บริเวณที่พักอาศัยโดยผ่าน App AirVisual

2.5.2 การใช้เครื่องวัดฝุ่น PM2.5 (PM2.5 Measuring)

โดยทั่วไปมีหลายวิธี แต่วิธีที่นิยมและเรียบง่ายที่สุดจะเป็นการวัดค่าจากเครื่องแบบพกพาหรือตามบ้าน โดยการใช้แสงเลเซอร์ (Laser) เราทราบว่าแสงสามารถตกกระทบและกระจายหรือสะท้อน เมื่อเจอกับบางอย่างแน่นอนว่าถ้ามันตกกระทบกับของที่ใหญ่กว่าย่อมทำให้เกิดการกระจายและสะท้อน ได้มากกว่าของเล็กแน่นอน เซ็นเซอร์จะวัดความเข้มและมุมสะท้อนกลับจากแสงเลเซอร์ (Laser) ที่ยิงผ่านอนุภาคที่อยู่ในอากาศ และประมวลผลออกมาเป็นค่าจำนวน PM2.5 ที่เราได้เห็นจากเครื่องนั่นเอง



ภาพที่ 2.12 การทำงานใน 4 ขั้นตอนของเครื่องตรวจวัดปริมาณฝุ่นจากยี่ห้อ Honeywell HPM Series

โดยขั้นตอนการทำงานของเครื่องตรวจวัดปริมาณฝุ่นจากยี่ห้อ Honeywell HPM Series ได้แบ่งการทำงานออกเป็น 4 ขั้นตอนดังนี้

1. พัดลมที่เต้าเสียบอากาศดึงอากาศเข้าผ่านทางอากาศเข้า
2. ตัวอย่างอากาศถูกดูดผ่านลำแสงเลเซอร์ซึ่งแสงที่สะท้อนออกมาจากอนุภาคถูกจับวิเคราะห์
3. โฟโตอิเล็กทริกคอนเวอร์เตอร์ประมวลผลสัญญาณเป็นขนาดและความหนาแน่นของอนุภาค
4. สัญญาณจะถูกส่งไปยังหน่วยควบคุมไมโคร(MCU) ซึ่งอัลกอริทึมที่ถูกออกแบบมาเพื่อการประมวลผลข้อมูลความหนาแน่นของอนุภาค ($\mu\text{g}/\text{m}^3$).

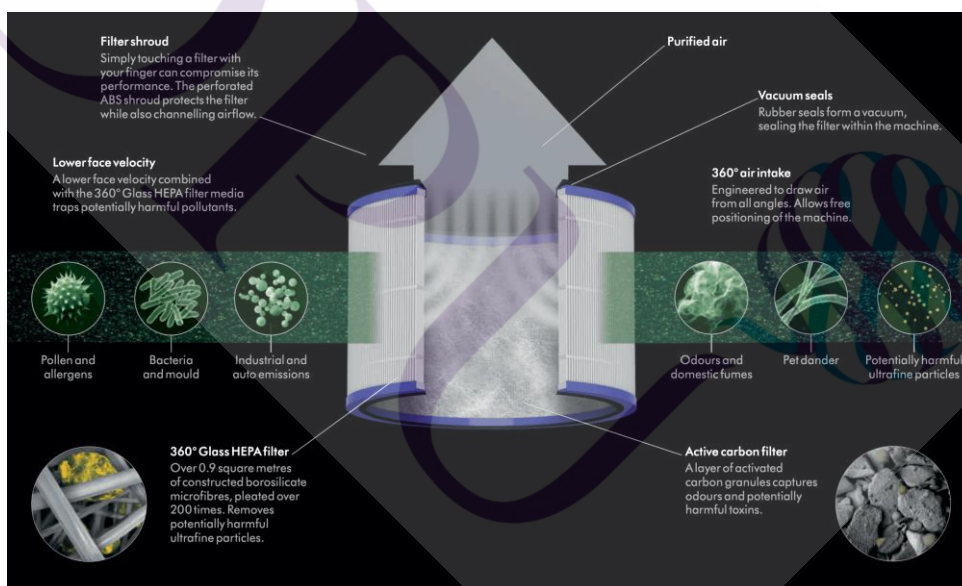
หลังจากได้รับเอาต์พุตของเซ็นเซอร์ อุปกรณ์จะทำการตอบสนองตามที่โปรแกรมได้มีการออกแบบไว้ตามการใช้งาน

2.5.3 การใช้แผ่นกรองอากาศ (Air Filter)

โดยทั่วไปเราจะพบแผ่นกรองทั่วไป โดยเฉพาะในหน้ากากกันฝุ่นและเครื่องฟอกอากาศ โดยทั้งหมดนี้มีหลักการเดียวกันคือ การทำให้อากาศไหลผ่านแผ่นกรองเพื่อป้องกันอนุภาคจากฝุ่นที่เราไม่ต้องการให้ผ่านเข้าจมูกหรือสูดดมสู่ปอดโดยไม่จำเป็น โดยมาตรฐานยุโรปหรือ (EU) ได้กำหนดมาตรฐานของแผ่นกรอง (Filters) ออกตามการใช้งานในลักษณะ 5 แบบ ด้วยกันคือ

Coarse Filters, Fine Filters, Semi HEPA (หรือบางที่เรียกว่า EPA), HEPA และ ULPA

แต่ที่เราคุ้นเคยกันดีจากเครื่องกรองอากาศคือ HEPA (High Efficiency Particulate Air Filters) ซึ่งมันสามารถกรองอนุภาคที่เล็กกว่า 0.3 Micron ทำให้อนุภาคขนาดเล็กและแบคทีเรียต่างๆ ไม่สามารถเล็ดรอดไปได้ ส่วนวัสดุที่นำมาใช้ผลิตแผ่นกรอง ส่วนใหญ่จะเป็น Fiberglass ที่ทอแน่น และ Random ที่สุดเท่าที่จะเป็นไปได้ เพื่อให้ได้ประสิทธิภาพสูงสุดของการกรองที่มากขึ้น ทำให้การดูแลแผ่นกรองที่เป็น Fiberglass ไม่ควรซักล้างด้วยน้ำยาอะไรเลย เพราะจะไปกัดกร่อน ทำให้ประสิทธิภาพของการกรองไม่มีประสิทธิภาพ หรือไม่สามารถกรองได้เลย การดูแลถ้าต้องทำความสะอาด ควรใช้เครื่องดูดฝุ่นหรือแปรงขัดทำความสะอาด หรือน้ำเปล่าก็ได้ ทำให้ HEPA Filters สามารถกรองฝุ่น PM2.5 ได้แน่นอน เพราะมีขนาดใหญ่กว่า 0.3 Micron ตามประสิทธิภาพของแผ่นกรอง ใน HEPA Filters จะถูกแบ่งย่อยออกเป็นอีก 5 Class ย่อยๆ ตั้งแต่ H10-H14 โดยจะต่างกันที่ระดับการป้องกันตั้งแต่ 85% สำหรับ H10 ไปจนถึง 99.95% สำหรับ H14

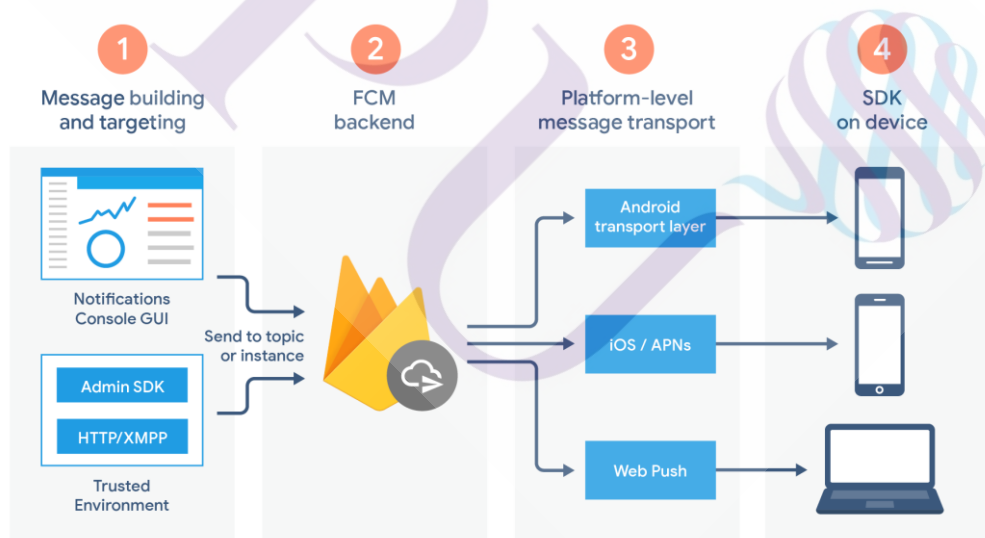


ภาพที่ 2.13 การทำงานของแผ่นกรอง HEPA Filters จาก Source : dyson.com

นอกจากมาตรฐานของ EU อย่าง EN779 ที่ได้กล่าวไปแล้ว ในส่วนของ ISO ก็ได้ออกมาตราฐานมาใหม่เช่นกัน คือ มาตรฐาน ISO 16890 ที่มีความละเอียดมากถึง 49 Class ในขณะที่ EN779 มีเพียง 9 Class เท่านั้น

2.6 องค์ประกอบของระบบ IoT

เทคโนโลยี Internet of Things (IoT) [3] หรือ “อินเทอร์เน็ตในทุกสิ่ง” หมายถึง การที่สิ่งต่างๆ ถูกเชื่อมโยงทุกอย่างเข้าสู่โลกอินเทอร์เน็ต ทำให้มนุษย์สามารถสั่งการควบคุมใช้งานอุปกรณ์ต่างๆ ผ่านทางเครือข่ายอินเทอร์เน็ต เช่น การสั่งเปิด-ปิด อุปกรณ์เครื่องใช้ไฟฟ้า รถยนต์ โทรศัพท์มือถือ เครื่องมือสื่อสาร เครื่องใช้สำนักงาน เครื่องมือทางการแพทย์ เครื่องจักรในโรงงานอุตสาหกรรม อาคาร บ้านเรือน เครื่องใช้ในชีวิตประจำวันต่าง ๆ ผ่านเครือข่ายอินเทอร์เน็ต เป็นต้น โดยเทคโนโลยีนี้จะเป็นทั้งประโยชน์อย่างมหาศาลและความเสี่ยงไปพร้อม ๆ กัน เพราะหากระบบรักษาความปลอดภัยของอุปกรณ์และเครือข่ายอินเทอร์เน็ตไม่ดีพอ จะทำให้ผู้ไม่ประสงค์ดีเข้ามากระทำการที่ไม่พึงประสงค์ต่ออุปกรณ์ข้อมูลสารสนเทศหรือความเป็นส่วนตัวของบุคคลได้ ดังนั้น การพัฒนาไปสู่ Internet of Things จึงมีความจำเป็นต้องพัฒนามาตรการและเทคนิคในการรักษาความปลอดภัยที่ควบคู่กันไปด้วย จากภาพที่ 2.14 ได้แสดง Network Layers ของ IoT ซึ่งประกอบไปด้วย Controlling device, Cloud Service, Global Network, Local Network และ Things ผู้วิจัยได้เล็งเห็นความสำคัญด้านการรักษาความปลอดภัย ดังนั้นจึงเลือกที่จะใช้บริการของ Console Firebase Google Cloud Website Service Platform ที่นำมาใช้ในการจัดการบริหารระบบ โดยมีรายละเอียดดังต่อไปนี้

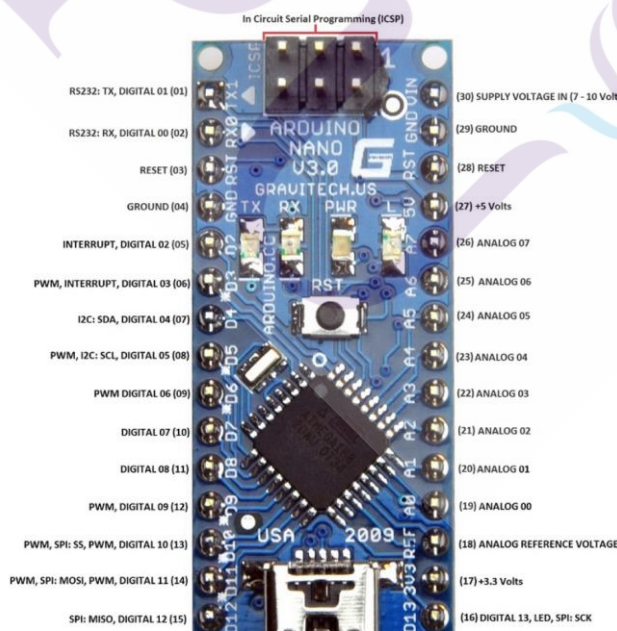


ภาพที่ 2.14 Network Layers ของ Internet of Things โดย Console Firebase Google Cloud Website Service Platform

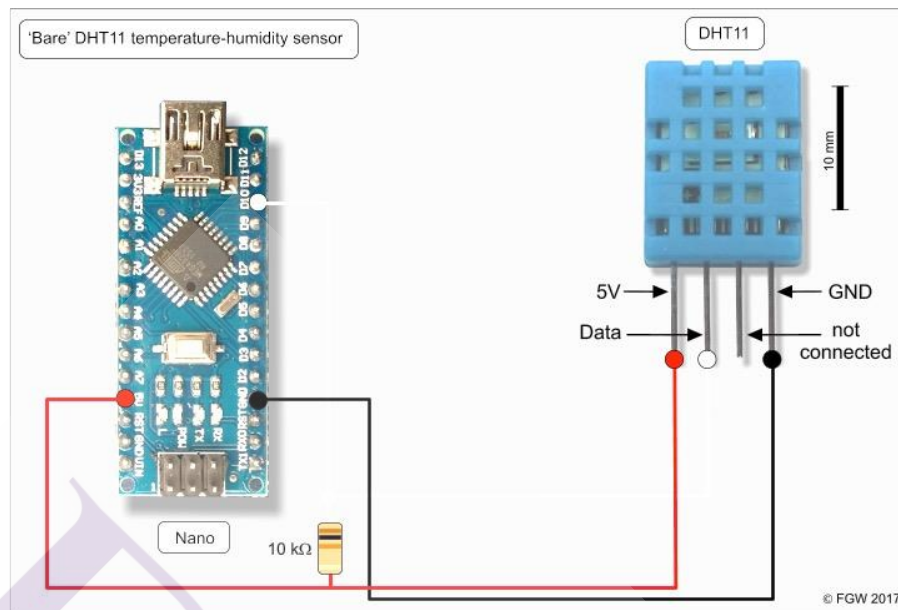
2.6.1 Things

Things ในที่นี้แทนความหมายถึง อุปกรณ์อิเล็กทรอนิกส์ที่เชื่อมอุปกรณ์กับเครื่องมือและเซนเซอร์ต่าง ๆ เพื่อเก็บรวบรวมข้อมูลตามที่ผู้ออกแบบต้องการเรียกว่า Sensor Node โดยอุปกรณ์อิเล็กทรอนิกส์หรือ Sensor Node ในปัจจุบันมีการพัฒนาอย่างรวดเร็วซึ่งยังมีราคาที่ถูกจึงเหมาะแก่การนำมาใช้งาน ในงานวิจัยนี้ผู้วิจัยได้ยกตัวอย่างและรายละเอียด Things และ Sensor ที่ได้นำมาใช้ในงานวิจัยชิ้นนี้

Arduino Uno Nano [14] จุดเด่นที่ทำให้เลือกใช้บอร์ด Arduino มาเป็น Sensor Node เพราะง่ายต่อการพัฒนามีรูปแบบคำสั่งพื้นฐานไม่ซับซ้อนเหมาะสำหรับผู้เริ่มต้นมี Arduino Community กลุ่มคนที่ร่วมกันพัฒนาที่แข็งแกร่ง Open Hardware ทำให้ผู้ใช้สามารถนำบอร์ดไปต่อยอดใช้งานได้หลายด้านราคาไม่แพง Cross Platform สามารถพัฒนาโปรแกรมบน OS ใดก็ได้และสำหรับรุ่น Nano นี้จะมีขนาดเล็ก ซึ่งมีคุณสมบัติดังนี้ ใช้โปรเซสเซอร์รุ่น Atmega 328 พอร์ต USB ใช้สำหรับต่อกับ Computer เพื่ออัปโหลดโปรแกรมเข้า MCU และจ่ายไฟให้กับบอร์ด พอร์ต I/O Digital I/O ตั้งแต่ขา D0 ถึง D13 นอกจากนี้ บาง Pin จะทำหน้าที่อื่นๆ เพิ่มเติมด้วย เช่น Pin0,1 เป็นขา Tx, Rx Serial, Pin3,5,6,9,10 และ 11 เป็นขา PWM พอร์ต I/O Analog ตั้งแต่ขา A0-A5 รองรับไฟที่แรงดัน 7-12 V

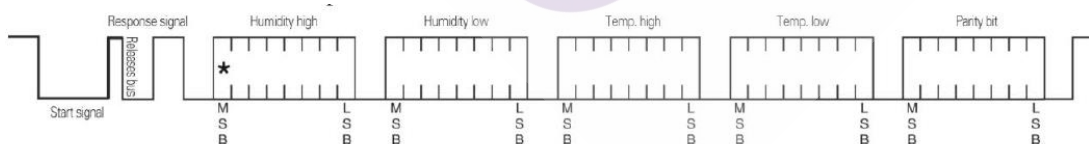


ภาพที่ 2.15 อุปกรณ์ Microcontroller Arduino Uno Nano



ภาพที่ 2.16 เซ็นเซอร์ตรวจวัดอุณหภูมิ DHT11 Temperature And Humidity Sensors

เซ็นเซอร์ตรวจวัดความชื้นสัมพัทธ์และอุณหภูมิ DHT11 / AM2301 [15] ที่สามารถวัดได้ในช่วง 0 ถึง 100% เปร้อเซ็นต์ความชื้นสัมพัทธ์ และ -40 ถึง 80 องศาเซลเซียส โดยมีความคลาดเคลื่อนอยู่ที่ ± 2 เปร้อเซ็นต์ความชื้นสัมพัทธ์ และ ± 0.5 องศาเซลเซียสตามลำดับ โดยมีหลักการทำงานดังนี้ เมื่อ Micro Controller Unit (MCU) ให้สัญญาณเริ่มการทำงาน DHT21 จะเปลี่ยนจากสถานะเตรียมพร้อมเป็นเริ่มทำงาน โดยจะส่งข้อมูลอุณหภูมิและความชื้นกลับไปยัง MCU ในรูปแบบสัญญาณดิจิทัล 40 บิต เมื่อส่งเสร็จแล้ว DHT21 จะกลับสู่สถานะเตรียมพร้อมอีกครั้งเพื่อรอสัญญาณเริ่มการทำงานจาก MCU



ภาพที่ 2.17 การทำงานของ DHT11

ข้อมูลดิจิทัล 40 บิต แบ่งเป็น 16 บิต ความชื้นสัมพัทธ์ 16 บิต อุณหภูมิ และ 8 บิต ตรวจสอบถูกต้อง (Check-sum)

- ตัวอย่าง 0000 0010 1000 1100 0000 0001 0101 1111 1110 1110 0000 0010 1000 1100 = 652

คิดเป็น 65.2 เปอร์เซ็นต์ความชื้นสัมพัทธ์

0000 0001 0101 1111 = 351 คิดเป็น 35.1 องศาเซลเซียส

ผลรวมของ 32 บิตแรก คือ

0000 0010+1000 1100+0000 0001+0101 1111 = 1110 1110

เท่ากับ 8 บิต สุดท้าย

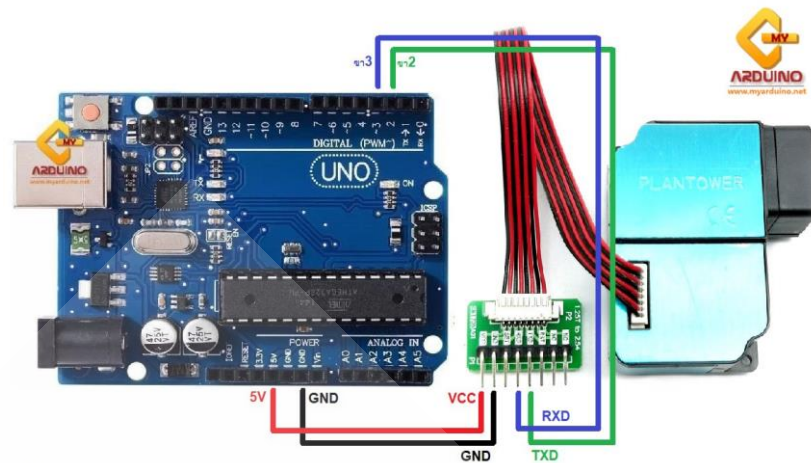
เซนเซอร์ตรวจวัดปริมาณฝุ่น Plantower Laser PM2.5 Dust Sensor PMS7003 [13] PMS7003 เป็น โมดูลตรวจวัดความเข้มข้นของอนุภาคแบบดิจิทัลโดยใช้หลักการการกระเจิงด้วยเลเซอร์สามารถรวบรวม และคำนวณจำนวนอนุภาคแขวนลอยได้ต่อเนื่องในปริมาตรอากาศที่แตกต่างกันต่อหน่วยปริมาตร นั่นคือการกระจายความเข้มข้นของอนุภาคแล้วแปลงเป็นความเข้มข้นของมวล หลักการทำงานของตัวตรวจจับใช้หลักการของการกระเจิงของเลเซอร์หมายความว่าแสงเลเซอร์ จะถูกยิงไปบนอนุภาคของฝุ่นละอองในอากาศและเกิดการกระเจิง

ในชุดประกอบด้วย

- PMS7003 Plantower
- แผงวงจรต่อใช้งาน
- สายเชื่อมต่อ JST8AA-6
- สาย IDC1MM-8 จำนวน 4 เส้น
- สาย IDC1MF-8 จำนวน 4 เส้น

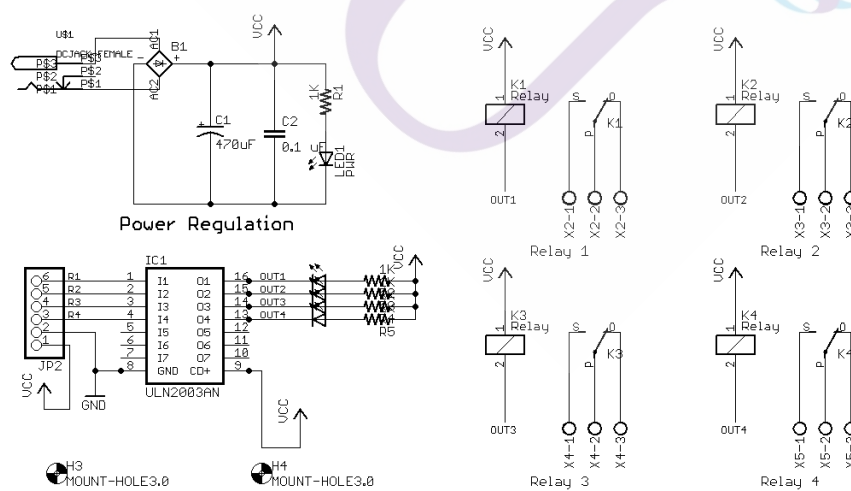
คุณสมบัติ

- อาศัยการกระเจิงของเลเซอร์เพื่อให้ได้การวัดที่แม่นยำ
- การตอบสนองและการสนับสนุนแบบเรียลไทม์
- ขนาดอนุภาคที่มีความละเอียดน้อยที่สุด 0.3 μm
- ช่องทางเข้าของฝุ่น ผู้ใช้ไม่จำเป็นต้องต่อท่อเสริม
- ใช้ไฟเลี้ยง 4.5 ถึง 5.5 โวลต์
- ขนาดบางเพียง 12mm



ภาพที่ 2.18 อุปกรณ์ เซ็นเซอร์ตรวจวัดปริมาณฝุ่น Plan tower Laser PM2.5 Dust Sensor PMS7003

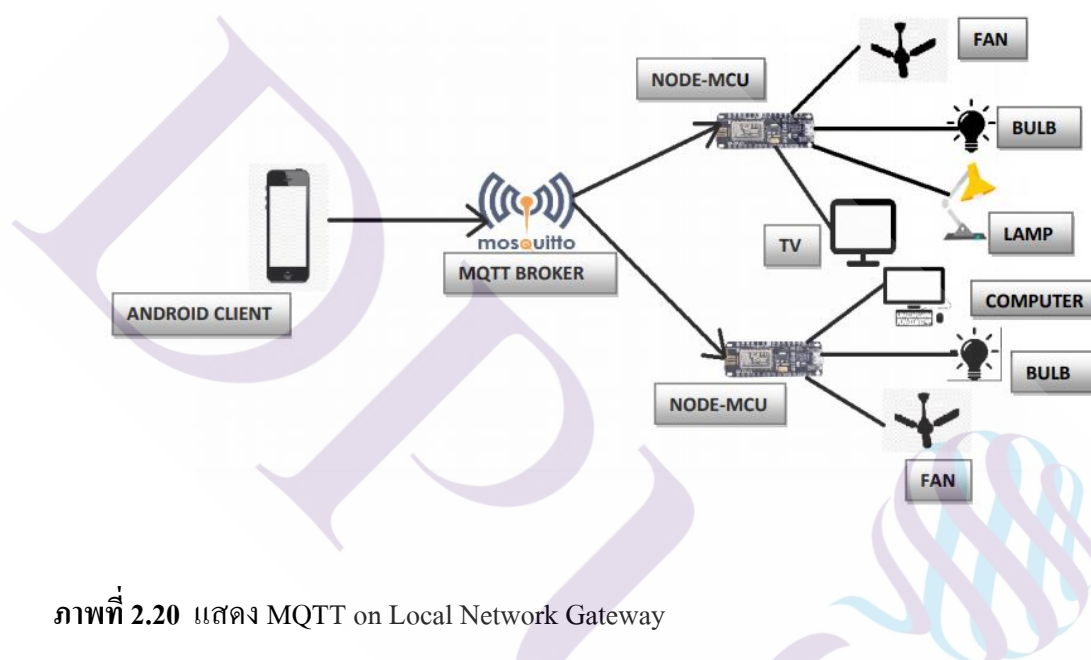
Relay Switch 4 Channel 12V relay Active High/Low แบบแยกอิสระ 2 ช่อง พร้อมไฟ LED แสดงผลการทำงานช่องต่อแบบ terminal สามารถต่อสายไฟได้สะดวกโมดูลนี้สามารถทำงานได้ทั้ง 2 แบบคือ Active High เซต Jumper ขากลางไว้ที่ H โดยเมื่อป้อนไฟ 3-12V หรือสัญญาณ 1 ที่ขา IN1-IN4 บอร์ดนี้เลยจะทำงาน Active Low เซต Jumper ขากลางไว้ที่ L โดยเมื่อป้อนไฟ 0V หรือสัญญาณ 0 จะทำงานเปิด Switch Relay โดยในงานวิจัยนี้เรานำมาประยุกต์ใช้สำหรับควบคุมการเปิด-ปิดอุปกรณ์ไฟฟ้าในระบบ คือ ระบบดูดอากาศเข้าและระบบระบายความร้อนออกนอกอาคาร



ภาพที่ 2.19 วงจรการทำงานของ Relay Switch 4 Channel

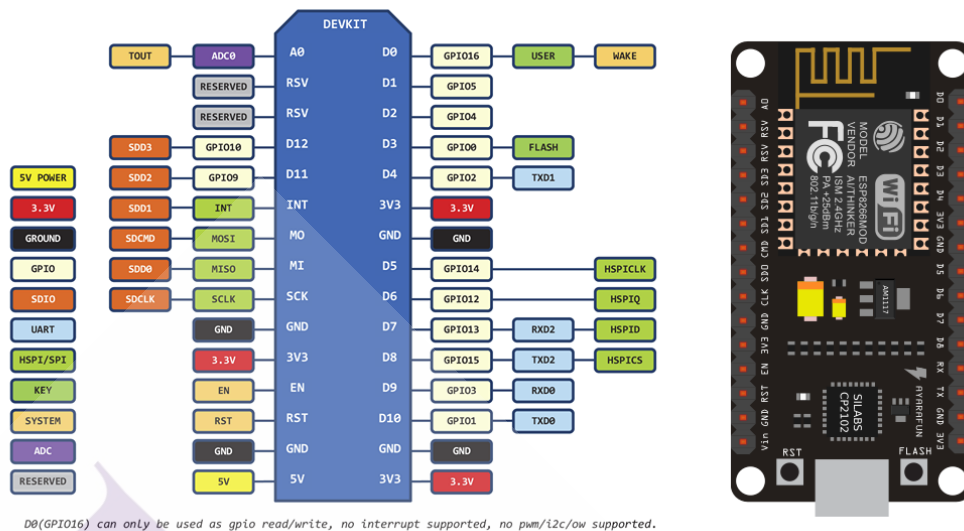
2.6.2 Local Network

เมื่อมีโครงข่าย Sensor nodes [3] แล้วก็จำเป็นจะต้องมี Gateway เพื่อจะเชื่อมต่อไปยังอินเทอร์เน็ตด้วย โดยตัว Gateway นี้จะทำหน้าที่เชื่อมต่อไปยังเครือข่าย Internet ให้อุปกรณ์ในโครงข่าย Sensor nodes ทั้งหมดส่งข้อมูลเข้าสู่อินเทอร์เน็ตได้นั่นเอง และเจ้า Gateway ที่ว่านี้ก็จะอยู่ภายใต้ Local network ซึ่งจะมีการกำหนดกันต่อไปว่า Gateway ภายใต้ Local network ที่ว่านั้นจะให้เชื่อมต่อไปยัง Internet ได้ด้วยหรือไม่ถ้าไม่ได้ อุปกรณ์ที่เชื่อมเข้ามาใน Gateway ก็อาจจะสื่อสารกันได้เฉพาะภายใน Local network เองได้เท่านั้น



ภาพที่ 2.20 แสดง MQTT on Local Network Gateway

ผู้วิจัยได้นำ Node MCU V2 ESP8266 [11] Gateway ทำหน้าที่เป็นตัวกลางในการรับส่งข้อมูลจากอุปกรณ์ Sensor Node และ AWS Cloud โดยเลือกใช้ Node MCU V2 ESP8266 [11] มีขนาดเล็กกะทัดรัด บอร์ดมีความกว้าง 2.5CM ใช้ชิพ USB เบอร์ CP2102ในการติดต่อกับคอมพิวเตอร์ เพื่อลงโปรแกรมสามารถลง Firmware Node MCU และเขียนโปรแกรมด้วยภาษา Lau และ Arduino ได้



ภาพที่ 2.21 อุปกรณ์ Microcontroller Node MCU V2 ESP8266

2.6.3 Global Network

Global Network หรือ Internet คือ เครือข่ายคอมพิวเตอร์ที่เชื่อมต่อกันเป็นจำนวนมาก ครอบคลุมไปทั่วโลกอาศัย โครงสร้างระบบสื่อสารโทรคมนาคม เป็นตัวกลางในการแลกเปลี่ยนข้อมูล มีการประยุกต์ใช้งานหลากหลายรูปแบบ ปัจจุบันข้อมูลข่าวสารที่ถูกส่งผ่านทางอินเทอร์เน็ตมีจำนวนมากขึ้น และเส้นทางของข้อมูลนั้นกำลังเปลี่ยนไป โดยที่สิ่งของวัตถุต่าง ๆ กลายเป็นส่วนหนึ่งของระบบข้อมูลที่เรียกว่า Internet of Things โดยการใช้ตัวเซนเซอร์ หรือตัวเก็บข้อมูลซึ่งฝังอยู่ในวัตถุต่าง ๆ โดยสิ่งเหล่านี้จะสร้างความได้เปรียบเพราะจะรู้ข้อมูลแม่นยำมากกว่า

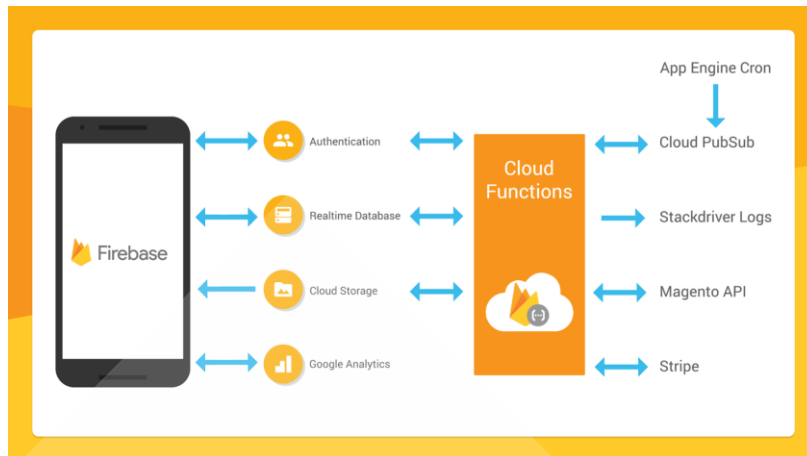
2.6.4 Cloud Service

Cloud คือ การให้บริการด้าน IT โดยผู้ใช้งานสามารถปรับเปลี่ยนรูปแบบการใช้งานได้อย่างอิสระ มีความยืดหยุ่นสูง บริการใช้งาน สามารถใช้งาน Cloud ผ่าน Internet ดังนั้น Cloud จึงเป็นการบริการด้าน IT ที่สามารถใช้งานได้โดยไม่จำกัดสถานที่ปรับเปลี่ยนตามความเหมาะสมได้อย่างรวดเร็ว และประหยัดค่าใช้จ่าย ยกตัวอย่างบริการของ Console Firebase Google Cloud Website Service [10] เป็นผู้ให้บริการ Cloud ที่เป็น Infrastructure as a Service. (IaaS) มี Service ต่าง ๆ ที่หลากหลายกว่าและมีโครงสร้างพื้นฐานที่ดีมาก Console Firebase Google Cloud โดยระบบได้ใช้บริการของ Console Firebase Cloud ในส่วนต่างๆ พร้อมทั้งรายละเอียดภายในระบบดังนี้



ภาพที่ 2.22 โครงสร้างของ Console Firebase Google Cloud Messaging

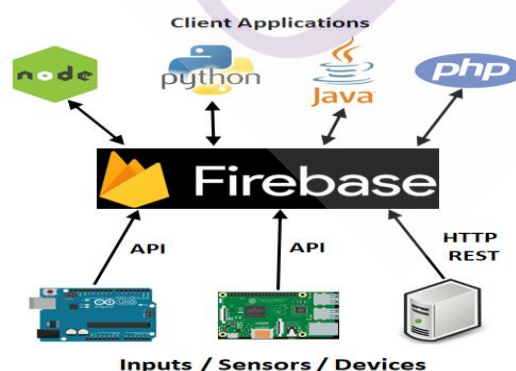
Firebase (Cloud Messaging) [10] หลักการทำงานก็คือการสร้าง Virtual Server ขึ้นมาใช้งาน โดยที่ผู้ใช้สามารถเลือกได้ว่าต้องการทรัพยากรขนาดมากน้อยเพียงใด ได้ตามต้องการ เหตุผลที่ระบบเลือกใช้ Firebase Cloud Messaging จะไม่มีค่าใช้จ่ายในส่วนของอุปกรณ์เซิร์ฟเวอร์ และค่าซอฟต์แวร์ โคลนเซนต์ ของระบบปฏิบัติการในการใช้งานระบบ Firebase เราก็จะเสียค่าใช้จ่ายตามปริมาณทรัพยากรที่เราใช้เท่านั้น จุดเด่นของ Firebase สามารถสร้าง VM ขึ้นมาใช้งานได้อย่างรวดเร็วภายใต้เลือกชนิดของ OS ที่ต้องการได้ สามารถเลือกขนาดที่ต้องการได้ ตั้งค่าระบบรักษาความปลอดภัย โดยสามารถกำหนดได้เอง สามารถต่อยอดในการทำระบบ Backup หรือ Disaster Recover ได้และมีค่าใช้จ่ายต่ำ ไม่มีค่าใช้จ่ายเรื่องฮาร์ดแวร์ ประหยัดค่าใช้จ่ายในการลงทุนเป็นอย่างมาก กรณีที่ระบบมีการเติบโตหรือมีผู้ใช้งานจำนวนมากขึ้น ก็สามารถเพิ่มขนาดรวมถึงทรัพยากรของ VM ให้มีขยายขนาดใหญ่ขึ้นได้อัตโนมัติ สำหรับเป็น Web Server ของระบบเลือกใช้ระบบปฏิบัติการ Ubuntu



ภาพที่ 2.23 โครงสร้างของ Console Firebase Google Cloud Functions

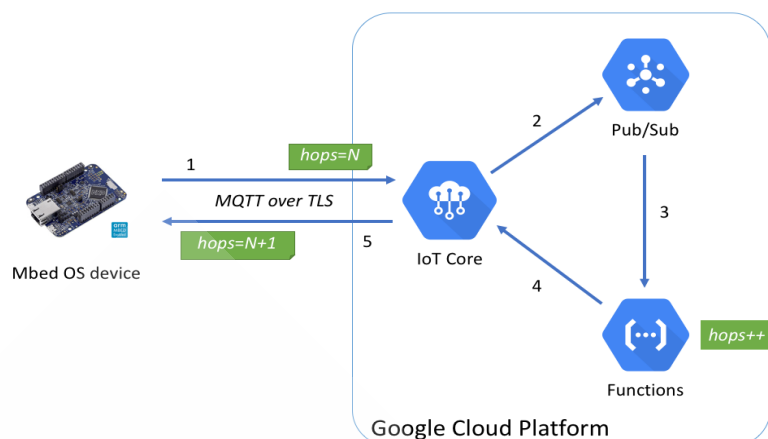
Firebase (Identity and Access Management) [10] คือระบบที่จะช่วยในการทำ Authentication และทำ Access Control ผู้เข้าถึงระบบ Server ต่างๆ เช่น สิทธิการใช้งาน ด้วยการให้ข้อมูลประจำตัวเฉพาะกับผู้ใช้ทุกคนที่อยู่ภายในบัญชี Firebase และการให้สิทธิ์เข้าถึงบริการและทรัพยากรของ Firebase กับผู้ใช้ที่จำเป็นต้องใช้ทำงานเท่านั้น Firebase Security เป็นการรักษาความปลอดภัยที่ใช้การสร้าง Key และออก Certificate ในการทำงานของระบบ

Firebase IoT [10] เป็น Service ที่ ได้ให้บริการ ทำงานบนฐานของ MQTT สำหรับ Firebase IoT จะมีระบบความปลอดภัยที่สูงมาก โดย Firebase IoT จะรองรับ Platform เพียง 2 อย่างเท่านั้นคือ Linux/OSX และ Windows เท่านั้น และสำหรับ Firebase มันจะมีตัว SDK ของมันเอง ซึ่ง SDK ที่ให้บริการกับ Firebase IoT มีเพียง 3 ตัวเท่านั้นคือ Node.js Python และ Java



ภาพที่ 2.24 การทำงานและองค์ประกอบของ Console Firebase Google Cloud Website Service

IoT

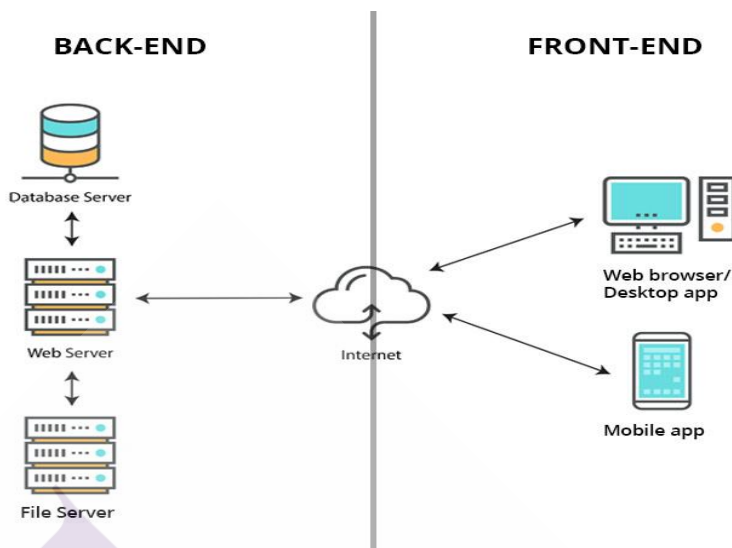


ภาพที่ 2.25 การทำงานและองค์ประกอบของ Console Firebase Google Cloud Website Service IoT

Firebase คือ โปรเจกต์ที่ถูกออกแบบมาให้เป็น API และ Cloud Storage สำหรับพัฒนา Real time Application รองรับหลายแพลตฟอร์มทั้ง iOS App, Android App, Web App.

Firebase ถูกสร้างขึ้นจากคุณสมบัติเสริมว่านักพัฒนาสามารถผสมและจับคู่ เพื่อให้พอดีกับความต้องการ ของคน บริษัท ก่อตั้งขึ้นในปี 2011 โดยแอนดรูว์และเจมส์ เทมปลิน สินค้าเริ่มต้น Firebase เป็นฐานข้อมูลเรียลไทม์ซึ่งมี API ที่ช่วยนักพัฒนาในการจัดเก็บและซิงค์ข้อมูล โดย Google Firebase 2.0 กูเกิลได้ซื้อกิจการ Firebase และมีการพัฒนาให้สามารถจากบริการ backend เก็บข้อมูลอย่างเดียว มาเป็นแพลตฟอร์ม ครอบคลุมสำหรับนักพัฒนาแอปพลิเคชันเพื่อรองรับบริการแทบทุกอย่างที่นักพัฒนาแอปต้องใช้งาน

Firebase เหมาะกับใคร เหมาะกับคนที่ทำ Front-end (โดยเฉพาะ Mobile App) แล้วอยากเชื่อมต่อกับ Back-end แต่ไม่อยากทำ Back-end เอง เช่น เซิร์ฟเวอร์ คิดตั้งโปรแกรม ตั้งค่าการเชื่อมต่อต่างๆ และในเรื่องของปลอดภัย เป็นต้น ซึ่ง Firebase ได้จัดการสิ่งเหล่านี้ให้เรียบร้อย และยังมีความสามารถอื่นๆอีกมากมายที่ช่วยเพิ่มประสิทธิภาพและคุณภาพให้กับงาน โดยเราอาจไม่คาดคิด เช่น รายงานข้อผิดพลาด ข้อมูลสถิติผู้ใช้งาน ระบบทดสอบ เป็นต้น



ภาพที่ 2.26 โครงสร้างการทำหน้าที่ Back-End และ Front-End ของ Console Firebase Cloud Website Service

ผลิตภัณฑ์ทั้งหมดของ Firebase มี 18 อย่างและแบ่งออกเป็น 3 หมวดดังนี้ Build better apps, Improve apps quality, Grow your business.

1. Build better apps มีทั้งหมด 7 รูปแบบได้แก่

- Realtime Database คือบริการฐานข้อมูล No SQL ใช้วิธีเก็บข้อมูลเป็น JASON Tree ขนาดใหญ่ และสามารถ Sync สถานะข้าม Client ได้แบบ Real time กล่าวคือ หากเชื่อมต่อ Database เดียวกัน 2 ที่ เมื่อใดที่หนึ่งมีการอัปเดตข้อมูล อีกที่หนึ่งก็จะมีการอัปเดตข้อมูลให้เหมือนกันโดยอัตโนมัติและสามารถทำงานแบบ Offline ได้บนแอป Android และ iOS ได้เช่นกัน

- Authentication คือการบริการตรวจสอบผู้ใช้โดยสามารถตรวจสอบได้หลายวิธี เช่น Email/Password, เบอร์โทรศัพท์บัญชี Google ,Facebook, Twitter, Github เป็นต้น มีฐานข้อมูลเป็นของตัวเองไม่ต้องสร้างใหม่หรือออกแบบวิธีการเก็บ ซึ่งสามารถดูได้ว่าสมัครด้วยวิธีไหนสมัครเมื่อไหร่ และเข้าใช้ระบบครั้งสุดท้ายเมื่อไหร่

- Hosting คือให้บริการฝากไฟล์ static เช่น HTML, CSS, JS, JPG (ไม่รองรับ PHP ซึ่งเป็น Dynamic) เพื่อให้คนอื่นๆ เข้าใช้งานเว็บของเราได้ มักนิยมใช้ในการฝากไฟล์ที่ได้จากการ Build ของ JavaScript Framework ต่างๆ เช่น Angular, React, Vue สังเกตว่าจะได้ไฟล์ HTML, CSS, JS ต่างๆ ตามที่ได้แจ้งไว้ข้างต้น หรือจะเป็ไฟล์ที่เขียนเองก็ได้ ไม่จำเป็นต้องใช้ Framework ก็ได้เหมือนกัน อีกทั้งมี CDN และ SSL มาด้วยแบบฟรีๆ เพื่อให้ผู้ใช้ของคุณได้รับประสบการณ์การ

ใช้งานที่ปลอดภัยเชื่อถือได้และไม่มีความล่าช้าแม้ว่าจะอยู่ที่ไหนก็ตาม (ทุกเว็บมี Domain Name ของ Firebase ให้อัปเดตโน้ตแต่เปลี่ยนมาใช้ของตัวเองได้)

- Cloud Functions คือบริการสำหรับ Deploy Function ที่พัฒนาด้วย JavaScript หรือ Type Script เพื่อทำงานตาม Tigger (คล้ายๆ event) ที่เกิดขึ้นบน Firebase เช่น ถ้า Database ถูกเขียน (Real time Database Triggers) ให้ Function เราส่ง Notification แจ้งไปบอกเราด้วยหรือ มีการเรียกมาที่ HTTP Endpoint (HTTP Triggers) ให้ Function เราคืนค่า HTML กลับไป (ใช้ทำ REST API) หรือถ้าแอปมีปัญหา (Crashlytics Triggers) ให้ส่งข้อความแจ้งเตือนไปที่ Slack ในส่วนนี้การใช้งานฟรีจะไม่สามารถเชื่อมต่อไปยัง API ช่างนอกที่ไม่ใช่ Google ได้

- Cloud Storage คือบริการเก็บไฟล์รูปภาพ, ไฟล์เสียง, วิดีโอ เพื่อใช้บน Application เช่น รูปภาพประจำตัวสมาชิก, วิดีโอสอนการใช้งานโปรแกรม เป็นต้น

- Cloud Fire Store (Beta) คือ Realtime Database รุ่นใหม่มาพร้อมการค้นหาและการปรับขนาดอัตโนมัติที่มีประสิทธิภาพมากขึ้น ปรับปรุงวิธีการเก็บข้อมูลใหม่เป็น Collections และสามารถทำงานแบบ Offline บน Web ได้อีกด้วย ซึ่งจากเดิมใช้ได้แค่บน Android และ iOS

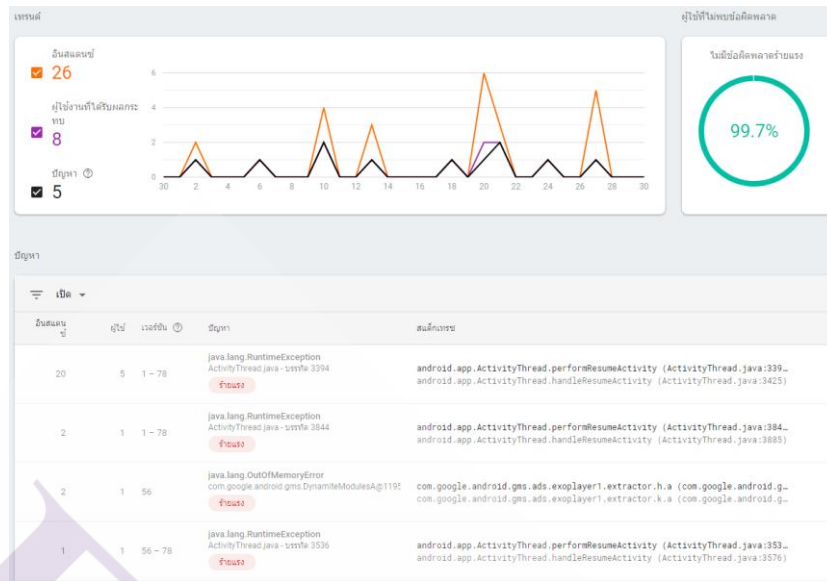
- ML Kit (Beta) คือ Machine Learning SDK ที่ช่วยให้แอปมือถือสามารถใช้ความสามารถของ ML ได้ง่ายยิ่งขึ้น สามารถทำงานได้ทั้งแบบ Online และ Offline ดังตารางนี้

Feature	On-device	Cloud
Text recognition	✓	✓
Face detection	✓	
Barcode scanning	✓	
Image labeling	✓	✓
Landmark recognition		✓
Custom model inference	✓	

ภาพที่ 2.27 ตารางระบุความสามารถที่ทำงานแบบ Offline (On-device) และ Online (Cloud)

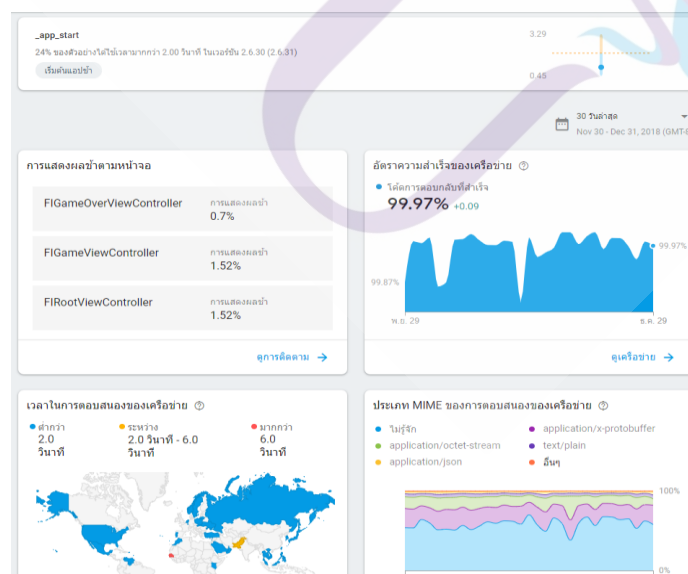
2. Improve apps quality มีทั้งหมด 3 รูปแบบได้แก่

- Crashlytics คือ บริการตรวจจับและแจ้งเตือนหากแอปเราเกิดอาการ Crash ขึ้นแบบ Real time เพื่อให้แอปเราเสถียรอยู่เสมอ



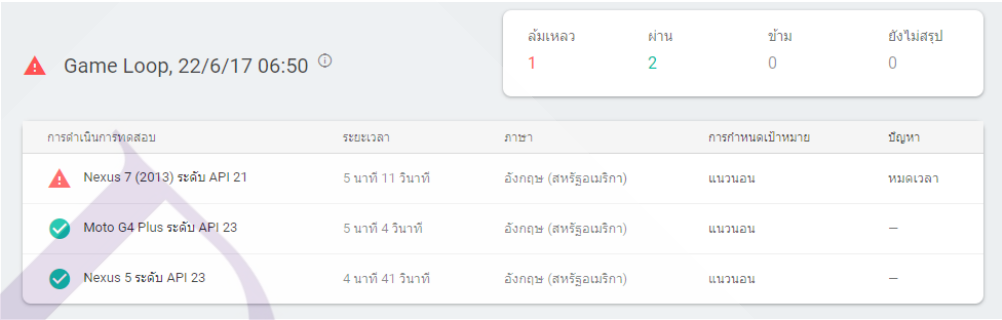
ภาพที่ 2.28 รูปแบบของการผิดพลาด

- Performance Monitoring คือ บริการตรวจสอบคุณภาพของแอปพลิเคชัน เพื่อให้แอปของเราตอบสนองได้เร็วอยู่เสมอ โดยสามารถตรวจสอบเวลาและรายละเอียดการทำงานต่างๆ เช่น เวลาที่ใช้ในการเปิดแอป, เวลาที่ใช้การเปลี่ยนหน้า UI, เวลาที่ใช้ในการโหลด API, ขนาดข้อมูลที่ Download/Upload, จำนวน API ที่สำเร็จหรือล้มเหลว เป็นต้น



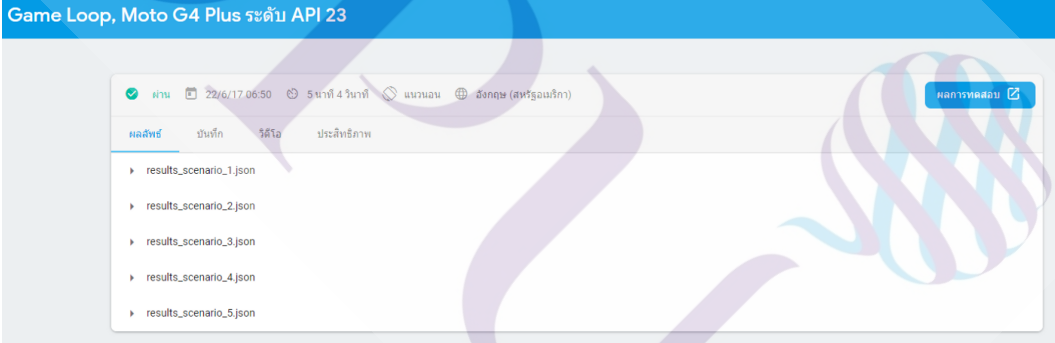
ภาพที่ 2.29 รูปแบบของการรายงานประสิทธิภาพ

- Test Lab คือบริการทดสอบแอปบนฮาร์ดแวร์จริงๆ เพื่อให้มั่นใจว่าแอปของเราสามารถรองรับฮาร์ดแวร์ที่เราต้องการได้จริงๆ โดยสามารถระบุรุ่นและเวอร์ชันที่ต้องการได้ แล้วระบุรูปแบบการทดสอบต่างๆ เพื่อทดสอบและรายงานผลกลับมาไม่ต้องซื้อโทรศัพท์เอง (สมมุติว่าจริงจริงเรื่องการรองรับทุกอุปกรณ์มาก) ซึ่งเป็นเรื่องยากด้วยหากจะซื้อทุกรุ่นที่คนนิยมใช้ในตลาดไหนจะต่อสาย จะนั่งทดสอบทีละเครื่องอีกใช้ตัวนี้จบหมดปัญหา



การดำเนินการทดสอบ	ระยะเวลา	ภาษา	การกำหนดเป้าหมาย	ปัญหา
⚠️ Nexus 7 (2013) ระดับ API 21	5 นาที 11 วินาที	อังกฤษ (สหรัฐอเมริกา)	แนวนอน	หมดเวลา
✅ Moto G4 Plus ระดับ API 23	5 นาที 4 วินาที	อังกฤษ (สหรัฐอเมริกา)	แนวนอน	-
✅ Nexus 5 ระดับ API 23	4 นาที 41 วินาที	อังกฤษ (สหรัฐอเมริกา)	แนวนอน	-

ภาพที่ 2.30 การทดสอบอุปกรณ์ต่างๆ

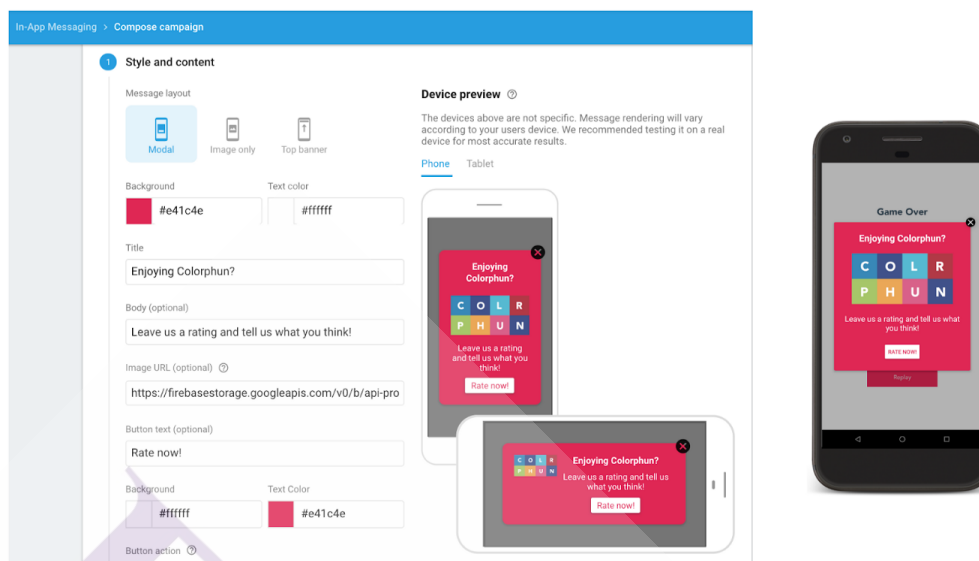


ผลการทดสอบ	บันทึก	วิดีโอ	ประวัติภาพ
▶️ results_scenario_1.json			
▶️ results_scenario_2.json			
▶️ results_scenario_3.json			
▶️ results_scenario_4.json			
▶️ results_scenario_5.json			

ภาพที่ 2.31 รูปแบบผลการทดสอบ

3. Grow your business มีทั้งหมด 8 รูปแบบได้แก่

- In-App Messaging คือบริการแสดงข้อความ pop-up ภายในแอปของเรา เช่น โฆษณา การแจ้งเตือน ข่าวสาร เป็นต้น



ภาพที่ 2.32 การแสดงผล In-App Messaging

- Google Analytics คือบริการแสดงข้อมูลสถิติต่างๆ ของแอป เช่น ใช้ด้วยระบบปฏิบัติการอะไร จำนวนเท่าไร มีผู้ใช้งาน ณ ปัจจุบันกี่คน ใช้งานส่วนไหนบ้าง เป็นต้น เพื่อวิเคราะห์กลุ่มเป้าหมายหรือรับทราบพฤติกรรมของผู้ใช้งานต่างๆ
- Predictions คือบริการวิเคราะห์ข้อมูลการใช้งานแอป ช่วยให้เรารู้ว่าผู้ใช้งานส่วนใดบ้างในแอป ช่วยให้เรารู้ว่าส่วนใดตอบสนองได้ดี ส่วนใดควรปรับปรุง หรืออาจต้องการที่จะหยุดรู้พฤติกรรมในอนาคตของผู้ใช้งานแอปของคุณ
- Cloud Messaging คือบริการส่งการแจ้งเตือนไปยังมือถือหรือเว็บของเราเพื่อแจ้งข้อความไปยังผู้ใช้ของเราแม้ว่าจะปิดแอปไปแล้วก็ตาม ถ้าใครใช้ Smartphone อยู่ จะคุ้นเคยกันเป็นอย่างดี เช่น การแจ้งเตือนจาก Facebook, Line, Instagram ต่างๆ เป็นต้น
- Remote Config คือสามารถที่จะเปลี่ยนลักษณะการทำงานและลักษณะที่ปรากฏของแอปของคุณได้ทันทีจากหน้าเว็บ Firebase โดยไม่ต้องรอการอนุมัติจาก App Store เช่น การเปลี่ยนรูปแบบตามเทศกาล เปลี่ยนภาษาตามผู้ใช้งาน เป็นต้น
- Dynamic Links คือลิงค์เชื่อมโยงไปยังแอปมือถือ ใช้สำหรับแสดงบนหน้าเว็บเพื่อให้ผู้ใช้งานติดตั้งแอปมือถือผ่านลิงค์ๆนี้ อีกทั้งยังสามารถแนบข้อมูลต่างๆ ของผู้ใช้ที่อยู่บนเว็บมาด้วยได้

- App Indexing คือการปรับแต่งแอปของเราให้แสดงผลข้อมูลภายในแอปบน Google Search ได้ หรือเรียกว่าการทำ SEO แบบ Mobile App ก็ได้ เช่น คั้นซื้อร้านอาหารแล้วปรากฏแอป Wongnai ขึ้นมาให้ดูรายละเอียดและรีวิว เป็นต้น

- A/B Testing (Beta) คือความสามารถในการแสดงผลแอปหลายรูปแบบเพื่อทดสอบ การแสดงผล หรือการทำงาน ว่าสิ่งไหนจะมอบประสบการณ์การใช้งานที่ดีกว่าให้แก่ผู้ใช้งาน เช่น การวางปุ่มกดแบบไหน ที่ผู้ใช้งานใช้สะดวก สมมติว่ามีผู้ใช้งาน 100 คน อาจจะมี 50 คนได้ปุ่มที่อยู่มุมบน อีก 50 คน ได้ปุ่มที่อยู่มุมล่าง หากว่ามีการใช้งานแบบไหนมากกว่ากันก็อาจจะสรุปผลและเลือกใช้แบบนั้นกับทุกคนในท้ายที่สุด

ในตอนนี Firebase รองรับ Android, iOS, Web, Unity และ C++ แต่ไม่ได้รองรับทุกภาษาหรือระบบในทุกผลิตภัณฑ์ ซึ่งจากทั้งหมด 18 ผลิตภัณฑ์มีสัดส่วนรองรับ ดังนี้

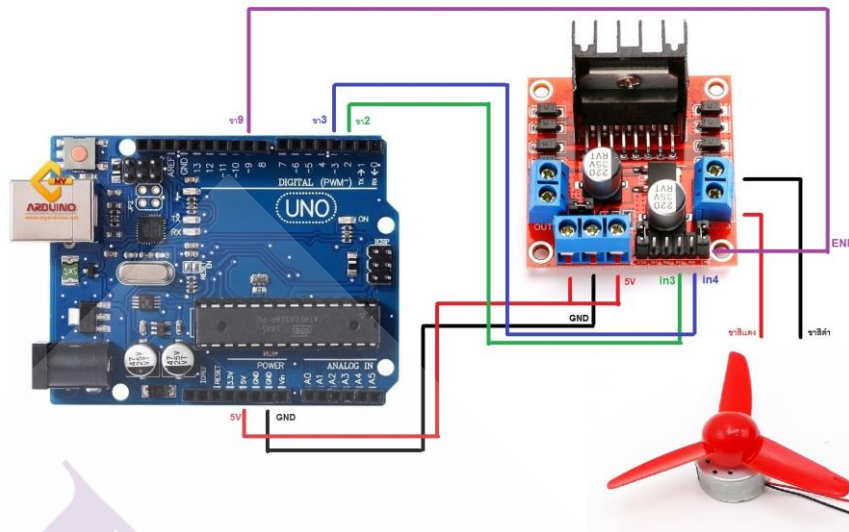
รองรับ Android และ iOS 17 ผลิตภัณฑ์ ยกเว้น Hosting

รองรับ Web 8 ผลิตภัณฑ์ ได้แก่ Cloud Firestore, Cloud Functions, Authentication, Hosting, Cloud Storage, Real time, Database, Cloud Messaging, Dynamic Links.

รองรับ C++ และ Unity 10 ผลิตภัณฑ์ ได้แก่ Cloud Functions, Authentication, Cloud Storage, Real time, Database, Google Analytics, Predictions, A/B Testing, Cloud Messaging, Remote Config, Dynamic Links

2.6.5 Controlling device

Controlling device เป็นอุปกรณ์ที่ใช้กับติดต่อกับผู้ใช้โดยตรง User interface เพื่อควบคุมหรือเรียกดูข้อมูลได้อย่างสะดวก โดยในปัจจุบันอุปกรณ์ Smartphone จัดเป็น Controlling device ที่ได้รับความนิยมสูงสุดเพราะความสะดวกในการพกพาและการใช้งาน ดังนั้นการออกแบบในส่วนของ User interface ให้กับระบบ IoT ใดๆควรให้ความสำคัญ นอกจากนี้ Controlling device ยังรวมถึงส่วนแสดงผลการทำงานในรูปแบบอื่นๆด้วย เช่น Table และ Computer เป็นต้น 1602 Character LCD Display with I2C Module [16] ถือว่าเป็นส่วนหนึ่งของ Controlling device ในส่วนของการแสดงผล เป็นจอที่แสดงผลเป็นตัวอักษรตามช่องแบบตายตัวขนาด 16x2 หมายถึงใน 1 แถว มีตัวอักษรใส่ได้ 16 ตัว และมีทั้งหมด 2 บรรทัดให้ใช้งาน การเชื่อมต่อแบบอนุกรม เป็นการเชื่อมต่อกับจอ LCD ผ่านโมดูลแปลงรูปแบบการเชื่อมต่อกับจอ LCD จากแบบขนาน มาเป็นการเชื่อมต่อแบบอื่นที่ใช้สายน้อยกว่า เช่น การใช้โมดูล I2C Serial Interface จะเป็นการนำโมดูลเชื่อมเข้ากับตัวจอ LCD แล้วใช้บอร์ด Arduino เชื่อมต่อกับบอร์ดโมดูลผ่านโปรโตคอล I2C ทำให้ใช้สายเพียง 4 เส้น ก็ทำให้หน้าจอแสดงผลข้อความต่างๆ ออกมาได้ช่วยเพิ่มความสะดวกต่อการใช้งาน



ภาพที่ 2.33 การเชื่อมต่อพัดลมกับ Arduino Uno R3

จากภาพที่ 2.33 แสดงการรับส่งข้อมูลแบบ I2C BUS [16] โดย MCU จะทำการส่งสถานะเริ่มต้น (START Conditions) เพื่อแสดงการขอใช้บัสแล้วตามด้วยรหัสควบคุม (Control Byte) ซึ่งประกอบด้วยรหัสประจำตัวอุปกรณ์ Device ID, Device Address และ Mode ในการเขียนหรืออ่านข้อมูล เมื่ออุปกรณ์รับทราบว่ามี MCU ต้องการจะติดต่อกับ บัส ก็ต้องส่งสถานะรับรู้ (Acknowledge) หรือแจ้งให้ MCU รับรู้ว่าข้อมูลที่ส่งมามีความถูกต้อง และเมื่อสิ้นสุดการส่งข้อมูล MCU จะต้องส่งสถานะสิ้นสุด (STOP Conditions) เพื่อบอกกับอุปกรณ์ว่าสิ้นสุดการใช้บัส



ภาพที่ 2.34 การรับส่งข้อมูลแบบ I2C BUS

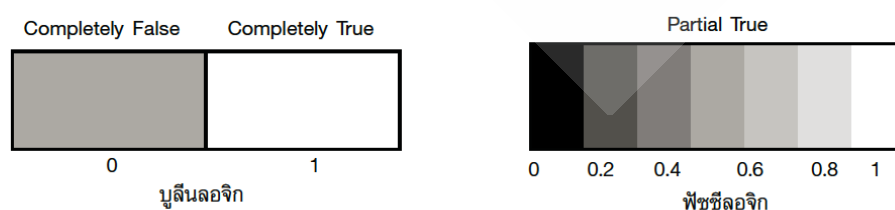
2.7 การนำระบบ Fuzzy logic เข้ามาช่วยในการจัดการ

หากจะถามว่าฟัซซีลอจิก (Fuzzy Logic) คืออะไร ในที่นี้จะขอแยกออกเป็น 2 คำหลักคือ ฟัซซี และลอจิก คำว่า “ฟัซซี (Fuzzy)” ในความหมายโดยทั่วไปก็คือ คลุมเครือ และถ้าหากจะถามต่อว่าความคลุมเครือคืออะไร ในโลกแห่งความเป็นจริงความไม่แน่นอน (Uncertainty)

ซึ่งหมายถึงสถานะภาพของความลังเล สงสัย ไม่แน่ใจ หรือไม่มั่นใจ โดยความไม่แน่นอนดังกล่าว อาจเกิดมาจากหลายปัจจัย เช่น ข้อมูลไม่ครบถ้วน มีเพียงบางส่วน ขาดหลักฐานยืนยันความ น่าเชื่อถือที่จะสร้างความมั่นใจในตัวข้อมูลได้การใช้ภาษามีรูปแบบที่เข้าใจยากหรือ ดี ความได้ยาก ข้อมูลจากแหล่งอ้างอิงแตกต่างกันจนเกิดความขัดแย้งไม่สามารถชี้แจงข้อมูลด้วยเหตุและผลได้ อย่างชัดเจน ข้อมูลอาจได้มาจากการประมาณการ แหล่งที่มาของข้อมูลขาดความน่าเชื่อถือ ซึ่งจากที่ กล่าวมาทั้งหมดเหล่านี้ก็คือความคลุมเครือหรือฟัซซีนั่นเอง ส่วนลอจิก (Logic) หรือตรรกศาสตร์ มีรากศัพท์มาจากคำว่า (Logos) ในภาษากรีก โดยทั่วไปประกอบ ด้วยการศึกษารูปแบบ ของการ ได้แย้งอย่างเป็นระบบ ข้อโต้แย้งที่สมเหตุสมผล คือข้อโต้แย้งที่มีความสัมพันธ์ของการสนับสนุน เซึ่งตรรกะที่เฉพาะเจาะจงระหว่างข้อสมมติฐานของ ข้อความโต้แย้งและข้อสรุป ตรรกศาสตร์เป็น การศึกษาเชิงปรัชญาว่าด้วยการให้เหตุผล โดยมักจะเป็นส่วนสำคัญของวิชาปรัชญา คณิตศาสตร์ คอมพิวเตอร์ รวมถึงภาษาศาสตร์ ตรรกศาสตร์เป็นการตรวจสอบข้อโต้แย้งที่สมเหตุสมผล (Valid Argument) จากที่กล่าวมาในข้างต้นก็พอที่จะสรุปได้ว่า ฟัซซีลอจิกก็คือ ตรรกศาสตร์คลุมเครือ และ วัตถุประสงค์ หลักของการพัฒนาฟัซซีลอจิกในเบื้องต้นก็เพื่อนำมาในเป็นเครื่องมือที่ช่วยในการ ตัดสินใจภายใต้ความไม่แน่นอนของข้อมูลโดยยอมให้มีความยืดหยุ่นได้ มีการใช้หลักเหตุผลที่คล้าย การเขียนแบบวิชีคิด และตัดสินใจที่ซับซ้อนของมนุษย์ ดังนั้นฟัซซีลอจิกจึงถือได้ว่าเป็นส่วนหนึ่ง ของระบบ AI (Artificial Intelligence) หรือระบบปัญญาประดิษฐ์นั่นเอง

ฟัซซีลอจิก กับ บูลีนลอจิก

แล้วบูลีนลอจิก(Boolean Logic) หรือตรรกะแบบดั้งเดิม(Classical Logic) หรือบางครั้ง อาจเรียกว่า คริสป์ลอจิก (Crisp Logic) กับฟัซซีลอจิก (Fuzzy Logic) มีความแตกต่างกันอย่างไร ตรรกะแบบดั้งเดิม หรือบูลีนลอจิก นั้นเป็นข้อสรุปแบบที่มีแค่ “จริง(True)” กับ “เท็จ (False)” หรือ “1” กับ “0” เท่านั้น แต่ฟัซซีลอจิกมีลักษณะที่พิเศษกว่าบูลีนลอจิก เป็นแนวคิดที่มีการต่อขยายใน ส่วนของความจริง (Partial True) โดยค่าความเป็นจริงจะอยู่ระหว่างจริง (1) กับเท็จ (0) ดังแสดงใน รูปที่ 2.35

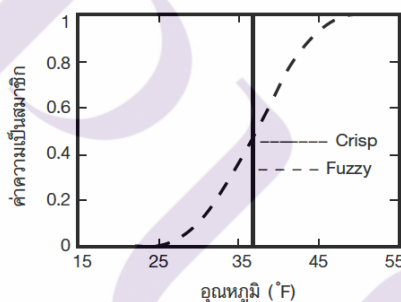


ภาพที่ 2.35 แสดงบูลีนลอจิกและฟัซซีลอจิก

จากรูปที่ 2.35 หากเราสังเกตก็จะพบว่าอุณหภูมิโลกจะมีค่าแค่ 0 กับ 1 แต่ฟัซซีลอจิกสามารถมีค่าได้ตั้งแต่ 0 ถึง 1 นั้นหมายความว่า อาจมีค่าเป็น 0.1, 0.2, 0.5 หรือ 0.8 เป็นต้น ซึ่งการมีค่ามากกว่า 2 ค่าจะถูกเรียกว่า “ความเป็นฟัซซี (Fuzziness)”

ฟัซซีเซต

“เซต (Set)” เป็นคำที่ไม่ต้องนิยามความหมาย เราใช้เซตแทนกลุ่มของคน สัตว์ สิ่งของ หรือสิ่งต่างๆ ที่อยู่รวมกันเป็นกลุ่ม นอกจากนี้สิ่งที่อยู่ในเซตที่เรียกว่า สมาชิก (Membership) จะต้องมีการระบุคุณสมบัติที่ชัดเจน แล้วฟัซซีเซต (Fuzzy Set) คืออะไร ฟัซซีเซตเป็นเครื่องมือทางคณิตศาสตร์ที่สื่อถึงความไม่แน่นอน ที่สามารถจะสร้างและกำหนดรูปแบบ (Modeling) ของลักษณะ ความไม่แน่นอนที่เป็นความคลุมเคลือ ความไม่ตายตัว รวมถึงข้อมูลที่ไม่สมบูรณ์ อยู่ในรูปของเซต โดยทฤษฎีของฟัซซีเซตจะใช้ลักษณะความหมายตัวแปรภาษา (Linguistic) มากกว่า ปริมาณ (Quantitative) ของตัวแปร ซึ่งแตกต่างจากเซตแบบดั้งเดิม (Classical Set) หรือเซตวินัย (Crisp Set) ที่เป็นเซต ที่มีค่าความเป็นสมาชิกตามแนวคิดเลขฐานสอง โดยที่ตัวแปรหนึ่งๆ จะมีค่าเพียง 2 ค่าคือ 0 ไม่เป็นสมาชิก หรือ 1 เป็นสมาชิก เท่านั้น

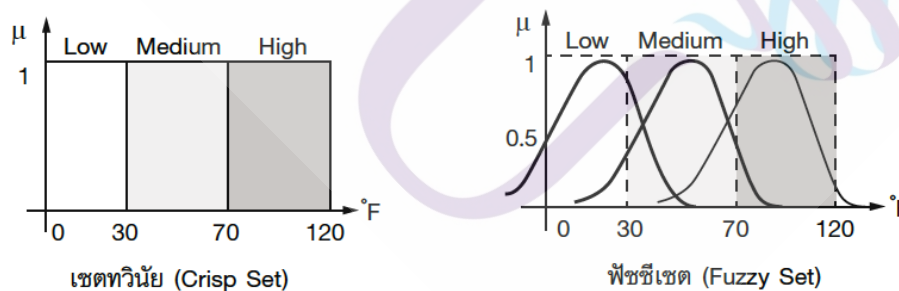


อุณหภูมิ “ปานกลาง” °F (°C)	15 (-9)	20 (-6)	25 (-4)	30 (-1)	35 (1.6)	40 (4)	45 (7)	50 (10)
Crisp (ค่าความเป็นสมาชิก)	0	0	0	0	0	0	1	1
Fuzzy (ค่าความเป็นสมาชิก)	0	0	0	0.1	0.3	0.6	0.8	1

ภาพที่ 2.36 ตัวอย่างอุณหภูมิปานกลางของเซตวินัยและเซตของฟัซซี

ดังนั้นฟัซซีเซตจึงเป็นเซตที่มีขอบเขตที่ราบเรียบและครอบคลุมเซตแบบดั้งเดิม โดยฟัซซีเซตยอมให้มีค่าความเป็นสมาชิก (μ) ของเซตอยู่ระหว่าง “0” และ “1” ซึ่งในโลกแห่งความเป็นจริง เซตไม่ได้มีเฉพาะเซตแบบดั้งเดิมเท่านั้น แต่จะมีฟัซซีด้วย โดยฟัซซีเซตจะมีขอบเขตที่ไม่ใช่การเปลี่ยนแปลง ทันทีทันใดจาก 0 เป็น 1 ยกตัวอย่างความแตกต่าง ของฟัซซีเซตแบบวินัย

เกี่ยวกับอุณหภูมิ เช่น อุณหภูมิปานกลาง ดังแสดงในรูปที่ 2.36 เขตทวินัย ค่าอุณหภูมิตั้งแต่ 15 – 37 °F ((-9) – 3 °C) ค่าความเป็นสมาชิกจะมีค่าเท่ากับ 0 และอุณหภูมิตั้งแต่ 37 – 55 °F (3 – 13 °C) ค่าความเป็นสมาชิกจะมีค่าเท่ากับ 1 สำหรับในกรณีของฟัซซีเซต อุณหภูมิตั้งแต่ 15 – 25 °F ((-9) – (-3) °C) 0 แต่หลังจากนั้น ค่าความเป็นสมาชิกจะค่อยๆ เพิ่มขึ้น ดังแสดงค่าในตาราง เช่น อุณหภูมิ 30 °F มีค่าเท่ากับ 0.1 ที่ 35 °F (1.6 °C) มีค่าเท่ากับ 0.3 และเป็น 1 ที่อุณหภูมิตั้งแต่ 50 °F (10 °C) เป็นต้น ไป เพื่อแสดงให้เห็นความแตกต่างของทั้งเขตทวินัยและฟัซซีเซตที่มีหลายเซต ดังนั้นในที่นี้จะขอแบ่งอุณหภูมิโดยทั่วไปเป็นเซตต่างๆ ดังแสดงด้วยรูป 2.28 กล่าวคือ อุณหภูมิตั้งแต่ 0 – 30 °F ((-17) – (-1) °C) กำหนดให้เป็นอุณหภูมิต่ำ (Low) ของเขตทวินัยโดยค่าความเป็นสมาชิกของเซตในช่วงดังกล่าวเป็น 1 นอกนั้นเป็น 0 แต่ในส่วนฟัซซีเซตกำหนดช่วงอุณหภูมิต่ำ เป็น (-15) – 45 °F ((-26) – -7 °C) แล้วค่าความเป็นสมาชิกของเซตอยู่ระหว่าง 0 กับ 1 เช่นที่อุณหภูมิ 15 °F (-9.4 °C) ค่าความเป็นสมาชิกเท่ากับ 0.5 และอุณหภูมิ 25 °F (3.8 °C) ค่าความเป็นสมาชิกเท่ากับ 1 เป็นต้น อุณหภูมิปานกลาง (Medium) ของเขตทวินัยคืออุณหภูมิตั้งแต่ 30 – 70 °F ((-1) – 21 °C) โดยค่าความเป็นสมาชิกของเซตในช่วงดังกล่าวเป็น 1 นอกนั้นเป็น 0 ส่วนฟัซซีเซตจะกำหนดช่วงอุณหภูมิต่ำ ปานกลาง ไว้ที่ 15 – 80 °F ((-9) – 27 °C) และหากเราพิจารณาจากรูป ก็จะพบว่าในกรณีของฟัซซีเซต เซตของอุณหภูมิต่ำ ปานกลางกับอุณหภูมิต่ำ มีการซ้อนทับกันอยู่ (Overlap) ซึ่งเป็นการสื่อแสดงถึงความคลุมเครือให้ความสำคัญต่อตัวแปรภาษามากกว่าตัวแปรปริมาณ สำหรับช่วงของอุณหภูมิสูง (High) ของทั้งเขตทวินัยและฟัซซีเซตก็จะคล้ายกับการอธิบายที่ผ่านมาดังแสดงในรูปที่ 2.37



ภาพที่ 2.37 การกำหนดค่าความเป็นสมาชิกของเขตทวินัยและฟัซซีเซต

2.8 ระบบปรับปรุงสภาพอากาศภายในอาคารแบบพลวัตโดยใช้ IoT เข้ามาช่วย

ระบบที่ควบคุมการปรับปรุงสภาพอากาศภายในอาคาร ระบบฝังตัวอัจฉริยะ [5] จะทำการปรับปรุง สภาพอากาศให้ดีขึ้นด้วยระบบอัตโนมัติ แบบ IoT มีการทำงานที่ออกเป็น 4 ส่วน ได้แก่

ส่วนรับค่า ทำหน้าที่รับค่าจากเซ็นเซอร์ต่างๆ ไม่ว่าจะเป็นเซ็นเซอร์วัดปริมาณฝุ่นละออง และเซ็นเซอร์ วัดอุณหภูมิ

ส่วนปรับค่า ทำหน้าที่รับคำสั่งจากส่วนประมวลผลเพื่อปรับอุปกรณ์ที่เชื่อมต่อกับ สวิตซ์อิเล็กทรอนิกส์ ดังนี้ มอเตอร์กระแสตรงแบบพัดลมดูดอากาศและกรองอากาศ Filter PM 2.5 พัดลมระบายอากาศ และ เครื่องตรวจวัดฝุ่นละออง PM 2.5

ส่วนควบคุม ทำหน้าที่ประมวลผลจากส่วนรับค่าส่งคำสั่งไปยังส่วนปรับค่าแสดงผลไปยังส่วน ติดต่อผู้ใช้ (User Interface) และชุดท้าย

ส่วนเก็บบันทึกข้อมูล (Data Logger) ทำหน้าที่บันทึกข้อมูล อุณหภูมิ,ขนาดของปริมาณ ฝุ่นละออง PM 2.5 และเหตุการณ์ที่เกิดขึ้นในอุปกรณ์ปรับค่าดังนี้ พัดลมระบายอากาศ ที่ช่องนำเข้า อากาศ และพัดลมระบายอากาศช่องฝ้าเพดาน และพัดลมระบายอากาศห้องโถงหลังคา พบว่าระบบ สามารถจัดการสภาพอากาศภายในอาคาร ได้ดี ไม่ว่าจะเป็นสภาพอากาศร้อน จัดระบบก็จะพยายาม ปรับเพื่อที่อุณหภูมิจะลดลง 2-5 องศาเซลเซียส เพื่อให้มีผลเสียต่อสภาพอากาศที่ดีขึ้น ระดับการ ไหลเวียนอากาศดีขึ้น จึงทำให้สามารถแบ่งเป็นกลุ่มตามขนาดของการใช้งานระบบ ได้ดังนี้

2.8.1 ระบบปรับปรุงสภาพอากาศภายในอาคารขนาดเล็ก ประเภทสิ่งปลูกสร้าง ที่พักอาศัย ประจำวัน การใช้งานประเภทนี้ ส่วนใหญ่มักเป็นการใช้งานแบบปกติคือไม่มีระบบเข้ามาควบคุม ในการทำงานของระบบระบายอากาศทั้งแบบปกติหรือจะใช้ระบบ IoT ก็ได้ ประสิทธิภาพอาจได้ ไม่คงที่ ในกรณีที่ใช้ระบบแบบธรรมดา จึงได้คิดที่จะนำระบบ IoT เข้ามาช่วย ส่วนใหญ่ระบบ มักจะมีราคาแพง แต่ถ้าหากสามารถลดต้นทุนของอุปกรณ์ที่นำมาใช้และอำนวยความสะดวก สำหรับพื้นที่ที่พักอาศัย ด้วยการ ติดตั้งระบบดังกล่าว จะเป็นการช่วยส่งเสริมการใช้งานและเกิด การประหยัดพลังงานได้เป็นอย่างดี

2.8.2 ระบบปรับปรุงสภาพอากาศภายในอาคารขนาดกลาง เช่น สำนักงาน ออฟฟิศ ที่มีผู้ ปฏิบัติงานตลอดเกือบทั้งวัน ทั้งในเวลากลางวันและเวลากลางคืนในช่วงเวลาพิเศษของการ ปฏิบัติงาน การนำเทคโนโลยีเข้ามาช่วยแก้ไขในจุดนี้สามารถช่วยได้เล็กน้อยเพื่อเพิ่มประสิทธิภาพ ของการทำงานและสามารถทำให้ทำงานได้มีระยะเวลาที่ยาวนานมากขึ้น

2.8.3 ระบบปรับปรุงสภาพอากาศภายในอาคารขนาดใหญ่ระดับอาคารขนาดใหญ่ ในการนำ เทคโนโลยี IoT เข้ามาใช้ นั้น ส่วนใหญ่มักถูกออกแบบมาใช้งานสำหรับ โรงงานอุตสาหกรรมขนาด

เล็ก และขนาดใหญ่ นิยมใช้มากในประเทศ และในต่างประเทศ โดยเฉพาะประเทศอากาศร้อนจัด หรือพื้นที่แห้งแล้ง เพราะจะสามารถใช้ให้เกิดประโยชน์และคุ้มค่าในอาคาร สามารถควบคุมคุณภาพอากาศ และ ปริมาณฝุ่นละออง PM2.5 ได้อย่างดีแต่จำเป็นต้องใช้เงินลงทุนจำนวนมาก สำหรับระบบที่มีขนาดใหญ่ แต่ จะช่วยเพิ่มประสิทธิภาพของอากาศที่มีความจำเป็นต่อการรักษาคุณภาพของสินค้าที่ผลิตและอายุการใช้งานที่ยาวนาน

2.9 สรุปเนื้อหา

เนื้อหาในบทนี้ได้กล่าวถึงองค์ประกอบของงานวิจัยในส่วนการศึกษาค้นคว้า เกี่ยวกับ ทฤษฎีที่เกี่ยวข้อง ข้องกับระบบ อุปกรณ์ที่เลือกใช้ภายในระบบ หลักการทำงานของอุปกรณ์ในแต่ละ ส่วน ทฤษฎีและขั้นตอนการทำงานเกี่ยวกับระบบปรับปรุงสภาพอากาศภายในอาคาร ส่วนประกอบ ระบบที่ต้องใช้ แนวทางการประยุกต์ใช้เทคโนโลยี อินเทอร์เน็ตสรรพสิ่ง IoT หลักการทำงานและการเข้าใช้งานของ Console Firebase Google Cloud Website Service การเข้าถึงการใช้งานระบบ จัดการระบบและระบบจัดเก็บข้อมูล

บทที่ 3

การวางแผนและการออกแบบ

เนื้อหาส่วนนี้ประกอบไปด้วย ส่วนของการวิเคราะห์ แนวทางการวิจัยและพัฒนา การวางแผนการดำเนินการ และระยะเวลาการดำเนินงาน ภาพรวมของระบบในส่วนการออกแบบ และองค์ประกอบ ขั้นตอนกระบวนการทำงานของแต่ละส่วน รูปแบบของการจัดเก็บข้อมูลและการจัดส่งข้อมูลตลอดจนไปถึงการจัดเก็บลงในฐานข้อมูล การออกแบบการแสดงผล และระบบแจ้งเตือนสำหรับผู้ใช้งาน การออกแบบและการสร้างส่วนของฮาร์ดแวร์ การเขียนโปรแกรมควบคุมการทำงานของระบบรวมถึงวิธีการทดสอบระบบในแต่ละส่วน และการนำไปทดสอบในการตรวจวัดและการระบายอากาศ เพื่อลดอุณหภูมิและปริมาณฝุ่นภายในอาคารที่พักอาศัยต่างๆ

3.1 แนวทางการวิจัยและพัฒนา

งานวิจัยนี้ได้ ทำการวิจัยพัฒนา ระบบจัดปรับปรุงสภาพอากาศภายในบ้านแบบพลวัต โดยการประยุกต์ใช้เทคโนโลยีของ Internet of Things (IoT) ช่วยในการปรับสภาพอากาศภายในอาคาร ได้วางแผนดำเนินการตามขั้นตอน โดยเริ่มจาก ศึกษาค้นคว้าทฤษฎีของการระบายอากาศ และการตรวจวัดปริมาณฝุ่น PM 2.5 เทคนิคและข้อจำกัดต่างๆของการติดตั้งและควบคุม ศึกษาการทำงานของอุปกรณ์เซ็นเซอร์ต่าง ๆ ที่จำเป็นต่อระบบ การออกแบบอุปกรณ์ ศึกษาการใช้งาน Firebase ในแต่ละส่วน นโยบายข้อกำหนดและสิทธิ์ในการใช้งาน ภาษาที่ต้องใช้เพื่อเขียนโปรแกรมการทำงานสำหรับอุปกรณ์และ Web Service การนำมาประยุกต์ใช้ เพื่อนำมาพัฒนาอุปกรณ์และสร้าง Web Service ออกแบบหน้าตาในส่วนของผู้ใช้ Interface เพื่อให้ง่ายต่อการใช้งานของ User ทดสอบการทำงานของอุปกรณ์เซ็นเซอร์ ทดสอบการทำงานประสานระหว่างเซ็นเซอร์ และ การส่งข้อมูลไปยังส่วนการจัดเก็บข้อมูล จนถึงการทำสอบการทำงานโดยรวมของระบบ หากดูครบพร้อม แล้วนำมาปรับปรุงแก้ไข สรุปผลงานวิจัย โดยได้มีการวางแผนและระยะเวลาการดำเนินงานตามตารางที่ 3.1

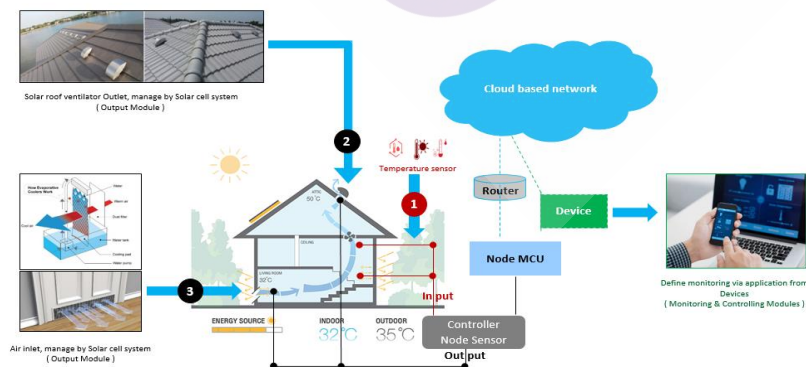
3.2 แผนการดำเนินงาน

ผู้วิจัยได้วางแผนการและระยะเวลาดำเนินงานโดยแบ่งขั้นตอนดำเนินงานเป็น 8 ขั้นตอน และใช้ระยะเวลาในการดำเนินงานทั้งสิ้นเป็นเวลา XXX เดือน โดยเริ่มต้นตั้งแต่เดือน มกราคม พ.ศ.2562 จนถึง เดือน กันยายน พ.ศ.2563 โดยรายละเอียดของแผนการดำเนินงานดูได้จากตารางที่ 3.1

ตารางที่ 3.1 แผนการดำเนินงาน

แผนการดำเนินงาน	ปี 2562-2563									
	ม.ค. 62	ก.พ. 62	มี.ค. 62	เม.ย. 62	พ.ค. 63	มิ.ถ. 63	ก.ค. 63	ส.ค. 63	ก.ย. 63	
1.ศึกษาค้นคว้าทฤษฎีและงานวิจัยในแต่ละขั้นตอนวิธี	→									
2.วิเคราะห์และออกแบบภาพรวมของระบบ			→							
3.พัฒนาอุปกรณ์และโปรแกรม			→							
4.พัฒนาอุปกรณ์ IOT สำหรับใช้งานระบบ			→							
5.ทดสอบระบบ					→					
6.ปรับปรุงแก้ไขจุดบกพร่อง					→					
7.ทดลองอุปกรณ์กับการใช้งานจริง							→			
8.สรุปผลงานวิจัยและนำเสนอ									→	

3.3 ภาพรวมของระบบ



ภาพที่ 3.1 แผนผังแสดงภาพรวมของระบบ

จากรูปที่ 3.1 จะมีส่วนของการใช้แอปพลิเคชันเพื่อใช้ควบคุมและสั่งการระบบ ผ่านเว็บเบราว์เซอร์ ส่วนอุปกรณ์ Sensor Node จะทำหน้าที่รับและส่งค่าต่างๆ หรือ Message จากตัวเซ็นเซอร์ต่างๆ ให้ส่งไปยังส่วน Gateway หรือเรียกว่าเส้นทางที่กำหนดขึ้นและส่งต่อไปยังส่วนระบบฐานข้อมูล เพื่อดำเนินการจัดเก็บข้อมูลที่ซับซ้อนของระบบ โดยภาพรวมของระบบทั้งหมดสามารถแบ่งส่วนการทำงาน ออกเป็นภาคส่วน 3 ส่วน โดยส่วนที่ 1 ตรวจวัดอุณหภูมิและปริมาณฝุ่น PM2.5 ส่วนที่ 2 ทำหน้าที่ระบายความร้อนภายในอาคาร และฝุ่นละอองเพื่อลดปริมาณฝุ่น PM2.5 และ ส่วนที่ 3 ทำหน้าที่นำความเย็นที่ได้จากอากาศภายนอกพร้อมการกรองฝุ่น PM2.5 ก่อนทำการนำเข้าอาคาร โดยให้เกิดการหมุนเวียนถ่ายเทอากาศภายในอาคาร โดยจุดประสงค์เพื่อลดอุณหภูมิและปริมาณฝุ่น PM2.5 ให้มีประสิทธิภาพที่ดีที่สุด

```

Macintoshs-MacBook-Pro-5:~ macintoshhd$ firebase --version
7.0.2
Macintoshs-MacBook-Pro-5:~ macintoshhd$ firebase login
? Allow Firebase to collect anonymous CLI usage and error reporting information? Yes

Visit this URL on any device to log in:
https://accounts.google.com/o/oauth2/auth?client_id=563584335869-farhmd47hnekij5i8h5pr03ho849e6.apps.googleusercontent.com&scope=email%20openid%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcloudplatformprojects.readonly%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Ffirebase%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcloud-platform&response_type=code&state=412472888&redirect_uri=https%3A%2F%2Flocalhost%3A9005

Waiting for authentication...

✔ Success! Logged in as phumipat.th@fastcontech.com
Macintoshs-MacBook-Pro-5:~ macintoshhd$ firebase init
-bash: $: command not found
Macintoshs-MacBook-Pro-5:~ macintoshhd$ firebase init

#####
##
##
##
##
##

You're about to initialize a Firebase project in this directory:
/Users/macintoshhd

Before we get started, keep in mind:
  * You are initializing your home directory as a Firebase project

? Which Firebase CLI features do you want to set up for this folder? Press Space to select features, then Enter to confirm your choices. (Press <space> to select, <tab> to toggle all, <esc> to invert selection)
> Database: Deploy Firebase Realtime Database Rules
   Firestore: Deploy rules and create indexes for Firestore
   Functions: Configure and deploy Cloud Functions
   Hosting: Configure and deploy Firebase Hosting sites
   Storage: Deploy Cloud Storage security rules

#####
##
##
##
##
##

You're about to initialize a Firebase project in this directory:
/Users/macintoshhd

Before we get started, keep in mind:
  * You are initializing your home directory as a Firebase project

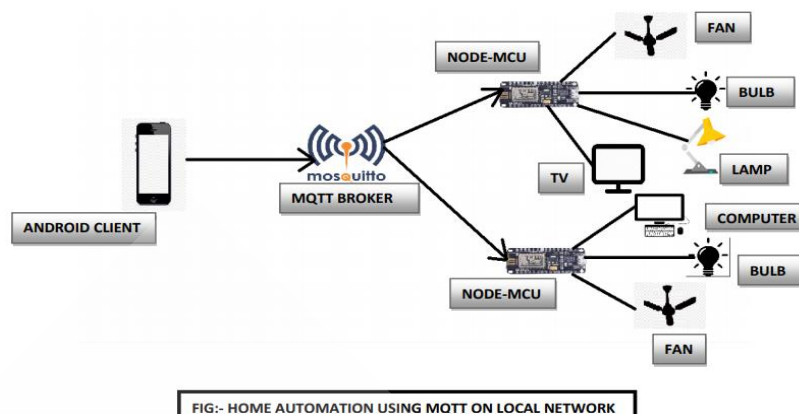
? Which Firebase CLI features do you want to set up for this folder? Press Space to select features, then Enter to confirm your choices. Database: Deploy Firebase Realtime Database Rules, Firestore: Deploy rules and create indexes for Firestore, Functions: Configure and deploy Cloud Functions, Hosting: Configure and deploy Firebase Hosting sites, Storage: Deploy Cloud Storage security rules

=== Project Setup

First, let's associate this project directory with a Firebase project.
You can create multiple project aliases by running firebase use --add,
but for now we'll just set up a default project.

? Select a default Firebase project for this directory:
[don't setup a default project]
> activeairflow-iot (ActiveAIRFLOW-IOT)
  airflow2-a4aa3 (AirFlow2)
  [create a new project]

```



ภาพที่ 3.2 ภาพรวมของระบบ Firebase on IoT

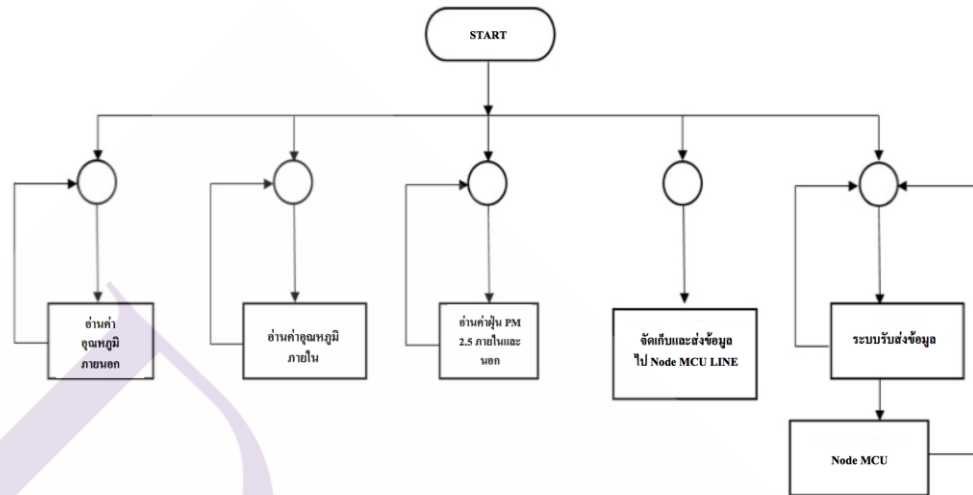
3.4 การออกแบบการทำงาน

การทำงานของระบบมีกระบวนการโดยเริ่มจากการตั้งค่าการทำงานจากส่วน User Interface โดยมีข้อมูลที่ได้จะถูกส่งจากระบบ Google Firebase จากนั้นจะถูกส่งไปยัง Gateway Firebase ต่อไป โดยตัว Gateway Firebase ทำหน้าที่เป็นทั้ง Gateway ติดต่อกันระหว่าง Sensor Node และ Firebase แล้วยังทำหน้าที่ในการประมวลผลการทำงานของระบบอีกด้วย โดยค่าที่ได้มาแล้วจะถูกเก็บเพื่อนำไปเปรียบเทียบเงื่อนไขในการทำงานในลำดับต่อไป จากนั้นส่วนอุปกรณ์ Sensor ต่าง ๆ จะทำการอ่านค่าแล้ว ส่งผลลัพธ์ไปยัง Sensor Node แล้วจึงส่งต่อไปยัง Gateway ทั้งสองตัวเพื่อทำการเปรียบเทียบเงื่อนไขในการทำงานตามคำสั่งที่ตั้งค่าไว้ โดยส่วนการทำงานของระบบได้ถูกแบ่งออกเป็นส่วนย่อย ๆ และมีรายละเอียดดังนี้

3.4.1 Sensor Node ระบบการอ่านค่าจากอุปกรณ์เซ็นเซอร์

การทำงานของระบบเริ่มจากการอ่านค่าจากอุปกรณ์เซ็นเซอร์วัดอุณหภูมิและความชื้น จากนั้นจะไปอ่านค่าอุณหภูมิของสภาพอากาศภายใน และอ่านค่ารอบการหมุนของพัดลมดูดระบาย แล้วจะจัดเก็บไว้รอสัญญาณร้องขอข้อมูลจาก Node MCU Firebase Gateway จึงจะส่งต่อไป และรอรับค่าจาก Node MCU Gateway เพื่อรวบรวมแล้วจึงจะจัดส่งไปยัง Node MCU LINE Notify ตามลำดับ ทั้งนี้เพื่อป้องกันความผิดพลาดจากการรับส่งข้อมูลซ้อนทับกันพร้อมทั้งระบบยังสามารถตั้งค่าของการส่งข้อความแจ้งเตือนไปยัง LINE API ได้สะดวกอีกด้วย ระบบจะกลับมาทำงานวนซ้ำต่อไป โดย Hardware ที่ใช้ คือ Arduino Nano ต่อกับ เซ็นเซอร์วัดอุณหภูมิ ความชื้นสัมพัทธ์ในอากาศ เซ็นเซอร์วัดอุณหภูมิในอากาศและเซ็นเซอร์วัดรอบการหมุน ของพัดลมดูดอากาศและระบายอากาศ ส่วนภาษาที่ใช้ในการเขียนโปรแกรมคือ ภาษา C เป็นหลัก โดยใช้โปรแกรม Arduino

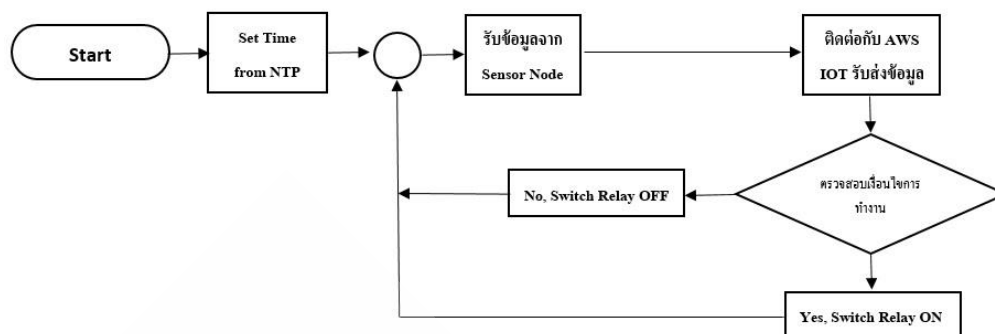
IDE ในการเขียนทั้งหมดโดยระบบมีการทำงานเป็นแบบวนลูปรการทำงาน จะทำตามขั้นตอนไปเรื่อย ๆ ซึ่งเป็นไปตามผังการทำงานของระบบดังตามรูปภาพประกอบที่ 3.3



ภาพที่ 3.3 ผังการทำงานของระบบ Sensor Node

3.4.2 Node MCU Gateway Firebase

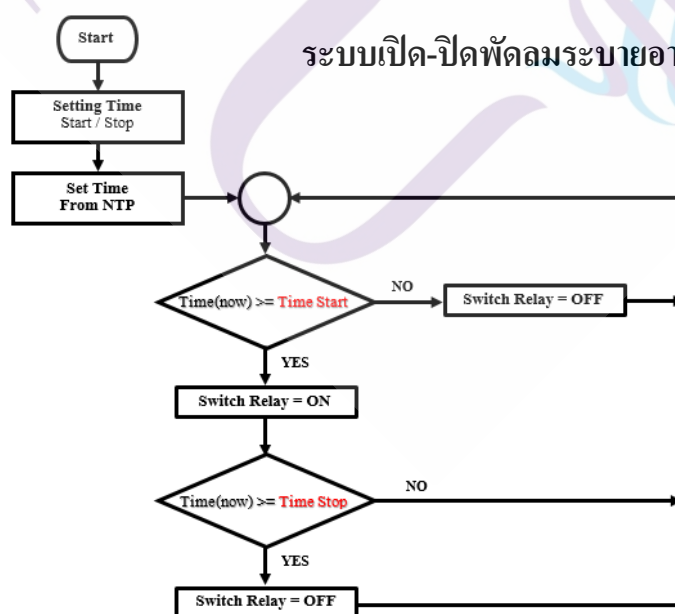
ทำหน้าที่เป็นทั้ง Gateway เพื่อติดต่อกับ Firebase Gateway ออกแบบให้เป็นตัวกลางในการรับส่งข้อมูล โดยทำหน้าที่รับข้อมูลจากตัว Sensor Node แล้วส่งต่อไปยัง AWS ต่อไป โดยระบบได้ออกแบบมาให้ ตัว Sensor Node ทำการส่งข้อมูลมาทุกระยะเวลา 1000 ms ข้อมูลจะถูกเก็บเอาไว้ที่ตัว Gateway แล้วจะถูกส่งต่อไปยัง AWS Cloud ในทุก ๆ 1 ชม. โดยอาศัยเวลาจาก NTP และจะมีที่ใช้ในการปรับปรุงระบบ ถ้าหากมีการเปลี่ยนแปลงเกิดขึ้นจะถูกส่งไปยัง AWS Cloud โดยทันที ส่วนฝั่งรับค่าออกแบบให้รอรับค่าอยู่ตลอดเวลา เมื่อมีค่าที่ถูกส่งเข้ามาใหม่ แล้วจะแจ้งสถานการณ์ปรับแต่งค่ากลับไปยัง AWS Cloud อีกครั้งหนึ่ง เพื่อเป็นการยืนยันการรับค่าสำเร็จ และยังทำหน้าที่เป็นหน่วยประมวลผลการทำงาน, ควบคุมการทำงาน และแสดงผลการทำงานทั้งหมดตามภาพที่ 3.4 โดยแบ่งเป็นระบบย่อย ๆ ภายในได้ดังนี้



ภาพที่ 3.4 ผังการทำงานของระบบ Gateway Firebase

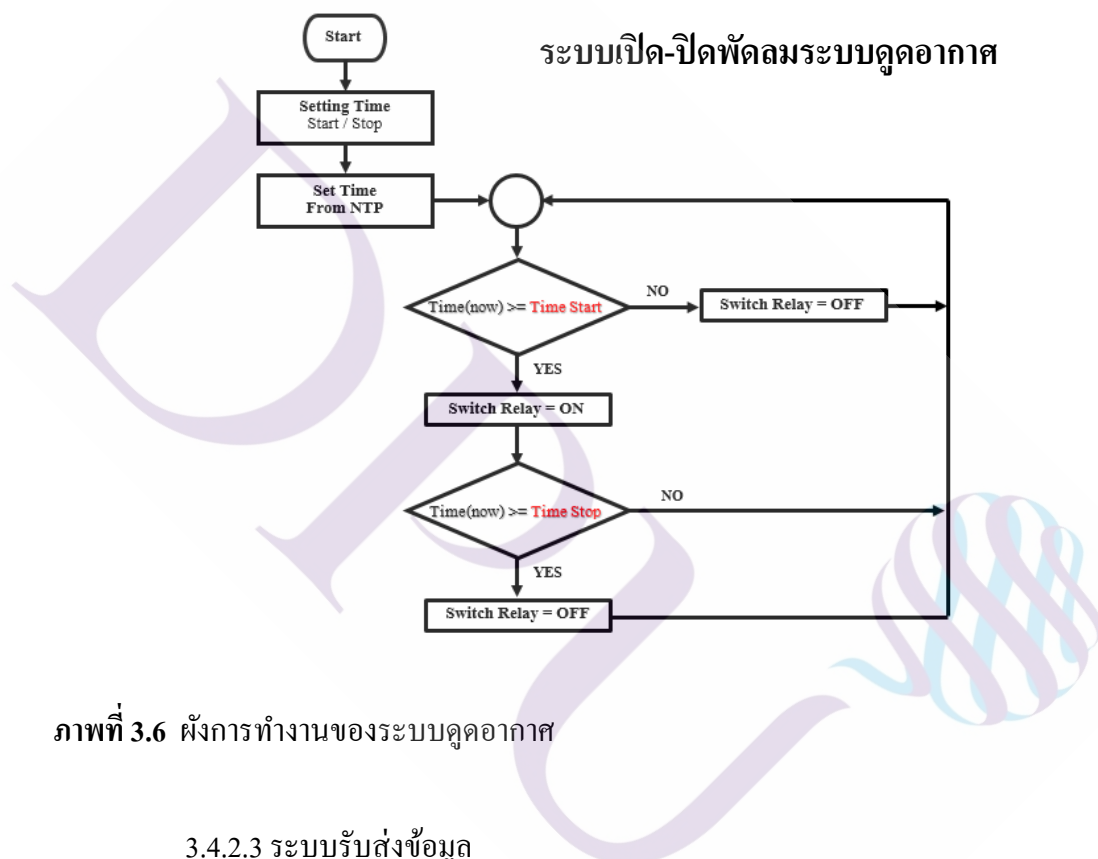
3.4.2.1 ระบบควบคุมการเปิดปิดพัดลมระบายและดูดอากาศ

ออกแบบ โดยการตั้งค่า เวลาเปิด และ เวลาปิด ตามเวลาจริงจาก NTP โดยระบบ จะมีการตรวจสอบเวลาปัจจุบันกับค่าที่ตั้งไว้ ถ้าเวลาปัจจุบันมีค่ามากกว่า หรือเท่ากับเวลาเปิดที่ตั้งไว้ ระบบจะทำการระบบจ่ายสัญญาณไฟฟ้า HIGH ไปยัง สวิตซ์อิเล็กทรอนิกส์เพื่อสั่งการทำงานให้ เปิดไฟส่องสว่างให้กับพีช ถ้าเวลาปัจจุบัน มีค่ามากกว่าหรือเท่ากับ เวลาปิดที่ตั้งไว้ ระบบจะทำการ ระบบจ่ายสัญญาณไฟฟ้า LOW ไปยัง สวิตซ์อิเล็กทรอนิกส์เพื่อสั่งการทำงานปิดไฟส่องสว่างให้กับ พีชตามภาพประกอบที่ 3.5



ภาพที่ 3.5 ผังการทำงานของระบบเปิดปิดพัดลมระบายอากาศ

3.4.2.2 ระบบควบคุมการจ่ายกระแสไฟเพื่อเพิ่มหรือลดรอบการหมุนของพัดลมระบายอากาศ โดยจะทำงานตามช่วงเวลาที่กำหนดเพื่อระบายอากาศออกจากอาคารและระบายออกสู่ภายนอกอาคาร โดยการเปิดปิดนั้น จะใช้การกำหนดตามช่วงเวลาที่เหมาะสมคือ ทำงานเป็นระยะเวลา 40 นาทีและหยุดทำงานเป็นเวลา 20 นาที เหตุเพื่อช่วยลดการใช้พลังงาน โดยระบบควบคุมการเปิดปิดกระแสไฟสามารถปรับค่าได้ในภายหลังตามความต้องการตามภาพที่ 3.6



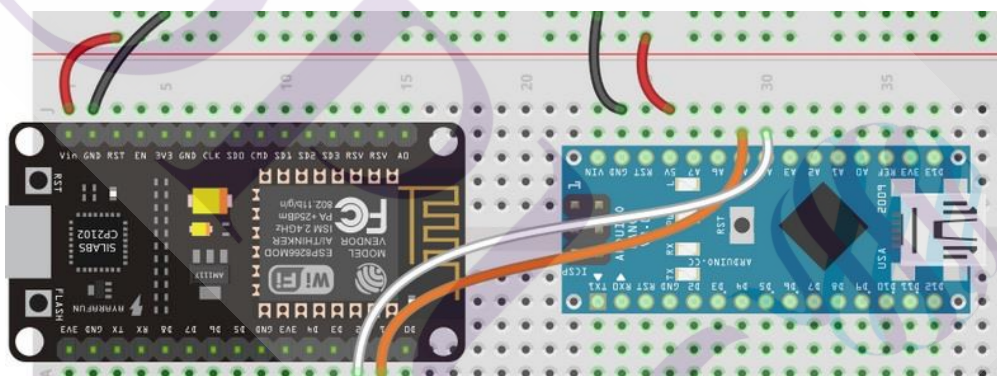
ภาพที่ 3.6 ฟังก์การทำงานของระบบดูดอากาศ

3.4.2.3 ระบบรับส่งข้อมูล

การรับส่งข้อมูลระหว่าง Node MCU Gateway Firebase กับ Sensor Node นั้น จะส่งข้อมูลผ่าน Serial ภายในตัวอุปกรณ์ตามภาพที่ 3.6 โดยเริ่มจากการที่ Node MCU Gateway Firebase จะส่ง Key เพื่อที่จะแจ้งแก่ Sensor Node ว่าต้องการจะรับข้อมูลแล้วจากนั้น Sensor Node ถึงจะเริ่มส่งข้อมูลที่ต้องการมา แล้วถึงจะเก็บข้อมูลไว้ภายใน Node MCU Gateway Firebase การรับส่งข้อมูลระหว่าง Google Firebase Cloud กับ Node MCU Gateway จะส่งผ่าน Firebase IoT โดยใช้ โพรโตคอล MQTT ในการติดต่อสื่อสารระหว่างกันผ่าน Internet โดยอาศัย wifi Router อีกที โดย มีการกำหนดตัวแปรสำคัญที่ใช้ในการรับส่งข้อมูลระหว่าง Sensor Node (Arduino Nano) และ Node MCU Firebase โดยมีรายละเอียดตามตารางที่ 3.2

ตารางที่ 3.2 ตัวแปรที่ใช้ในการรับส่งข้อมูลระหว่าง Sensor Node และ Node MCU Firebase

Sensor Node			Node MCU		
ชื่อตัวแปร	ชนิด	ความหมาย	ชื่อตัวแปร	ชนิด	ความหมาย
KEY1	String	อ้างอิงรับส่งข้อมูล	KEY1	String	อ้างอิงรับส่งข้อมูล
KEY2	String	อ้างอิงรับส่งข้อมูล	KEY2	String	อ้างอิงรับส่งข้อมูล
H Hour	Int	จำนวนชั่วโมง	H Hour	Int	จำนวนชั่วโมง
Humi	Float	ไม่ได้ใช้งาน	Humi	Float	ไม่ได้ใช้งาน
Temp	Float	อุณหภูมิอากาศ	Temp	Float	อุณหภูมิอากาศ
H Temp	Float	อุณหภูมิความร้อน	H Temp	Float	อุณหภูมิความร้อน
Temp C	Int	ระดับอุณหภูมิอากาศที่ต้องการแจ้งเตือน	Temp C	Int	ระดับอุณหภูมิอากาศที่ต้องการแจ้งเตือน



ภาพที่ 3.7 การเชื่อมต่อ Arduino Nano กับ Node MCU

3.4.2.4 ระบบการบันทึกลงหน่วยความจำ EEPROM

ข้อมูลการตั้งค่าและข้อมูลที่มีความสำคัญอื่น ๆ หลังจาก that Node MCU Gateway AWS ทำงานเสร็จแล้ว จะถูกเขียนเก็บบันทึกไว้ใน EEPROM เพื่อป้องกันข้อมูลเสียหายหากเกิดกรณีไฟฟ้าดับ, ระบบขัดข้องหรือในกรณีรีเซ็ตระบบใด ๆ รวมไปถึงการจัดเก็บวันที่และเวลาของการระบายอากาศ เนื่องจาก Node MCU มีหน่วยความจำหลักที่สามารถใช้ได้ 4096 bytes และ จำกัดจำนวนครั้งในการบันทึกลงใน EEPROM ด้วย ผู้วิจัยจึงได้ทำการออกแบบการทำงาน

การบันทึกลงหน่วยความจำ EEPROM เพียงบางส่วนเท่านั้น โดยมีการกำหนดตัวแปรดังตารางที่ 3.3 โดยมีรายละเอียด ของการใช้งานค่าตัวแปรดังต่อไปนี้

FG : เก็บ timestamp เพื่อเก็บค่าวันที่และเวลาของการดูด และระบายอากาศในครั้งแรก สามารถกำหนดได้โดยผู้ใช้งาน

E1 : เก็บค่าเพื่อใช้ในการอ้างอิงกับ Light On เพื่อใช้สำหรับเปรียบเทียบเงื่อนไขในการเปิดพัดลมดูด-ระบายอากาศตามช่วงเวลา โดยการทำงานจะเริ่มจากอ่านค่าใน EEPROM/E1 แล้วบันทึกค่าที่ได้ลงไปที Light On จากนั้นเมื่อค่า Light On มีการเปลี่ยนแปลง จะทำการเขียนค่ากลับไปยัง EEPROM/E1 เพื่อปรับปรุงค่าโดยการทำงานนี้จะทำงานตามเงื่อนไขทางเวลาอีกด้วย เพื่อไม่ให้มีการเขียน EEPROM มากเกินไป

E2 : เก็บค่าเพื่อใช้ในการอ้างอิงกับ Light OFF เพื่อใช้สำหรับเปรียบเทียบเงื่อนไขในการเปิดพัดลมดูด-ระบายอากาศตามช่วงเวลา โดยการทำงานจะเริ่มจาก อ่านค่าใน EEPROM/E1 แล้วบันทึกค่าที่ได้ลงไปที Light OFF จากนั้น เมื่อค่า Light OFF มีการเปลี่ยนแปลง จะทำการเขียนค่ากลับไปยัง EEPROM/E1 เพื่อปรับปรุงค่าโดยการทำงานนี้ จะทำงานตามเงื่อนไขทางเวลาอีกด้วย เพื่อไม่ให้มีการเขียน EEPROM มากเกินไป

E3 : เก็บค่าเพื่อใช้ในการอ้างอิงกับ Temp C เพื่อใช้สำหรับกำหนดค่าในการแจ้งเตือนอุณหภูมิ ที่ผู้ใช้งานได้ทำการปรับแต่งค่าเอาไว้ มีการทำงานในการอ่านเขียน EEPROM เช่นเดียวกับ การอ่านเขียนค่าใน E1,E2

E4 : เก็บค่าเพื่อใช้ในการอ้างอิงกับ W temp C เพื่อใช้สำหรับกำหนดค่าในการแจ้งเตือนอุณหภูมิของอากาศภายในอาคาร ที่ผู้ใช้งานได้ทำการปรับแต่งค่าเอาไว้ มีการทำงานในการอ่านเขียน EEPROM เช่นเดียวกับ การอ่านเขียนค่าใน E1,E2

E5 : เก็บค่าเพื่อใช้ในการอ้างอิงกับ MODE เพื่อใช้สำหรับกำหนดค่าในการทำงานพัดลมดูด-ระบายอากาศสามารถเลือกได้ ทั้งแบบกำหนดเองตามผู้ใช้งานและแบบอัตโนมัติพัดลมดูด-ระบายอากาศ มีการทำงานในการอ่านเขียน EEPROM เช่นเดียวกับ การอ่านเขียนค่าใน E1,E2

ตารางที่ 3.3 ตัวแปรที่จัดเก็บใน EEPROM

ชื่อตัวแปร	ชนิดตัวแปร	อ้างอิง
FG	Long	timestamp
E1	Int	Light On
E2	Int	Light OFF
E3	Int	Temp C
E4	Int	H temp C
E5	Int	MODE

3.4.3 Node MCU LINE ระบบแจ้งเตือน

ระบบแจ้งเตือนได้ออกแบบการทำงานโดยการรอรับค่าจาก Sensor Node แล้วจึงนำค่าที่ได้มาเปรียบเทียบกับเงื่อนไข ถ้าตรงไปตามเงื่อนไขจะทำการส่งไปยัง LINE API Notify เพื่อส่งข้อความแจ้งเตือนมายังโทรศัพท์เคลื่อนที่ โดยการแจ้งเตือนนั้นได้กำหนดเอาไว้ คือ การแจ้งเตือนระดับอุณหภูมิร้อนที่สะสมขึ้นภายในอาคาร การแจ้งเตือนอุณหภูมิของสภาพแวดล้อม การแจ้งเตือนอุณหภูมิภายในอาคาร และการรายงานประจำวันเพิ่มเติมอื่น ๆ โดยอาศัยเวลาจาก NTP มาเป็นตัวช่วยในการทำงานตามเงื่อนไข มีการกำหนดตัวแปรที่สำคัญที่ใช้ โดยการรับค่ามาจาก Sensor Node โดยมีรายละเอียดตามตารางที่ 3.4

ตารางที่ 3.4 แสดงค่าที่ Node MCU LINE รับมาจาก Sensor Node

ชื่อ	ชนิดตัวแปร	ความหมาย	หมายเหตุ
LINEKEY	String	เป็น Key ที่ใช้อ้างอิงในการรับข้อมูล	ถ้า Key ไม่ถูกต้อง จะไม่รับข้อมูล
HUMI	Float	ค่าความชื้นสัมพัทธ์ในอากาศ	จัดเก็บ
TEMP	Float	อุณหภูมิในอากาศ	จัดเก็บเพื่อนำไปวิเคราะห์ต่อไป
Hot TEMP	Float	อุณหภูมิในอาคาร	จัดเก็บเพื่อนำไปวิเคราะห์ต่อไป
Hot Level	Int	ระดับอุณหภูมิที่มีค่าเป็น 0,1	ถ้าค่าเป็น 0 จะทำการแจ้งเตือน
Temp C	Int	ระดับอุณหภูมิที่ต้องการแจ้งเตือน	ถ้าค่าที่ได้น้อยกว่า TEMP จะทำการแจ้งเตือน
Hot Temp C	Int	ระดับอุณหภูมิที่ต้องการแจ้งเตือน	ถ้าค่าที่ได้น้อยกว่า Hot TEMP จะทำการแจ้งเตือน

จากตารางที่ 3.4 ได้แสดงชื่อของตัวแปรและชนิดของตัวแปรโดยมีรายละเอียดดังต่อไปนี้

LINEKEY : ใช้สำหรับเป็น Key ที่ใช้อ้างอิงในการรับข้อมูล ถ้าค่า Key ที่ได้ไม่ตรงกับค่ากำหนดไว้ระบบจะไม่ทำการใดๆ แต่ถ้ามีค่าที่ตรงกันจะทำการ update ข้อมูลต่อไป

HUMI : ใช้สำหรับเก็บค่าความชื้นสัมพัทธ์ในอากาศ

TEMP : ใช้สำหรับเก็บค่าอุณหภูมิในอากาศ

Hot TEMP : ใช้สำหรับเก็บค่าอุณหภูมิภายในอาคาร

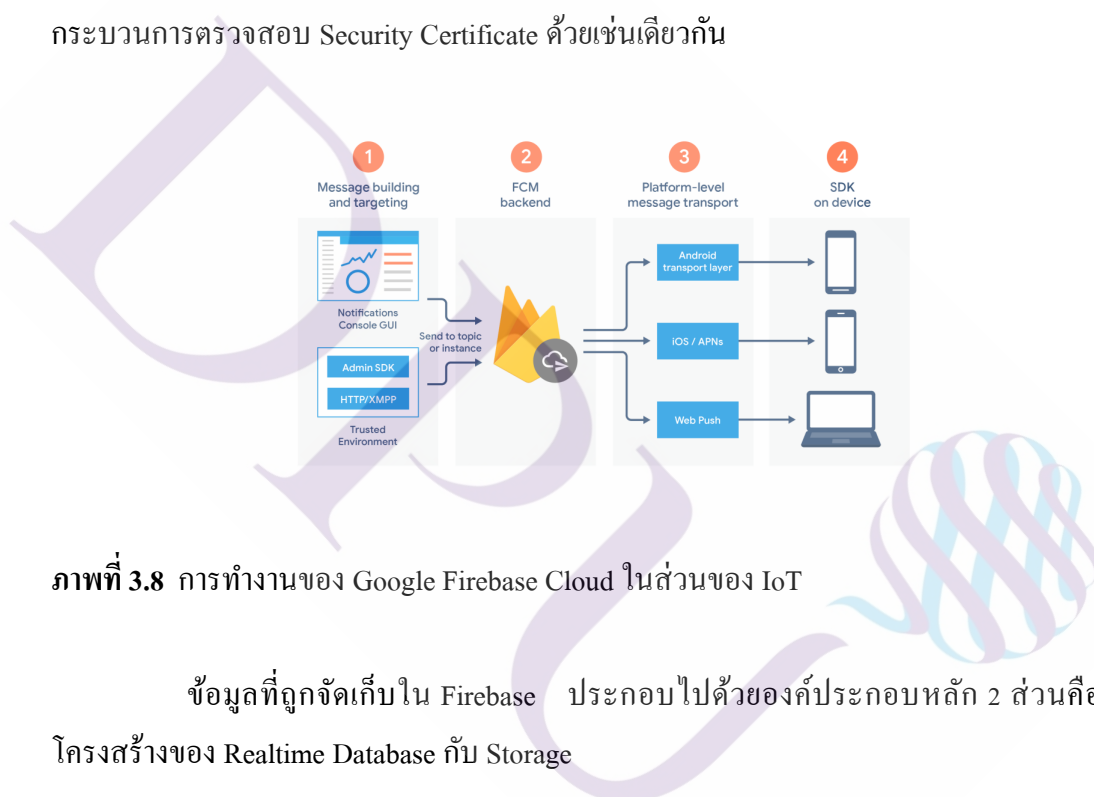
Hot Level : ใช้สำหรับเก็บค่าสถานะของระดับอุณหภูมิภายในอาคาร ถ้าระดับน้ำต่ำกว่าที่กำหนดจะแสดงค่า 0

Temp C : ใช้สำหรับเก็บค่าของการแจ้งเตือนอุณหภูมิในอากาศ โดยผู้ใช้สามารถกำหนดค่าได้เอง ถ้าค่าอุณหภูมิในอากาศจริง มากกว่าค่าของ Temp C ระบบจะทำการแจ้งเตือนผ่านระบบ LINE API Notify ทันที

Hot Temp C : ใช้สำหรับเก็บค่าของการแจ้งเตือน อุณหภูมิในน้ำ โดยผู้ใช้สามารถกำหนดค่าได้เอง ถ้าค่าอุณหภูมิภายในอาคารมีมากกว่าค่าของ Hot Temp C ระบบจะทำการแจ้งเตือนผ่านระบบ LINE API Notify ทันที

3.4.4 Firebase Cloud ระบบจัดเก็บข้อมูล

Fire Cloud ทำหน้าที่รับส่งข้อมูลต่าง ๆ โดยใช้ MQTT แล้วจะเก็บไว้ใน FCM Backend โดยมีขั้นตอนการรับข้อมูลคือ เริ่มด้วยกระบวนการตรวจสอบ Security Certificate จาก Firebase IoT ก่อน ถ้าตรวจสอบสำเร็จจะถูกส่งไปยัง IOT Thing and Shadow ที่ทำหน้าที่เป็น MQTT-Broker แล้ว Firebase จะทำการ Subscribe เพื่อจัดเก็บข้อมูล หลังจากนั้นจะทำการเรียกข้อมูลจาก Firebase โดยตรง (เรียกผ่าน Web browser) ส่วนกระบวนการส่งข้อมูล จะเรียกผ่านหน้า Web browser ภายในระบบ Platform-level message transport จะถูกส่งไปยัง IOT Thing and Shadow โดยตรง เพื่อรอการ Subscribe จาก Device Gateway โดยผ่าน IOT Thing and Shadow ตามภาพที่ 3.8 โดยมีกระบวนการตรวจสอบ Security Certificate ด้วยเช่นเดียวกัน



ภาพที่ 3.8 การทำงานของ Google Firebase Cloud ในส่วนของ IoT

ข้อมูลที่ถูกจัดเก็บใน Firebase ประกอบไปด้วยองค์ประกอบหลัก 2 ส่วนคือส่วนโครงสร้างของ Realtime Database กับ Storage



ภาพที่ 3.9 ส่วนโครงสร้าง Realtime Database ใน Storage

1. ส่วนโครงสร้าง Realtime Database จะมีข้อมูลหลักที่ใช้คือ Partition Key (NodeID) กับ Sort Key (timestamp) Partition Key (NodeID) ทำหน้าที่สำหรับอ้างอิงอุปกรณ์ IOT โดยต้องมีเปิดการใช้งานและมีตรวจสอบกับ Firebase IoT Certificate เสียก่อน ผู้วิจัยได้ออกแบบให้ชื่อ NodeMCU_1 เพื่อเป็นตัวแทนสำหรับผู้ใช้แต่ละรายไป โดยที่ผู้ใช้สามารถมีอุปกรณ์ได้หลายชิ้น โดยอาศัย IoT Thing เดียวกัน Sort Key (timestamp) ใช้สำหรับอ้างอิงถึงเวลาที่รับส่งข้อมูล

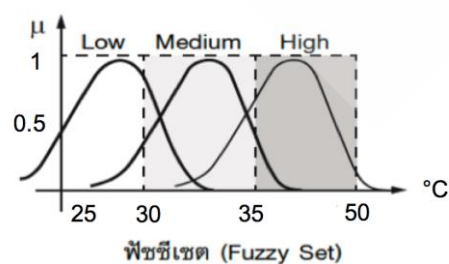
2. ส่วน Storage เป็นส่วนของข้อมูลข้างในที่จะถูกส่งออกไปจาก Node MCU Gateway การส่งข้อมูลจะมีอยู่ 2 ประเภทด้วยกันคือ ข้อมูลรายงานสถานะสภาพของอุณหภูมิ

3.4.5 Fuzzy Logic การประมวลผล ความคลุมเครือให้สอดคล้องกับการปรับอุณหภูมิและปริมาณฝุ่น โดยเพื่อแสดงให้เห็นความแตกต่างของทั้งเซตวินัยและฟัซซีเซตที่มีหลายเซต ดังนั้นในที่นี้จะขอแบ่ง อุณหภูมิโดยทั่วไปเป็นเซตต่างๆ ดังแสดงด้วยรูป 3.10 กล่าวคือ อุณหภูมิตั้งแต่ 30 – 50 °C กำหนดให้เป็นดังนี้

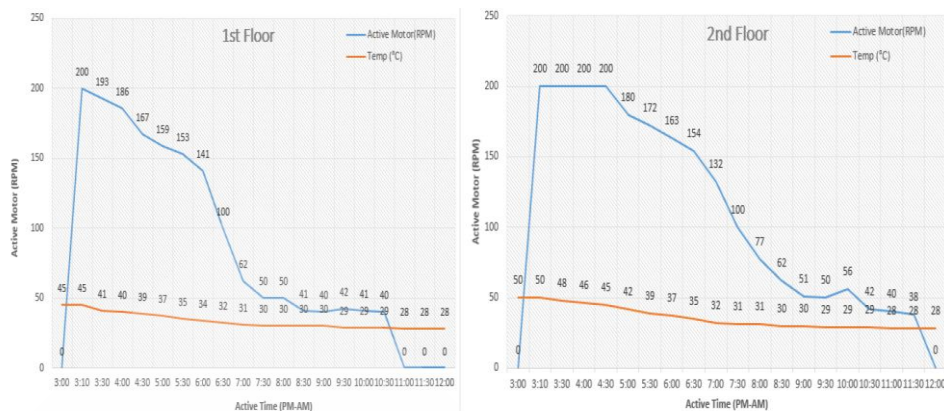
อุณหภูมิต่ำ (Low) ของเซตวินัยโดยค่าความเป็นสมาชิกของเซตในช่วงดังกล่าวเป็น 1 นอกนั้นเป็น 0 แต่ในส่วนฟัซซีเซตกำหนดช่วงอุณหภูมิต่ำ เป็น 30 – 32 °C แล้วค่าความเป็นสมาชิกของเซตอยู่ระหว่าง 0 กับ 1 เช่นที่อุณหภูมิ 30 °C ค่าความเป็นสมาชิกเท่ากับ 0.5 และอุณหภูมิ 32 °C ค่าความเป็นสมาชิกเท่ากับ 1 เป็นต้น

อุณหภูมิปานกลาง (Medium) ของเซตวินัยคืออุณหภูมิตั้งแต่ 33 – 35 °C โดยค่าความเป็นสมาชิกของเซตในช่วงดังกล่าวเป็น 1 นอกนั้นเป็น 0 ส่วนฟัซซีเซตจะกำหนดช่วงอุณหภูมิปานกลางไว้ที่ 33 – 35 °C และหากเราพิจารณาจากรูป ก็จะพบว่าในกรณีของฟัซซีเซต เซตของอุณหภูมิปานกลางกับอุณหภูมิต่ำมีการซ้อนทับกันอยู่ (Overlap) ซึ่งเป็นการสื่อแสดงถึงความคลุมเครือให้ความสำคัญต่อตัวแปรภาษามากกว่าตัวแปรปริมาณ

สำหรับช่วงของ อุณหภูมิสูง (High) ของทั้งเซตวินัยและฟัซซีเซตก็จะคล้ายกับการอธิบายที่ผ่านมาดังแสดงในรูปที่ 3.10



ภาพที่ 3.10 การกำหนดค่าอุณหภูมิโดยใช้ทฤษฎีของฟัซซีเซต



ภาพที่ 3.11 การกำหนดรอบความเร็วของพัดลมและอุณหภูมิโดยใช้ทฤษฎีของฟิชชีเซต

3.5 สรุปเนื้อหา

เนื้อหาในบทนี้ได้กล่าวถึง องค์ประกอบและขั้นตอนของการศึกษาวิจัยและการออกแบบระบบ โดยกระบวนการที่ต้องนำมาออกแบบระบบในทุกส่วน ทั้งระบบโดยรวมและระบบโดยย่อยในงานวิจัยทั้งหมด การประมวลผลในการทำงาน การคิดและการตัดสินใจ และกำหนดอุปกรณ์ที่จะนำมาใช้ในขั้นตอนการรับและส่งข้อมูลจาก Node ต่างๆ จนมาจัดเก็บใน Database จนถึงการนำข้อมูลมาใช้งาน การออกแบบ Hardware , Diagram และการออกแบบส่วน ของ User Interface ของการแสดงผลและควบคุมการทำงาน ระบบแจ้งเตือน ตลอดจนวิธีที่จะใช้ทดสอบประสิทธิภาพของระบบ

บทที่ 4

ผลการทดสอบระบบ

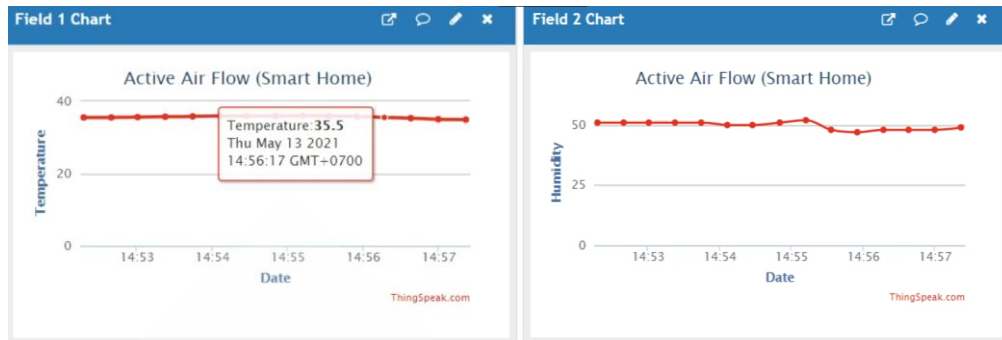
สำหรับเนื้อหาภายในบทนี้จะประกอบไปด้วย แนวทางของวิทยานิพนธ์ที่มีองค์ประกอบของการทดลองระบบ จากการออกแบบตัวควบคุมพีซีซีลอจิกและตัวควบคุมอินทิกรัลแบบปรับตัวได้ ตัวควบคุมทั้งสองได้ถูก นำมาทดสอบร่วมกับชุดควบคุมอุณหภูมิภายในอาคาร สำหรับช่วงของการควบคุมจะอยู่ระหว่าง 0 ถึง 100 เปอร์เซ็นต์ (ประมาณ 25-35 องศาเซลเซียส) จุดทำงานกำหนดไว้ที่ 50 เปอร์เซ็นต์ (ประมาณ 30 องศาเซลเซียส) และความเร็วใบพัดปกติขณะทดสอบจะอยู่ที่ประมาณ 4000 รอบต่อนาที โดยการทดสอบประสิทธิภาพของตัวควบคุมผ่านแพลตฟอร์มผู้ให้บริการเก็บข้อมูลบนระบบคลาวด์ ด้วยบริการของ ThingSpeak ซึ่งจะถูกแบ่งออกเป็น 3 ส่วน ดังนี้

ส่วนแรกเป็นสถานะการตรวจวัดด้วยเซ็นเซอร์วัดอุณหภูมิ ที่ย่านอุณหภูมิสูงที่ 30-35 องศาเซลเซียส ปานกลางที่ 27 องศาเซลเซียส ปกติที่ 25 องศาเซลเซียส และต่ำกว่า 25 องศาเซลเซียส ระบบหยุดการทำงาน ส่วนที่สองเป็นสถานะที่ควบคุมการหมุนใบพัดลม ดูดและระบายอากาศ ด้วยรอบ สูง ปานกลาง ปกติ ส่วนที่สามเป็นผลจากการจำลองทำงานของระบบควบคุมแบบพีซีซีลอจิก

4.1 สถานะการตรวจวัดด้วยเซ็นเซอร์วัดอุณหภูมิ

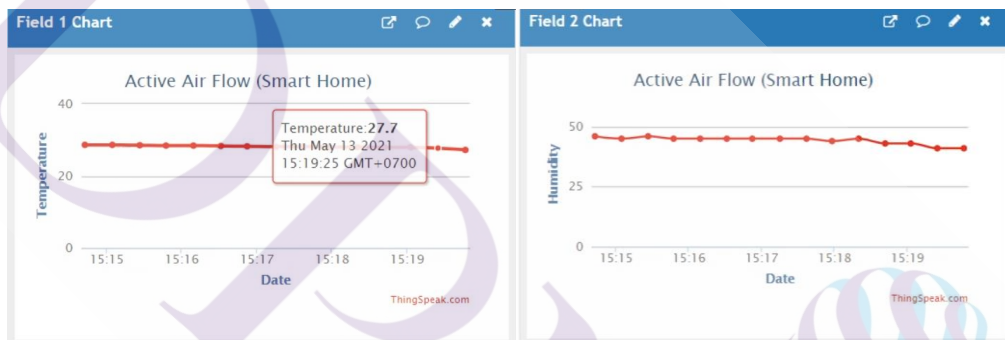
ในการทดสอบระบบจะแบ่งการใช้งานออกเป็น 4 ส่วนด้วยกัน คือ ส่วนแรกอุณหภูมิสูง 30-35 องศาเซลเซียส ส่วนที่สองอุณหภูมิปานกลาง 27 องศาเซลเซียส ส่วนที่สามอุณหภูมิปกติ 25 องศาเซลเซียส ส่วนสุดท้ายอุณหภูมิต่ำกว่า 25 องศาเซลเซียส ระบบหยุดการทำงานทันที ด้วยผลกราฟตามเวลาจริงและบนสมาร์ตโฟน

4.1.1 ส่วนแรกอุณหภูมิสูง 30-35 องศาเซลเซียส



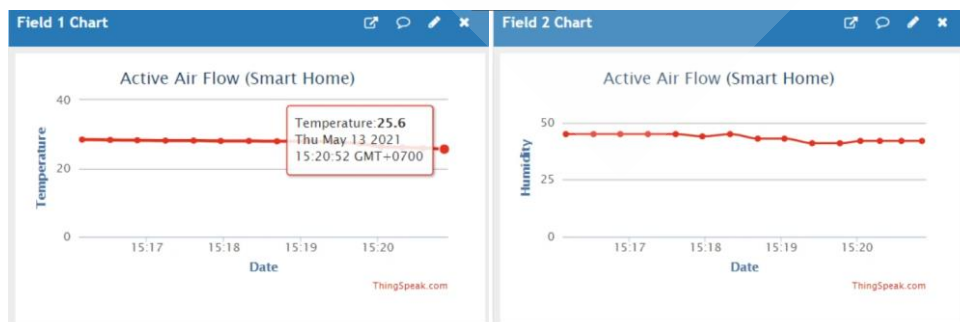
ภาพที่ 4.1 รายงานผลกราฟตามเวลาจริง แสดงอุณหภูมิสูง และความชื้นผ่าน HTTP : ThingSpeak

4.1.2 ส่วนที่สองอุณหภูมิปานกลางที่ 27 องศาเซลเซียส



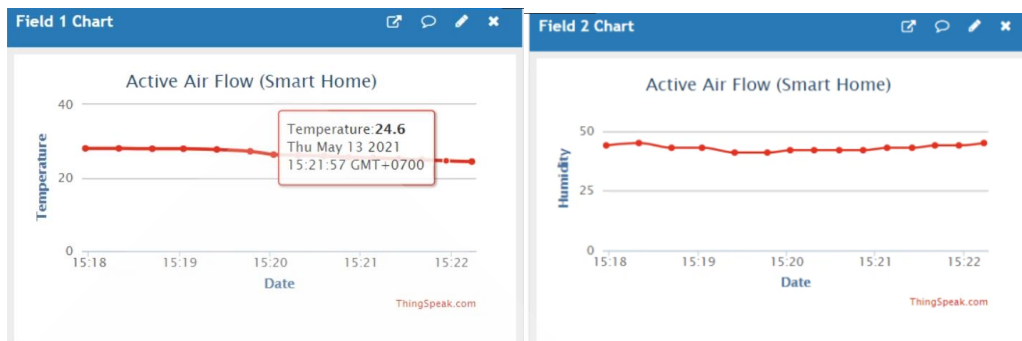
ภาพที่ 4.2 รายงานผลกราฟตามเวลาจริง แสดงอุณหภูมิปานกลาง และความชื้น ผ่าน HTTP : ThingSpeak

4.1.3 ส่วนที่สามอุณหภูมิปกติที่ 25 องศาเซลเซียส



ภาพที่ 4.3 รายงานผลกราฟตามเวลาจริง แสดงอุณหภูมิปกติ และความชื้นผ่าน HTTP : ThingSpeak

4.1.4 ส่วนที่สี่อุณหภูมิต่ำกว่า 25 องศาเซลเซียส ซึ่งจะกำหนดให้ระบบหยุดการทำงาน



ภาพที่ 4.4 รายงานผลกราฟตามเวลาจริง แสดงอุณหภูมิต่ำกว่า 25 องศาเซลเซียส ผ่าน HTTP :
ThingSpeak





ภาพที่ 4.5 รายงานผลแสดงอุณหภูมิและความชื้นภายในอาคาร เพื่อเปรียบเทียบและยืนยันผ่านสมาร์ตโฟนโดยใช้ MQTT : Blynk Application

Range of Temperature (T) จะทำหน้าที่แบ่งออกเป็น 4 target group โดยกำหนดช่วงการทำงานพัดลมโดยการกำหนดของอุณหภูมิ ส่งข้อมูลผ่านแพลตฟอร์ม Thing Speak สำหรับมอเตอร์ พร้อม Blynk Application เพื่อควบคุมการใช้งาน ดังต่อไปนี้

- T1 :30, Blynk write // ThingSpeak: float cuff_t_top >= 30; (High Speed)
- T2 :27, Blynk write // ThingSpeak: float cuff_t_middle >= 27; (Middle Speed)
- T3 :25, Blynk write // ThingSpeak: float cuff_t_bottom = 25; (Normal Speed)
- T4 :<25, Blynk write // ThingSpeak: float cuff t stop = <25; (Stop)

โดยกำหนดให้ T คือ Temperature อุณหภูมิ เป็นตัวกำหนดทุกฟังก์ชันเพื่อควบคุมรอบการหมุนของพัดลม จากตัวอย่างกราฟในรูปที่ 4.1.1 ถึง 4.1.4 สามารถอธิบายได้เป็นลำดับขั้นตอนดังนี้

1. 30°C ถึง 35°C เนื่องด้วยอุณหภูมิต่ำกว่าหรือมากกว่า 30 องศาเซลเซียส ขึ้นไป ระบบจะประมวลผลสั่งการให้พัดลมฝั่ง Fresh Air และ Exhaust Air ทำงานในรอบสูง

2. 27°C ถึง 29°C เนื่องด้วยอุณหภูมิเท่ากับหรือมากกว่า 27 องศาเซลเซียส ขึ้นไปจนถึง 29 องศาเซลเซียส ระบบจะประมวลผลสั่งการให้พัดลมฝั่ง Fresh Air และ Exhaust Air ทำงานในรอบปานกลาง

3. 25°C ระบบกำหนดให้มีการหมุนในรอบปกติคือ 4,000 รอบต่อนาที

4. ต่ำกว่า 25°C องศาเซลเซียส ระบบจะประมวลผลสั่งการให้พัดลมฝั่ง Fresh Air และ Exhaust Air หยุดการทำงานทันที เพื่อส่งต่อหน้าที่การทำงานให้กับเครื่องปรับอากาศ

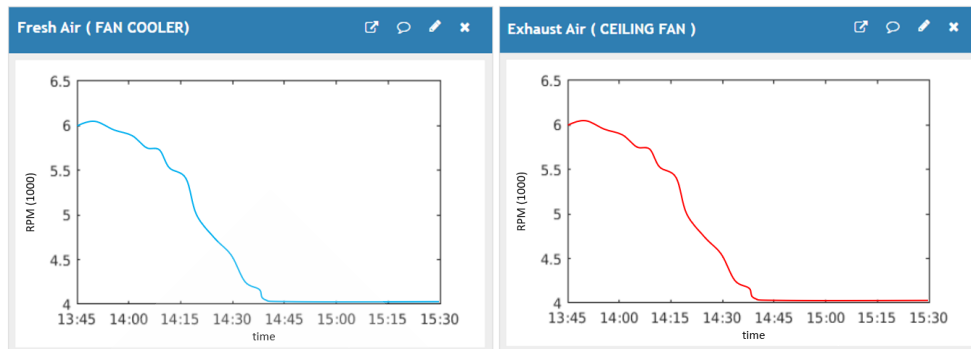
ตารางที่ 4.1 แสดงข้อมูลสรุปการทดสอบหัวข้อที่ 4.1

Sender (Sensing)	Device (CPU)	Web Browser HTTP	Application MQTT	Temperature (C°)	Humidity (%)	Maximum Temperature (C°)	Average Temperature (C°)	Minimum Temperature (C°)
DHT11	ESP8266	Thing Speak	Blynk	≥ 30	48	35.5	32.8	30.2
DHT11	ESP8266	Thing Speak	Blynk	≥ 27	42	29.5	28.6	27.7
DHT11	ESP8266	Thing Speak	Blynk	≥ 25	40	25.6	25.3	25.1
DHT11	ESP8266	Thing Speak	Blynk	< 25	39	24.6	24.1	23.7

4.2 สถานะที่ควบคุมการหมุนใบพัดลมดูดและระบายอากาศด้วยรอบ สูง ปานกลาง ปกติ หยุดการทำงาน

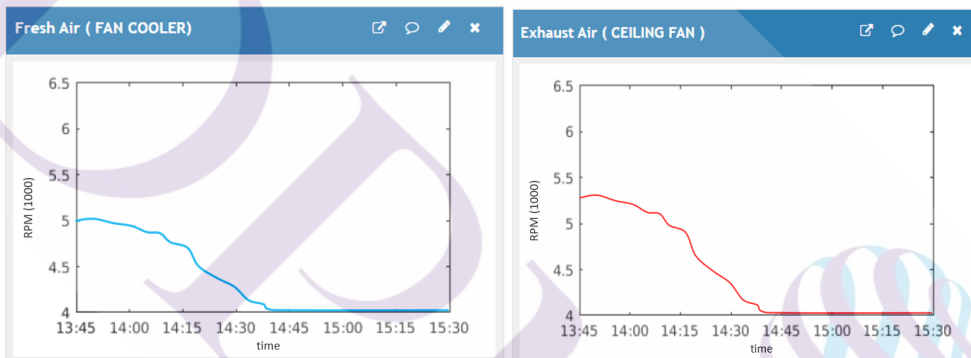
ในการทดสอบระบบจะแบ่งการใช้งานออกเป็น 4 ส่วนด้วยกัน คือ ส่วนแรกรอบการหมุน 6,000 รอบต่อนาที (อุณหภูมิสูง 30-35 องศาเซลเซียส) ส่วนที่สองรอบการหมุน 5,000 รอบต่อนาที (อุณหภูมิปานกลาง 27 องศาเซลเซียส) ส่วนที่สามรอบการหมุน 4,000 รอบต่อนาที (อุณหภูมิปกติ 25 องศาเซลเซียส) และส่วนสุดท้ายรอบการหมุนเป็น 0 (อุณหภูมิต่ำกว่า 25 องศาเซลเซียส) โดยจะกำหนดให้ระบบหยุดการทำงานทันที

4.2.1 ส่วนแรกรอบการหมุน 6,000 รอบต่อนาที (อุณหภูมิสูง 30-35 องศาเซลเซียส)



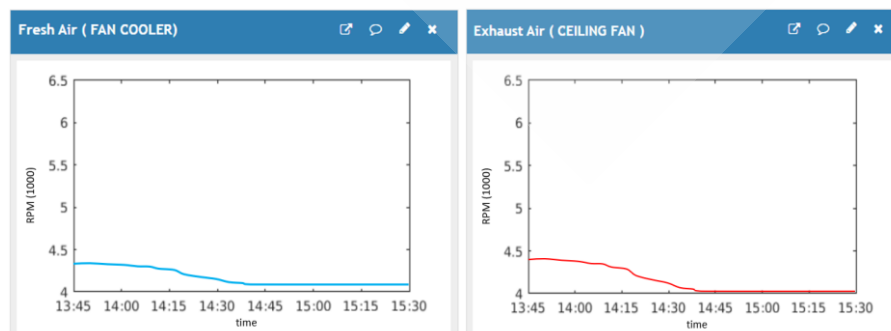
ภาพที่ 4.6 กราฟแสดงรอบการหมุนพัดลมดูดและระบายอากาศรอบสูง 6,000 รอบต่อนาที

4.2.2 ส่วนแรกรอบการหมุน 5,000 รอบต่อนาที (อุณหภูมิปานกลาง 27 องศาเซลเซียส)



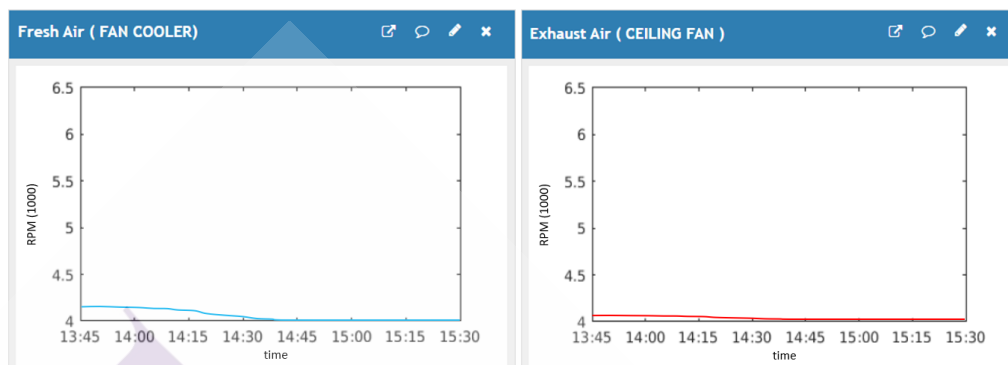
ภาพที่ 4.7 กราฟแสดงรอบการหมุนพัดลมดูดและระบายอากาศรอบสูง 5,000 รอบต่อนาที

4.2.3 ส่วนที่สามรอบการหมุน 4,000 รอบต่อนาที (อุณหภูมิปกติ 25 องศาเซลเซียส)



ภาพที่ 4.8 กราฟแสดงรอบการหมุนพัดลมดูดและระบายอากาศรอบสูง 4,000 รอบต่อนาที

4.2.4 ส่วนสุดท้ายรอบการหมุนเป็นศูนย์ (อุณหภูมิต่ำกว่า 25 องศาเซลเซียส) โดยจะกำหนดให้ระบบหยุดการทำงานทันที



ภาพที่ 4.9 กราฟแสดงรอบการหมุนพัดลมดูดและระบายอากาศรอบเป็นศูนย์

Radian per Minute (%) จะทำหน้าที่แบ่งออกเป็น 3 target group โดยกำหนดให้พัดลมช่วยทำงานในย่านความเร็วรอบ โดยผ่านแพลตฟอร์ม ThingSpeak สำหรับมอนิเตอร์และผ่าน Blynk Application เพื่อควบคุมการใช้งาน ดังต่อไปนี้

RPM: 6000, Blynk write // ThingSpeak: float cuff_t_top >= 30; (ON, Speed 6000 rpm)

RPM: 5000, Blynk write // ThingSpeak: float cuff_t_top >= 27; (ON, Speed 5000 rpm)

RPM: 4000, Blynk write // ThingSpeak: float cuff_t_middle >= 25; (ON, Speed 4000 rpm)

RPM: 0, Blynk write // ThingSpeak: float cuff_t_bottom < 25; (OFF, Stop)

โดยที่ RPM เท่ากับ Radian Per Minute (%) โดยเป็นตัวกำหนดทุกฟังก์ชันเพื่อควบคุมรอบการหมุนของพัดลม

จากตัวอย่างกราฟในรูปที่ 4.5 และ 4.8 สามารถอธิบายได้เป็นลำดับขั้นตอนดังนี้

1. 30C° ถึง 35C° เนื่องด้วยอุณหภูมิเท่ากับหรือมากกว่า 30 องศาเซลเซียส ขึ้นไป ระบบจะประมวลผลสั่งการให้พัดลมฝั่ง Fresh Air และ Exhaust Air ทำงานในรอบสูงที่ 6,000 รอบต่อนาที

2. 27C° ถึง 29C° เนื่องด้วยอุณหภูมิเท่ากับหรือมากกว่า 27 องศาเซลเซียส ขึ้นไปจนถึง 29 องศาเซลเซียส ระบบจะประมวลผลสั่งการให้พัดลมฝั่ง Fresh Air และ Exhaust Air ทำงานในรอบปานกลางที่ 5,000 รอบต่อนาที

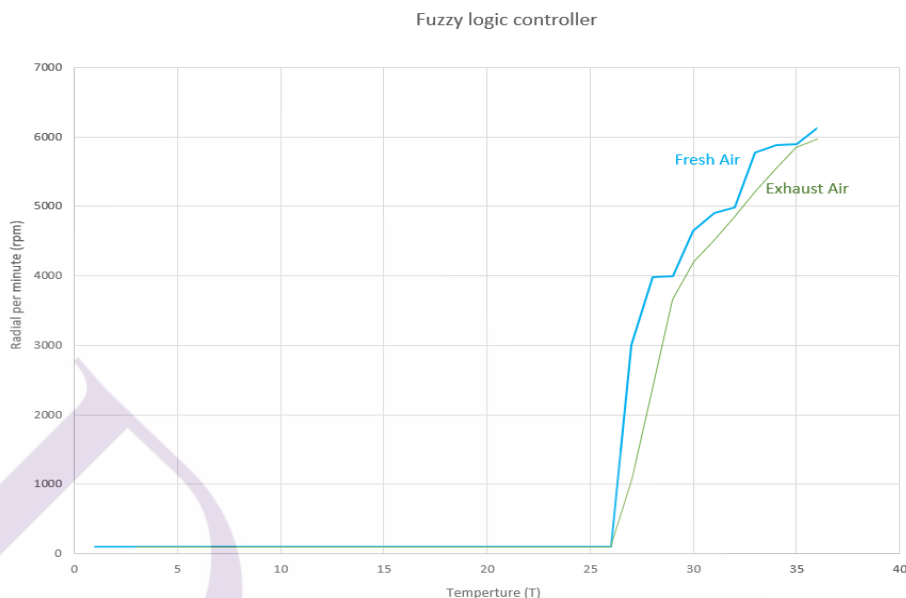
3. 25C° เนื่องด้วยอุณหภูมิเท่ากับ 25 องศาเซลเซียส ระบบจะประมวลผลสั่งการให้พัดลมฝั่ง Fresh Air และ Exhaust Air ทำงานรอบปกติที่ 4,000 รอบต่อนาที

4. 25C° ต่ำลงมา เนื่องด้วยอุณหภูมิเท่ากับหรือต่ำกว่า 25 องศาเซลเซียส ระบบจะประมวลผลสั่งการให้พัดลมฝั่ง Fresh Air และ Exhaust Air หยุดการทำงาน

ตารางที่ 4.2 ข้อมูลสรุปการทดสอบหัวข้อที่ 4.2

Receiver	Device (CPU)	Web Browser (HTTP)	Application (MQTT)	RPM (%)	Maximum (%)	Middle (%)	Minimum (%)
Fresh Air (L298N)	Arduino Uno WiFi	ThingSpeak	Blynk	6000	5980	6034	6089
Exhaust Air (L298N)	Arduino Uno WiFi	ThingSpeak	Blynk	6000	6128	5618	5108
Fresh Air (L298N)	Arduino Uno WiFi	ThingSpeak	Blynk	5000	4899	4561	4224
Exhaust Air (L298N)	Arduino Uno WiFi	ThingSpeak	Blynk	5000	4980	4541	4102
Fresh Air (L298N)	Arduino Uno WiFi	ThingSpeak	Blynk	4000	3890	3485	3080
Exhaust Air (L298N)	Arduino Uno WiFi	ThingSpeak	Blynk	4000	3981	3580	3180
Fresh Air (L298N)	Arduino Uno WiFi	ThingSpeak	Blynk	0	0	0	0
Exhaust Air (L298N)	Arduino Uno WiFi	ThingSpeak	Blynk	0	0	0	0

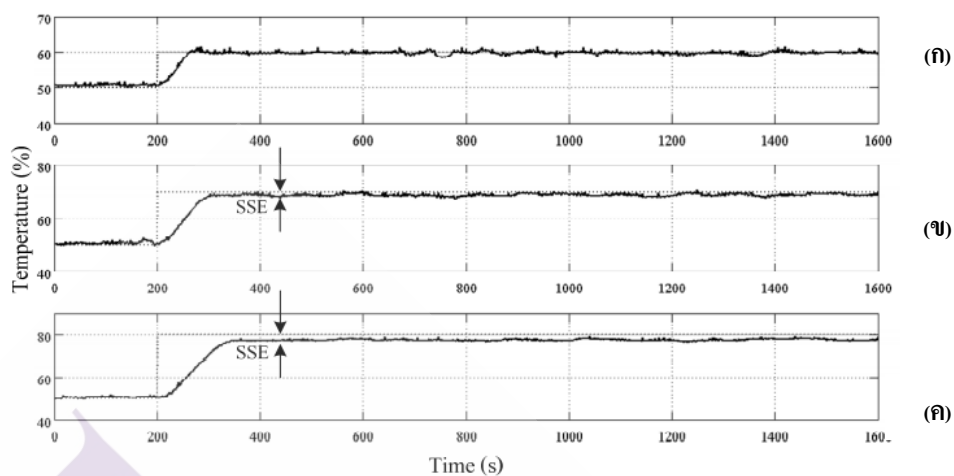
4.3 ผลจากการจำลองการทำงาน พัดลมดูด-ระบายอากาศ ด้วยระบบควบคุมแบบฟัซซี่ลอจิก



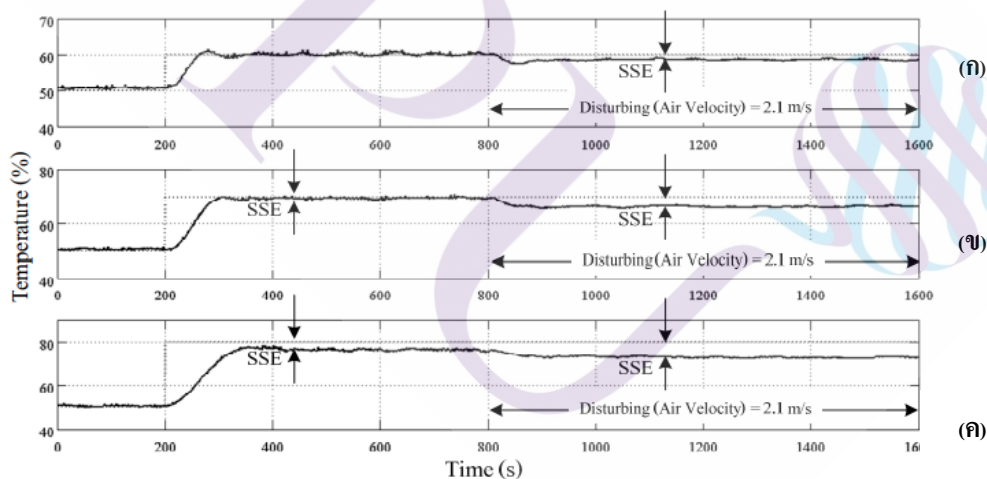
ภาพที่ 4.10 กราฟแสดงช่วงเวลาและรอบการทำงานของพัดลมดูดอากาศ (Fresh Air) และระบายอากาศ (Exhaust Air)

4.4 ผลการทดสอบการควบคุมฟัซซี่ลอจิกและตัวควบคุมอินทิกรัลแบบปรับตัวได้

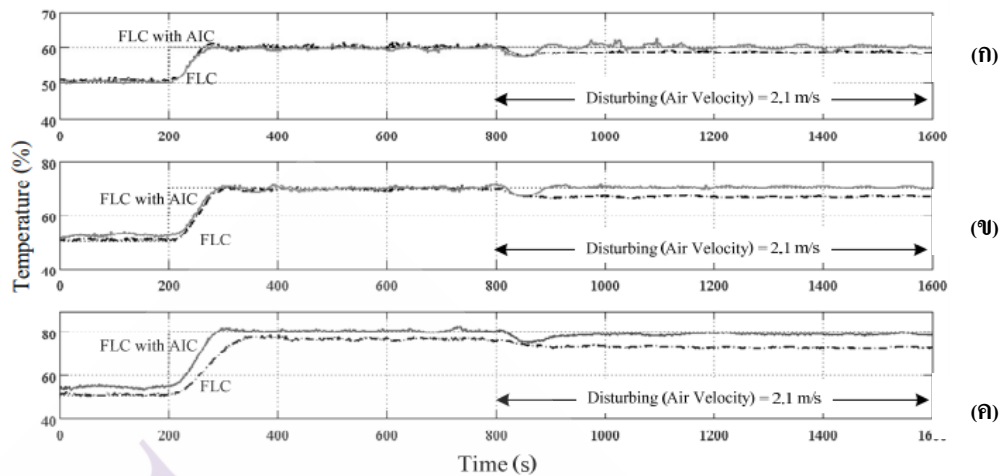
จากการออกแบบตัวควบคุมฟัซซี่ลอจิกและตัวควบคุมอินทิกรัลแบบปรับตัวได้ ตัวควบคุมทั้งสองได้ถูกนำมาทดสอบร่วมกับชุดควบคุมกระบวนการอุณหภูมิแบบการพา สำหรับการช่วงของการควบคุมจะอยู่ระหว่าง 0 ถึง 100 เปอร์เซ็นต์ (ประมาณ 25-35 องศาเซลเซียส) จุดทำงานกำหนดไว้ที่ 50 เปอร์เซ็นต์ (ประมาณ 30 องศาเซลเซียส) ตามที่ได้กล่าวไว้ในหัวข้อที่ 3 และความเร็วลมปกติขณะทดสอบอยู่ที่ประมาณ 1.2 เมตรต่อวินาที การทดสอบประสิทธิภาพของตัวควบคุมจะถูกแบ่งออกเป็น 3 ส่วน โดยส่วนแรกเป็นสถานะที่ระบบมีตัว ควบคุมฟัซซี่ลอจิกเพียงอย่างเดียว ส่วนถัดมาเป็นสถานะที่ระบบถูกรบกวน (ความเร็วลมถูกสมมติเป็นตัวรบกวน) มีค่าการรบกวนสูงสุดอยู่ที่ประมาณ 2.1 เมตรต่อวินาที และส่วนสุดท้ายเป็นสถานะที่ระบบมีตัวควบคุมอินทิกรัล แบบปรับตัวได้ร่วมอยู่ ดังนั้นเพื่อให้เห็นผลการทดสอบประสิทธิภาพของตัวควบคุมชัดเจนยิ่งขึ้นของทั้ง 3 สถานะจะมี การเปลี่ยนแปลงอุณหภูมิอ้างอิงจาก 50 ไปเป็น 60 เปอร์เซ็นต์ 70 เปอร์เซ็นต์ และ 80 เปอร์เซ็นต์ตามลำดับ ผลจากการทดสอบถูกนำมาเปรียบเทียบกันทั้ง 3 สถานะ



ภาพที่ 4.11 ผลการทดสอบระบบด้วยตัวควบคุมพีชซีลอจิก (ก) การเปลี่ยนแปลงอุณหภูมิอ้างอิงจาก 50% เป็น 60% (ข) การเปลี่ยนแปลงอุณหภูมิอ้างอิงจาก 50% เป็น 70% และ(ค) การเปลี่ยนแปลงอุณหภูมิอ้างอิงจาก 50% เป็น 80%



ภาพที่ 4.12 ผลการทดสอบระบบด้วยตัวควบคุมพีชซีลอจิกเมื่อระบบถูกรบกวน (ก) การเปลี่ยนแปลงอุณหภูมิอ้างอิงจาก 50% เป็น 60% (ข) การเปลี่ยนแปลงอุณหภูมิอ้างอิงจาก 50% เป็น 70% และ (ค) การเปลี่ยนแปลงอุณหภูมิอ้างอิงจาก 50% เป็น 80%



ภาพที่ 4.13 ผลการทดสอบระบบด้วยตัวควบคุมพีชชีลอจิกร่วมกับตัวควบคุมอินทิกรัลแบบปรับตัวได้ (ก) การเปลี่ยนแปลงอุณหภูมิอ้างอิงจาก 50% เป็น 60% (ข) การเปลี่ยนแปลงอุณหภูมิอ้างอิงจาก 50% เป็น 70% และ (ค) การเปลี่ยนแปลงอุณหภูมิอ้างอิงจาก 50% เป็น 80%

ผลจากการทดสอบประสิทธิภาพของตัวควบคุมพีชชีลอจิกร่วมกับตัวควบคุมอินทิกรัลแบบปรับตัวได้ของทั้ง 2 สถานะพบว่าในกรณีที่ระบบมีตัวควบคุมพีชชีลอจิกเพียงอย่างเดียว ค่าความผิดพลาดของอุณหภูมิจะยังคงมีอยู่ในระดับหนึ่งและจะเห็นได้ชัดที่สุดเมื่อช่วงของอุณหภูมิอ้างอิงเปลี่ยนแปลงจาก 30 องศาเซลเซียส เป็น 35 องศาเซลเซียส เนื่องจากการปรับแต่งระบบฐานกฎของพีชชีลอจิกอาจมีความไม่เหมาะสม และหากมีการเพิ่มความเร็วลม การรักษาเสถียรภาพของระบบจะเห็นว่า ค่าความผิดพลาดของอุณหภูมิที่เกิดขึ้นอยู่ในระดับที่สูง ผลตอบสนอง ของอุณหภูมิแสดง ได้ดังรูปที่ 4.11 แต่เมื่อมีการใช้ตัวควบคุมอินทิกรัลแบบปรับตัวได้เข้ามาเป็นตัวควบคุมร่วมกับตัวควบคุมพีชชีลอจิกพบว่า การรักษาเสถียรภาพของระบบทำได้ดี มีค่าความผิดพลาดที่ต่ำ เนื่องจากตัวควบคุมดังกล่าวมีจุดเด่นในการแก้ปัญหาเรื่องความผิดพลาดของระบบ

ผลตอบสนอง ของอุณหภูมิแสดง ได้ดังรูปที่ 4.12 จากผลการทดสอบทั้งสองส่วนถูกเปรียบเทียบกันระหว่างกรณีที่ใช้ตัวควบคุมพีชชีลอจิกเพียงอย่างเดียวกับกรณีที่ใช้ตัวควบคุมพีชชีลอจิกร่วมกับตัวควบคุมอินทิกรัลแบบปรับตัวได้ ผลตอบสนองของอุณหภูมิแสดง ได้ดังรูปที่ 4.13

จากที่กล่าวมาข้างต้นสามารถสรุปและแสดงผล ได้ดังตารางที่ 4.3 พบว่าระบบควบคุมที่ใช้ตัวควบคุมพีชชีลอจิกร่วมกับตัวควบคุมอินทิกรัลแบบปรับตัวได้ มีค่าความผิดพลาดในสถานะคงตัวต่ำกว่าระบบควบคุมที่มีตัวควบคุมพีชชีลอจิกเพียงอย่างเดียว ค่าความผิดพลาดที่เกิดขึ้น

จะยังอยู่ในช่วงขอบเขตที่ได้กำหนดไว้เท่ากับ $\pm 2\%$ ทั้ง ในกรณีถูกรบกวนและไม่ถูกรบกวน ทั้งนี้ ไม่เพียงแต่ลดความผิดพลาดของอุณหภูมิ แต่ยังพบว่าช่วงเวลาไต่ขึ้นของระบบควบคุมที่ใช้ตัวควบคุมพีซีลอจิกร่วมกับตัวควบคุมอินทิกรัลแบบปรับตัวได้ มีผลตอบสนองของ อุณหภูมิที่รวดเร็วกว่าอีกด้วย

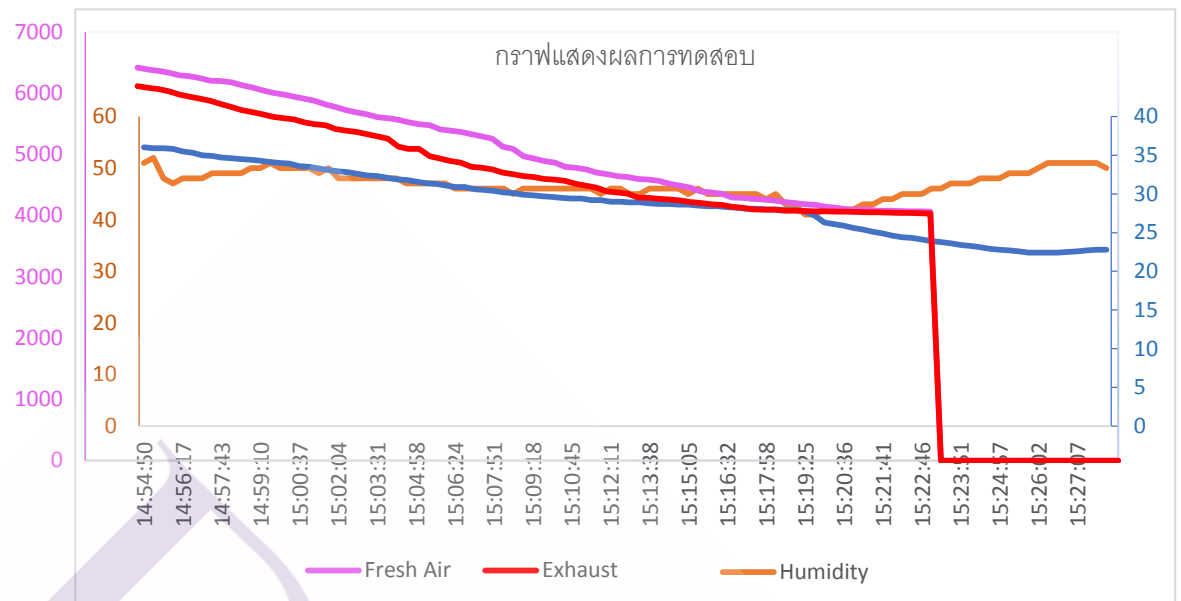
ตารางที่ 4.3 การเปรียบเทียบค่าความผิดพลาดระหว่างตัวควบคุมพีซีลอจิกและตัวควบคุมพีซีลอจิก ร่วมกับตัวควบคุมอินทิกรัลแบบปรับตัวได้

Controller \ Condition	Non-Disturbing System						Disturbing System		
	Rising Time (s)			Steady State Error (%)			Steady State Error (%)		
	Step Response (%)			Step Response (%)			Step Response (%)		
	50-60	50-70	50-80	50-60	50-70	50-80	50-60	50-70	50-80
FLC	47	49	93	+0.1	+2.6	+4.3	+2.4	+4.1	+7.7
FLC with Adaptive Integral Control	44	45	50	+0.3	+0.2	-0.6	+0.2	+0.1	+0.9

4.5 ผลการแสดงความสัมพันธ์ระหว่างรอบการหมุนของพัดลมและตัววัดอุณหภูมิ

เป้าหมายของการนำ fuzzy logic มาใช้เพื่อทำให้ระบบควบคุมเข้าสู่ความเย็นสบายเร็วขึ้น โดยผลลัพธ์จะแสดงค่าความสัมพันธ์ระหว่างความเร็วรอบที่ช้าลงของพัดลมดูดอากาศและระบายอากาศ แบบอัตโนมัติ ด้วย fuzzy logic โดยประมวลผลด้วยอุณหภูมิภายในที่ลดลง ซึ่งเป็นตัวแปรสำคัญในการกำหนดความเร็วรอบของพัดลม จากหมุนเร็วให้หมุนช้าลงตามเงื่อนไขของอุณหภูมิที่ต่ำลง จนหยุดการทำงานดังแสดงผลลัพธ์ในรูปที่ 4.14 กราฟประมวลผลรวม fuzzy logic แสดงความสัมพันธ์ของอุณหภูมิที่ลดลง และ ความเร็วรอบของพัดลมที่ช้าลง

โดยระบบพร้อมที่จะส่งต่อให้เครื่องปรับอากาศ ทำงานในฝั่งของการทำความเย็นที่ต่ำกว่า 25 องศาเซลเซียส เพื่อเป็นการลดภาระการทำงานของระบบปรับอากาศในช่วงเวลาเริ่มต้นของการใช้งาน โดยไม่ต้องคำนึงถึงความร้อนที่สะสมภายในห้อง ซึ่งข้อดีคือจะทำให้เกิดความเย็นสบายได้อย่างรวดเร็ว และจะเป็นข้อดีในการประหยัดพลังงานในอนาคต



ภาพที่ 4.14 กราฟประมวลผลรวม fuzzy logic แสดงความสัมพันธ์อุณหภูมิที่ลดลงและความเร็วรอบของพัดลมที่ช้าลง

4.6 ผลกราฟแสดงเปรียบเทียบระดับการประหยัดพลังงานของระบบ Fuzzy และ Normal

ผลการประหยัดพลังงาน ซึ่งคิดหน่วยวัตต์เป็น วัตต์ จะพบว่าสถานะการทำความเย็นของระบบเครื่องปรับอากาศ มีความรวดเร็วขึ้น และระดับการใช้พลังงานลดลง โดยสามารถเปรียบเทียบข้อมูลจากกราฟ ในรูปที่ 4.15 และให้เห็นความแตกต่างระหว่างการใช้งานของระบบ Fuzzy และ Normal ซึ่งสามารถอธิบายความสัมพันธ์บนกราฟที่แสดงระดับของการประหยัดพลังงานได้ดังต่อไปนี้

ระบบ Normal และ Fuzzy เมื่ออ้างอิงจากข้อมูลบนกราฟ จะพบว่าได้ใช้พลังงาน 1181 วัตต์ โดยใช้เวลาประมาณ 30 นาที โดยสามารถคำนวณผลลัพธ์การใช้พลังงานแอร์ 13,000 BTU/hour ออกมาได้ดังนี้

ในช่วงที่ 1 ; เวลา 14:54:50 ระบบทำงานเต็มที่

$$\text{Fuzzy} \quad 1181 \text{ วัตต์} \times 30 \text{ นาที} / 60 \text{ นาที} = 0.59 \text{ kWh}$$

$$\text{Normal} \quad 1181 \text{ วัตต์} \times 30 \text{ นาที} / 60 \text{ นาที} = 0.59 \text{ kWh}$$

ในช่วงที่ 2 ; เวลา 15:10:00 – 16:00:00 ระบบเริ่มมีการใช้พลังงานแตกต่างกัน เมื่อเปรียบเทียบข้อมูลบนกราฟ โดยใช้ค่า EER (Energy Efficiency Ratio) เป็นตัวกำหนดดังนี้

$$\text{ระบบของ Fuzzy} \quad \text{จะมีค่า EER (ในช่วงโหลดต่ำประมาณ 75\%)} = 19$$

ระบบของ Normal จะมีค่า EER = 11 * ตัวแปรมาตรฐาน

เช่น เปิดแอร์ ขนาด 13,000 BTU / hour ที่มีค่า EER = 11 ต่อ 1 ชั่วโมง และใช้พลังงานเท่ากับ $13,000 / 11 = 1181.82$ Wh เรียกว่า แอร์ตัวนี้กินไฟที่ 1181.82 วัตต์ ดังนั้นจึงสามารถนำมาคำนวณหาค่าการใช้พลังงานได้ดังต่อไปนี้

แบบ Fuzzy ใช้พลังงานไป 300 วัตต์ x 8 ชม. / 1000 วัตต์ = 2.40 kWh

$EER \times Wh = 19 \times 2.40 \times 1000 = 45,600$ BTU

แบบ Normal ก็จะต้องใช้พลังงานไป 45,600 BTU เช่นกันเพื่อรักษาอุณหภูมิที่เท่ากัน ดังนั้นระบบจึงต้องใช้พลังงานเท่ากันดังนี้

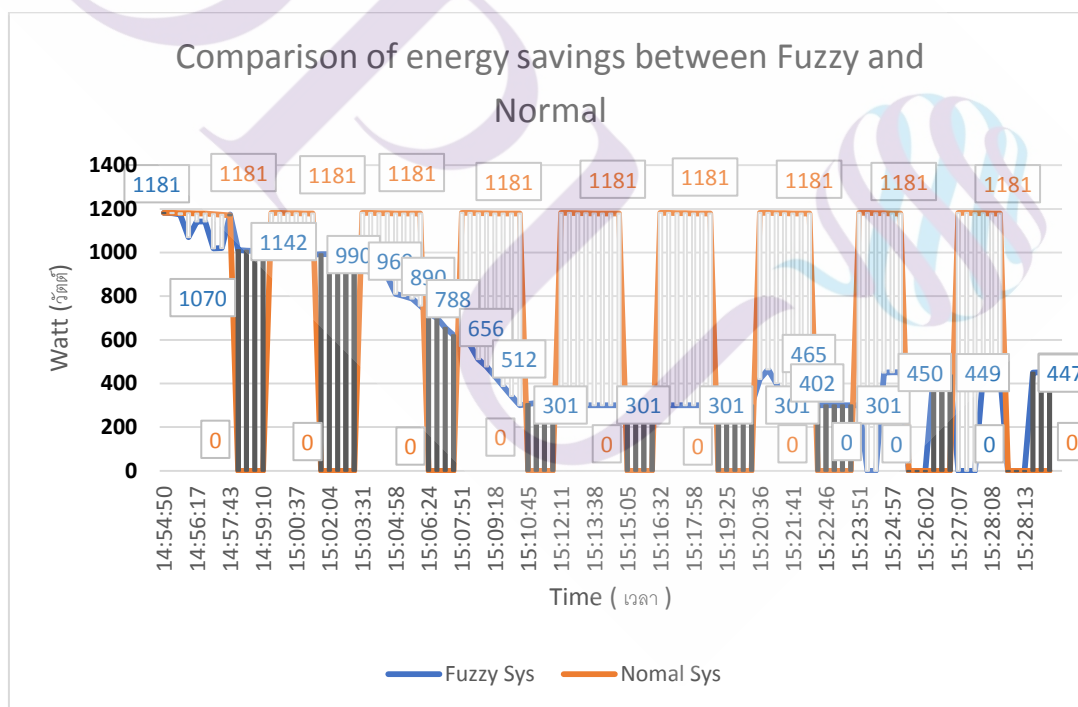
$BTU / EER = (45,600 / 11) / 1000 = 4.15$ kWh

สรุปผลเปรียบเทียบพลังงาน

- การใช้พลังงานแบบระบบ Fuzzy = 0.59 + 2.40 = 2.99 kWh

- การใช้พลังงานแบบระบบ Normal = 0.59 + 4.15 = 4.74 kWh

คิดเป็นผลประหยัดพลังงาน ได้ที่ $((4.74 - 2.99) / 4.74 \times 100) / 10000 = 36.9\%$



ภาพที่ 4.15 กราฟเปรียบเทียบระดับการประหยัดพลังงานระหว่างระบบ Fuzzy และ Normal

บทที่ 5

สรุปผลการทดลองและข้อเสนอแนะ

เนื้อหาในบทนี้จะเป็นการอภิปรายเพื่อสรุปผลที่ได้จากการทดลองงานวิจัย รวมทั้งข้อจำกัดของระบบที่พบจากการทดลองระบบ และข้อเสนอแนะสำหรับแนวทางในการพัฒนา งานวิจัยนี้ต่อไปเพื่อแก้ไขข้อบกพร่องของระบบให้มีประสิทธิภาพมากยิ่งขึ้น

5.1 สรุปผลการวิจัย

ในวิจัยนี้ตัวควบคุมแบบฟัซซี่ลอจิก (fuzzy logic controller) ถูกใช้ในการจำลองการควบคุมการทำงานของเซ็นเซอร์วัดอุณหภูมิในอากาศและการดูระบายอากาศในอาคาร โดยผลลัพธ์ที่ได้รับแสดงถึงการควบคุมที่มีความแม่นยำและมีประสิทธิภาพ ดังนั้นตัวควบคุมแบบฟัซซี่ลอจิก (fuzzy logic controller) สามารถนำไปใช้ควบคุมระบบระบายอากาศและการถ่ายเทอากาศในอาคาร ตัวควบคุมฟัซซี่ลอจิก (fuzzy logic controller) ที่คณะผู้วิจัยนำเสนอมีการทำงานที่มีความแม่นยำสูง (high precision) เสถียรภาพที่ดี (stability) และตอบสนองรวดเร็ว (fast response) โดยเมื่อมีการใช้ระบบนี้แล้ว ส่งผลให้เกิดการพัฒนาาระบบให้สอดคล้องตามวัตถุประสงค์ดังนี้

5.1.1 พัฒนาระบบไหลเวียนอากาศด้วยพัดลมอัตโนมัติโดยการประยุกต์ใช้งานจากเทคโนโลยีของอินเทอร์เน็ตสรรพสิ่งให้เกิดความสะดวกรวดเร็วควบคุมผ่านระบบได้

5.1.2 พัฒนาระบบควบคุมแบบพลวัต โดยใช้ Fuzzy logic ให้เกิดการประหยัดพลังงานในการควบคุมสภาพแวดล้อมภายในห้องเป็นการลดภาระการทำงานของเครื่องปรับอากาศเมื่อต้องการอากาศที่เย็นสบาย

5.1.3 พัฒนาส่วนการเชื่อมต่อ Gateway ให้ทำหน้าที่จัดส่ง ค่าวัดอุณหภูมิจากระบบเครือข่าย โดยใช้โปรโตคอลที่แตกต่างกัน เช่น MQTT และ HTTP protocols ระหว่าง Blynk แอปพลิเคชัน และ ThingSpeak เว็บเบราว์เซอร์

5.1.4 พัฒนาส่วนแสดงผลต่อผู้ใช้งานผ่านแอปพลิเคชัน Blynk บนสมาร์ตโฟน

5.2 ปัญหาที่พบและข้อเสนอแนะ

5.2.1 ปัญหาที่พบในงานวิจัย

5.2.1.1 ปัญหาที่พบในระหว่างการทำงานของระบบระบายอากาศ จะยังคงมีความไม่แม่นยำสำหรับเซ็นเซอร์การตรวจวัดอุณหภูมิ เนื่องจากเป็นอุปกรณ์ที่มีราคาถูก แต่สามารถนำมาใช้ในทดลองได้ เพื่อให้เป็นไปตามวัตถุประสงค์

5.2.1.2 ระบบ fuzzy logic มีความสลับ ซับซ้อนในการใช้งาน จึงต้องอาศัยระยะเวลาในการพัฒนา และทดสอบร่วมกัน กับระบบควบคุมให้มีความหลากหลายและเงื่อนไขที่ดี ในการนำมาประมวลผลให้ให้มีประสิทธิภาพมากขึ้น

5.2.1.3 ส่วนการเชื่อมต่อของ Gateway และระบบเครือข่าย ที่ใช้โปรโตคอล ที่แตกต่างกัน ยังมีข้อจำกัดในเรื่องการแสดงผลที่ไม่แม่นยำมากนัก โดยเฉพาะเรียลไทม์แสดงผลค่าวัดอุณหภูมิและรอบการหมุนของใบพัดลม

5.2.1.4 ส่วนของแอปพลิเคชัน เป็นแบบใช้งานทั่วไป มีฟังก์ชันที่ไม่สามารถนำมาพัฒนาต่อได้ จึงจำเป็นต้องมองหา แอปพลิเคชันที่สามารถนำมาพัฒนาให้มีความหลากหลายในเรื่องของการออกแบบ และการเชื่อมต่อกับ โปรโตคอลอื่นๆ

5.2.2 ข้อเสนอแนะ

จากผลการทดลอง เป็นจุดเริ่มต้นและแนวคิดของระบบปรับปรุงสภาพอากาศภายในบ้านแบบพลวัต โดยใช้เทคโนโลยีอินเทอร์เน็ตของสรรพสิ่ง ซึ่งมีความสามารถที่จะนำไปพัฒนาให้เป็นต้นแบบต่ออุตสาหกรรมดังต่อไปนี้

5.2.2.1 เป็นต้นแบบในการพัฒนาและปรับปรุงห้องผ่าตัด และห้องรักษาผู้ป่วยติดเชื้อแบบห้องความดันลบ โดยการทำงานแบบอัตโนมัติ และพัฒนาขีดความสามารถในการตรวจวัดอุณหภูมิผู้ป่วย เพื่อช่วยการตัดสินใจของแพทย์และพยาบาล ในการบริหารจัดการต่อคนไข้

5.2.2.2 เป็นต้นแบบของการรักษาและควบคุมอุณหภูมิในการถนอมอาหาร สด แช่แข็ง เพื่อให้ทราบถึงสถานะของอาหารและสามารถสั่งการและแก้ไขเมื่อเกิดปัญหาได้

5.2.2.3 เป็นต้นแบบของอุตสาหกรรมยาในการควบคุมอุณหภูมิ และความสะอาด ก่อนเริ่มการผลิตและตรวจสอบตามขั้นตอนของมาตรฐาน QC

5.2.2.4 เป็นต้นแบบในการพัฒนาระบบอากาศ สะอาดปลอดฝุ่นในกระบวนการผลิตชิ้นส่วน และสมองกลไอซี หรือชิพ ประมวลผล ในอุตสาหกรรมเซมิคอนดักเตอร์ ให้สะอาดซึ่งเป็นหัวใจหลักของการผลิตและจำหน่าย



บรรณานุกรม

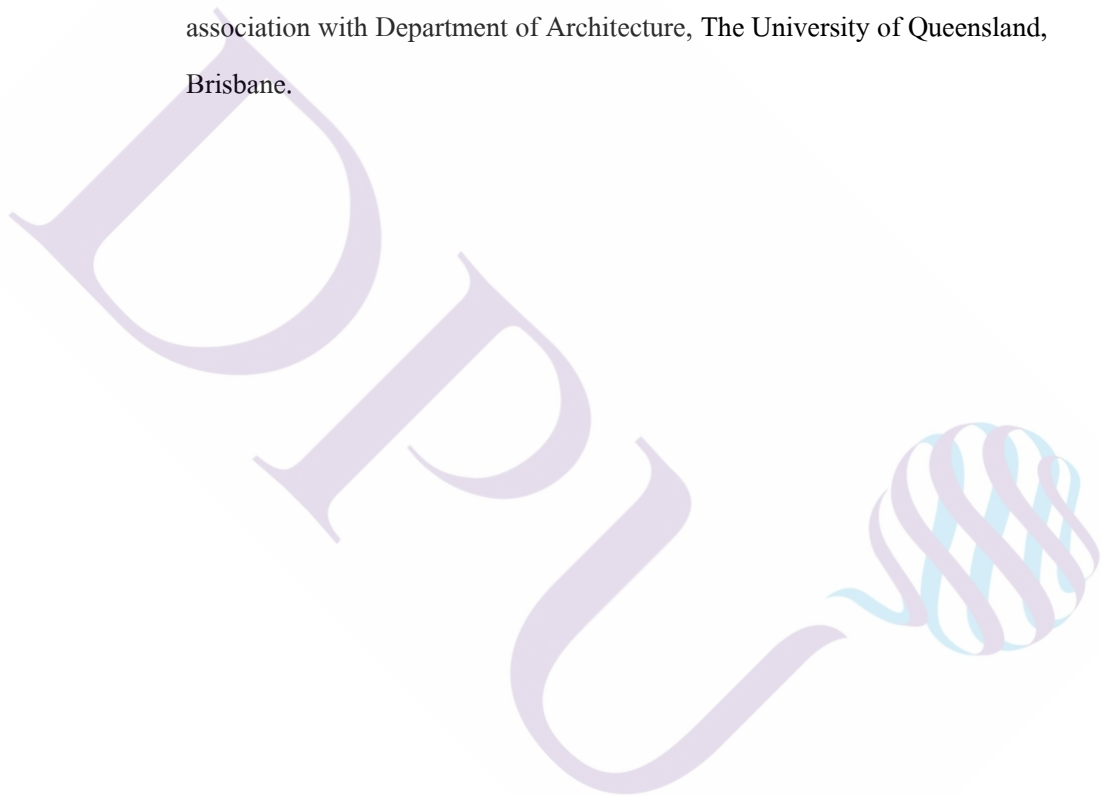
บรรณานุกรม

- [1] J. Gubbi, R. Buyya, S. Marusic and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Generation Computer Systems*, Elsevier Science Publishers, September, 2013.
- [2] D. Bandyopadhyay and J. Sen, "Internet of Things: Applications and Challenges in Technology and Standardization", *J. Wireless Pers Commun*, Kluwer Academic Publishers, Hingham, MA, USA, May 2011.
- [3] Arduino, Arduino uno (2014), <http://arduino.cc/en/Main/arduinoBoardUno>.
- [4] M. Margolis, Arduino Cookbook, O'Reilly Media, Inc., 2011.
- [5] P. LechEmail and P. Włodarski, "Analysis of the IoT WiFi Mesh Network", In: Cybernetics and Mathematics Applications in Intelligent Systems (CSOC 2017), *Advances in Intelligent Systems and Computing*, vol 574, Springer, 2017.
- [6] X. R. Jiang, Y. M. Lv, X. H. Cheng, "Design of Wireless Communication System Based on nRF24L01", *Advanced Materials Research*, Vols. 945-949, pp. 1756-1759, 2014.
- [7] P. Mell and T. Grance, "The NIST Definition of Cloud Computing", *National Institute of Standards & Technology*, Gaithersburg, MD, United States, 2011.
- [8] Amazon Web Service (2017), <https://aws.amazon.com>
- [9] Amazon Web Services (2017), "AWS IoT", <https://aws.amazon.com/iot>.
- [10] J. Clark, "Amazon switches on DynamoDB cloud database service", ZDNet, 201
- [11] "MQTT 3.1.1 specification", OASIS, December 10, 2015.
- [12] A. Stanford-Clark and H. L. Truong, "MQTT For Sensor Networks (MQTT-SN) Protocol Specification Version 1.2", mqtt.org, p. 27, November 14, 2013.
- [13] Myhrvold, A. N., Olsen, E., & Lauridsen, Ø. (1996). Indoor environment in schools – Pupils health and performance in regard to CO2 concentrations. Proceedings of Indoor Air' 96: the 7th International Conference on Indoor Air Quality and Climate, (pp. 369 – 374,). Nagoya, Japan.
- [14] Muangkaew, K. (2016). Indoor air quality in office buildings. Journal of Department of Science Service Ministry of Science and Technology, 1-14.

บรรณานุกรม (ต่อ)

- [15] Northern Meteorological Center. (2010). Climatological data for period 1981-2010 (Report). Chiang Mai, Thailand: Meteorological Department.
- [16] Olgyay, V. (1963). Design with climate: Bioclimatic approach to architectural regionalism (4th ed.). New Jersey: Princeton University Press.
- [17] Santamouris, M., Synnefa, A., Assimakopoulos, M., Livada, I., Pavlou, K., Papaglastra, M., Gaitani, N., Koloskotsa, D., & Assimakopoulos, V. (2008). Experimental investigation of the air flow and indoor carbon dioxide concentration in classrooms with intermittent natural ventilation. *Energy and Buildings*, 40, 1833-1843.
- [17] Sawatmongkhonkul, N., & Moorapun, C. (2015). Perception of interior air quality in public buildings. *Veridian E-Journal, Slipakorn University*, 8(2), 1583-1594.
- [18] Schibuola, L., Scarpa, M., & Tambani, C. (2016). Natural ventilation level assessment in a school building by CO2 concentration measures. *Energy Procedia* 101, 71st Conference of the Italian Thermal Machines Engineering Association, ATI2016, 14-16 September 2016, (pp. 257-264). Turin, Italy.
- [18] Swiss Standards [SIA]. (1992). SIA 328/1: Technical requirements for ventilation systems. Zurich: Schweizerische Normen Vereinigung.
- [19] The Engineering Institute of Thailand (2008). *Building Audit Techniques for Safety*, Engineering Institute of Thailand: Bangkok, pp. 142-14.
- [20] The Engineering Tool Box. (2016). Air change rates for typical rooms and buildings. Retrieved December 8, 2016 from http://www.engineeringtoolbox.com/air-change-rate-room-d_867.html.
- [21] Turunen, M., Toyinbo, O., Putus, T., Nevalainen, A., Shaughnessy, R., & Haverian-Shaughnessy, U. (2014). Indoor environmental quality in school buildings, and the health and wellbeing of students. *International Journal of Hygiene and Environmental Health*, 217, 733-739.
- [22] Waeochan, N. (2017). Interior Architectural Student's Learning Behavior: Case Study of the KMITL's Interior Architecture Curriculum. *Journal of the Faculty of Architecture King Mongkut's Institute of Technology Ladkrabang*, 25(2), 107-123.

- [23] American Society of Heating, Refrigerating and Air-Conditioning Engineers [ASHRAE]. (1999). ASHRAE 62- Ventilation for acceptable indoor air quality. Atlanta: Author.
- [24] American Society of Heating, Refrigerating and Air-Conditioning Engineers [ASHRAE]. (2010). ASHRAE 62.1 standard thermal environmental conditions for human occupancy. Atlanta: Author.
- [25] Auliciems, A., & Szokolay, S. V. (2007). Thermal comfort. PLEA Notes: Passive and low energy architecture international, design tools and techniques (2nd revision ed.). In association with Department of Architecture, The University of Queensland, Brisbane.





ภาคผนวก



ICEMI 2017



Certificate of Appreciation

This Committee is pleased to present this certificate to

Phumipat Niyomtham

in recognition and appreciation on the valued contribution as

Speaker

in

2017 2nd International Conference on Economics and Management Innovations

Organizing Committee of ICEMI 2017
15-16, July, 2017
Bangkok, Thailand





Contents List available at VOLKSON PRESS
Economics & Management Innovations(ICEMI)

ISBN: 978-1-948012-02-7



Auto Active Airflow-Fan Cooler System Using Internet of Things Technology

Phumipat Niyemtham, Narongdech Keeratipranon and Ratthaslip Ranokphanuwat

* College of Innovative Technology and Engineering Dhurakij Pundit University Bangkok Thailand

*Email: udem.ran@su.ac.th

ARTICLE DETAILS

Article History

Received 02 October 2017

Accepted 06 October 2017

Available online 11 October 2017

Keywords:

Internet of Things, Amazon Web Service, Cloud technology, Sensor, Gateway, Cooling system.

ABSTRACT

This paper is a study and development of a cooling system in automatic control to help solve common household problems, called an Auto Active Airflow-Fan Cooler System, especially in a city surrounded by concrete forests, which are very hot during the daytime. Moreover, heat transfers into a building, for example, a wall, floor or a window, and then kept inside the building. As a result, the building accumulates heat. In addition, the air conditioning results in slower working conditions and uses greater power consumption. The Auto Active Airflow-Fan Cooler System has a mechanism that allows for good ventilation, where windows are closed. The system will automatically run when the house temperature reaches a set point. The evening garden system will continue to work to provide effective ventilation, and will drip during the night, when the temperature in a building drops to a set level. If the Auto Active Airflow-Fan Cooler System is installed in conjunction with the ceiling insulation, it will help to improve the performance. The system developed using Internet of Things (IoT) and Cloud (AWS) technology. The IoT is designed and implemented using low cost Arduino + NRF240L1 technology for automatic temperature measurement, cooling the air in the building, and transmitting/receiving information to/from Cloud via the internet. Cloud (AWS) is used to operate automatically and deciding to run a ventilated fan works. It includes AWS IoT, DynamoDB and EC2 services. The system features are low cost. This technology automatically monitors manages the energy inside the building via web applications and mobile phones.

1. Introduction

Thailand is one of the world's top ten hotspots. Innovative home cooling, or "cool home", is a good solution. The overall climate trend is getting warmer every year. Good house design in today's climate is a process that can reduce the house temperature. At present, many construction material suppliers choose to innovate that can help keep the house cool in the cold, such as roofing materials, thermal insulation and aerated brick wall. Moreover, the design of the house in accordance with wind direction and sunlight becomes more important. However, there are limits to the design when changing lifestyles, especially of a houses located with

designs such that, they cannot open.

Nowadays, the innovation of ventilation and cooling of a house and the roof becomes too expensive. We have invented the new Active Air Flow System that keeps the home temperature cool. In this paper, we focus on experiments on simulators to study the process and work process through the Internet of Things (IoT) and Cloud (AWS) technology. Our contributions are as follows:

1. To design and develop an automatic cooling control system.
2. To design and develop a system for controlling and displaying fan on and off via web application or mobile application.

The rest of the paper is organized as follows. Section 2 presents the model and discusses the implementation details. Section 3 describes the results and discussions. Finally, Section 4 concludes and suggests future work.

2. The Auto Active Airflow-Fan Cooler Model

The ventilation system is designed to have an external air intake by means of a fan motor controlled by the temperature installed inside. There is a vent for exhaust air at the top, with a temperature sensor that determines the speed of the fan, both above and below. If the temperature is less than 32 degrees, the system will not work. When the temperature is more than

35 degrees, the system will control the fan to take the air out and drain it. At a speed of 50 rpm, and when the temperature exceeds 45 degrees, the system speeds up to 200 rpm, with the fan having a separate control knob to separate the two fans. Working on both nodes sends the temperature and fan speed to the Node MCU to send up the AWS IoT.

In the case of AWS IoT data, there is a rule to control the value stored in the DynamoDB data base. The stored value is the temperature and fan speed with the timestamp specified. The Node MCU sends the value to the Cloud when the temperature changes. To save energy, the benefit of the following steps is as shown in Figure 1



Figure 1. Active Airflow-Fan Cooler Model Overview

The model consists of three key components (according to the IoT concept [1][2]), where each component is assigned to any functional. The Arduino (Device) received sensor value from any sensor, after that send data to NodeMCU and NodeMCU (Gateway) send data to AWS Cloud, with MQTT protocol, which the data store in DynamoDB and executed by EC2 (Compute Cloud Service) on web browser with HTTP protocol as shown in Figure 2.



Figure 2. Auto Active Airflow-Fan Cooler model details

2.1 System Architecture

The main components used in this system are divided into three parts: Devices, Gateway, AWS Cloud computing, and Data transmission.

2.1.1 Devices

1) DHT11 Humidity/Temperature sensor, this module measures the surrounding air, and sends out a digital signal. It is calibrated with digital output of the temperature and humidity sensor. Accuracy for humidity is 20-80% and temperature is 0-50° C. This sensor is used for measuring moisture content in the atmosphere. The sensor is shown in Figure 3.



Figure 3. DHT11 Humidity and Temperature Sensor

2) KEYES L9110 Fan Module is for Arduino and experiments on DC motor, and uses IC L9110 as motor drive. This module is used for air controlling simulation, and is shown in Figure 4.



Figure 4. KEYES L9110 Fan Module

3) Arduino is a microcontroller board with open-source physical computing platform based on a simple I/O board that can connect to electrical circuits [3-4]. Arduino uses an ATmega328P chip as a microcontroller. We use it for Sensor Node to collect data, and it is networked by a NRF24L01 wireless communication system. The Arduino board is shown in Figure 5.



Figure 5. Arduino board

2.1.2 Gateway

An IoT gateway device is the communication between IoT devices, Sensors, Gateway and AWS cloud. By connecting the field device and AWS cloud, the IoT gateway devices operate local processing and storage solutions, has the ability to automatically control field devices based on data input by sensors. Node MCU is chosen for our purpose. It is a development board and part of the Internet of Things (IoT) [5]. The Node MCU board is based on ESP8266EX [6], which is a microcontroller integrated Wi-Fi receiver, and transmitter. The Node MCU board is shown in Figure 6.



Figure 6. Node MCU

2.1.3 AWS Cloud computing

Cloud computing is a model that provides convenient, on-demand network access to a shared pool resources related functionality as a service (networks, servers, storage, applications) [7]. We choose Amazon Web Services (AWS) [8] which is a Public cloud type for the experiments. The AWS Cloud model is shown as Figure 7.

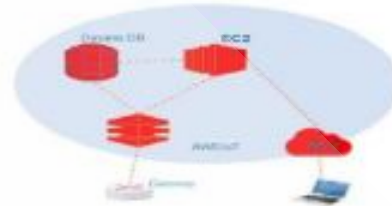


Figure 7. AWS Cloud Computing

The AWS Cloud model includes 1) Amazon Elastic Compute Cloud (EC2) for Application service and Web server; 2) AWS IoT is a IoT hub that connects with Gateway easily and securely to interact with Application service on EC2 [9]; with AWS IoT, sensor values can be transferred to store on DynamoDB; 3) DynamoDB [10] is a NoSQL database for collecting various sensor data (Temperature, Humidity and Fan status). These data will be stored in JSON format. In DynamoDB, we use NodeID as primary keys to identify uniquely each item in a table and Timestamps as a secondary index to provide greater querying flexibility.

2.1.1 Data transmission

Another feature of IoT is that the user can control and manage things from a wireless remote distance, and also on PC or mobile.

1) **Monitoring:** NodeMCU receives parameters detected by any Sensors from the Sensor node, and passes through the MQTT protocol to AWS IoT and transfers the data to DynamoDB. The monitoring data transmission is shown in Figure 8.



Figure 8. Monitoring Data Transmission

2) **Controlling:** Figure 11 shows the controlling data transmission. The user can control Fan via Web browser. If the user switches to the

control function, EC2 (Application service + Web server) will send the request to AWS IoT, and pass into NodeMCU and Sensor node boards to enable the Fan. The controlling data transmission is shown in Figure 9.

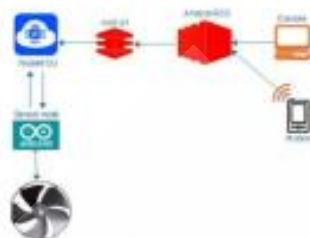


Figure 9. Controlling Data Transmission

2.2 Protocols

IoT requires a standard protocol. This system uses MQTT protocol [11][12]. MQTT is a machine-to-machine protocol. It is an extremely lightweight Publish/Subscribe messaging transport, as shown in Figure 10. It is useful for connections with remote locations, where a small code footprint is required with low network bandwidth (small size, low power usage, minimized data packets). For our model, it has been used in Gateway communicating to AWS IoT via the internet. In this work, Gateway is assigned to the Publisher, and AWS IoT to be the MQTT broker, while Application service is the Subscriber.



Figure 10. MQTT Protocol Model

3. Results and Discussion

Based on our experiments, we have simulated additional heated temperatures. By using a hair dryer, the temperature sensor is over 35 degrees. The motor must be driven at a speed of 50, and greater than 45 degrees at 200 cycles. The motor can work well. When testing with NodeMCU, the NodeMCU cannot receive some values. However, this case occurs infrequently. Web browser displays the error message, as shown in Figure 11.

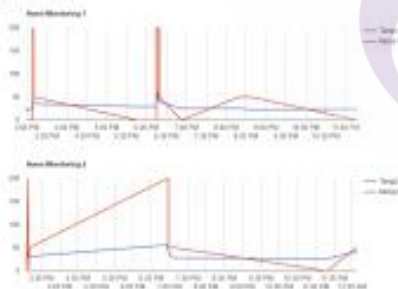


Figure 11. Motor constantly

Sometimes, it is necessary to check NodeMCU for correcting values. When we press the reset button, these symptoms will disappear. For communications testing, the same communication channel is used without collision detection. NodeMCU cannot accept the values sometimes Home Monitoring 2 graph will notice that the Motor2 value is constant from 3:30 PM to 7:00 PM, that is, the motor level increases from 50 to 200, which may not be accurate.

Referring to the information obtained from the results, we discussed in each subsystem where of was found that the overall systems, worked, as described in the manual. However, some improvements, such as position and number of sensors installed, may affect the accuracy of the measured data. Proper positioning and numbering should ensure that the measured data are accurate to meet the requirements.

4. Conclusions

This paper presented an IoT based designed of the Auto Active Airflow-Fan Cooler system. The model operates for data gathering and transmission using wireless internet under the Amazon Web Service, and preliminary tests, showed that the developed model was capable of monitoring and controlling devices in the environment, and had several advantages in terms of fast delivery, zero data loss, low cost, flexibility, and user friendliness. The user interfaces are simple and intuitive. The proposed project helps residents in real time monitoring of temperatures, and also reduces cost and expense for electricity.

For future work, this work can be applied, not just to ventilation, but also other projects, such as plant control, agricultural water control systems, and water monitoring systems, etc.

References

- [1] J. Gubbi, R. Buyya, S. Marusic and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Generation Computer Systems*, Elsevier Science Publishers, September, 2013.
- [2] D. Bandyopadhyay and J. Sen, "Internet of Things: Applications and Challenges in Technology and Standardization," *J. Wireless Pers Commun*, Kluwer Academic Publishers, Hingham, MA, USA, May 2011.
- [3] Arduino, Arduino uno (2014), <http://arduino.cc/en/Main/ArduinoBoardUno>
- [4] M. Margolis, *Arduino Cookbook*, O'Reilly Media, Inc., 2011.
- [5] P. LechEmail and P. Wlodarski, "Analysis of the IoT WiFi Mesh Network", In: *Cybernetics and Mathematics Applications in Intelligent Systems (CSOC 2017), Advances in Intelligent Systems and Computing*, vol 574, Springer, 2017.
- [6] X. R. Jiang, Y. M. Lv, X. H. Cheng, "Design of Wireless Communication System Based on nRF24L01", *Advanced Materials Research*, Vols. 945-949, pp. 1756-1759, 2014.
- [7] P. Mell and T. Grance, "The NIST Definition of Cloud Computing", *National Institute of Standards & Technology*, Gaithersburg, MD, United States, 2011.
- [8] Amazon Web Service (2017), <https://aws.amazon.com>
- [9] Amazon Web Services (2017), "AWS IoT", <https://aws.amazon.com/iot>
- [10] J. Clark, "Amazon switches on DynamoDB cloud database service", *ZDNet*, 2011 [11] "MQTT 3.1.1 specification", OASIS, December 10, 2015.
- [12] A. Stanford-Clark and H. L. Traug, "MQTT For Sensor Networks (MQTT-SN) Protocol Specification Version 1.2", mqtt.org, p. 27, November 14, 2013.

ประวัติผู้เขียน

ชื่อ-นามสกุล

นายภูมิพัฒน์ นิยมธรรม

ประวัติการศึกษา

ปีการศึกษา 2545 สำเร็จการศึกษาระดับปริญญาตรี
สาขาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์
มหาวิทยาลัยวงษ์ชวลิตกุล

ตำแหน่งและสถานที่ทำงานปัจจุบัน

กรรมการผู้จัดการและบริหารด้านการตลาด
อุตสาหกรรมยานยนต์ บริษัท ฟาส์คอม เทคโนโลยี จำกัด

