



รายงาน

โครงการระบบผึ้งตัว

เรื่อง Smart humidifier

โดย

กลุ่ม W.A.N.G

นายพสิษฐ์

ชาดาชวากุล

รหัสประจำตัวนิสิต 6733166921

นายภัทรนันท์

เจริญผล

รหัสประจำตัวนิสิต 6733194421

นายศิปปวิชญ์

แข็งชุมดาว

รหัสประจำตัวนิสิต 6733272721

นายศิริวิชญ์

เพ็ชรศรีอุดม

รหัสประจำตัวนิสิต 6733275621

โครงการเล่นนี้เป็นส่วนหนึ่งของรายวิชา 2110356 ระบบผึ้งตัว

ภาคเรียนที่ 1 ปีการศึกษา 2568

สาขาวิชาวิศวกรรมคอมพิวเตอร์และเทคโนโลยีดิจิทัล

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

คำนำ

รายงานฉบับนี้เป็นส่วนหนึ่งของวิชา 2110356 ระบบฝังตัว (Embedded Systems) จัดทำขึ้นเพื่อศึกษาการออกแบบและพัฒนาระบบควบคุมสำหรับ เครื่องทำความชื้นอัจฉริยะ(Smart humidifier) โดยมุ่งเน้นการประยุกต์ใช้ในโครงการ โนเทลเลอร์ เซนเซอร์ตรวจความชื้น และอุปกรณ์อิเล็กทรอนิกส์ที่เกี่ยวข้อง เพื่อสร้างต้นแบบอุปกรณ์ที่สามารถทำงานได้จริง มีความเสถียร และตอบสนองต่อสภาวะแวดล้อมอย่างเหมาะสม
ผู้จัดทำหวังว่ารายงานฉบับนี้จะเป็นประโยชน์ต่อผู้อ่านที่สนใจด้านระบบฝังตัว อุปกรณ์ IoT และการพัฒนาระบบควบคุม ตลอดจนช่วยเพิ่มพูนความรู้ความเข้าใจเกี่ยวกับการประยุกต์ใช้ Embedded System ในงานจริง

คณะกรรมการ

กลุ่ม W.A.N.G

8 ธันวาคม 2568

สารบัญ

หน้า

คำนำ	ก
สารบัญ	ข
บทนำ	1
เอกสารที่เกี่ยวข้อง	2
วิธีการออกแบบ	3
Overview diagram	3
Sensor node	3
Gateway	4
Cloud / Storage	5
Dashboard	6
ผลการทดสอบระบบ	7
สรุปผล	8
ตำแหน่งหน้าที่	8

บทนำ

ในปัจจุบันช่วงฤดูหนาว สภาพอากาศมักมีความชื้นสัมพัทธ์ลดลงอย่างมาก ส่งผลให้ผู้คนจำนวนมากเกิดอาการผิวแห้ง ริมฝีปากแห้ง รากกายเคืองระบบทางเดินหายใจ รวมถึงทำให้สิ่งของบางประเภท เช่น เครื่องดูดควัน เฟอร์นิเจอร์ไม้ หรืออุปกรณ์อิเล็กทรอนิกส์ ก่อความเสียหายได้ร้ายจากการขยาย-หดตัวของวัสดุในสภาพอากาศแห้ง ด้วยเหตุนี้ เครื่องทำความชื้น(Humidifier) จึงมีความสำคัญมากขึ้นในช่วงหน้าหนาว เพื่อช่วยปรับสภาพแวดล้อมให้อุ่นในระดับความชื้นที่เหมาะสมต่อการอยู่อาศัยและใช้งาน

การทำงานของเครื่องพ่นไอน้ำสมัยใหม่จำเป็นต้องอาศัยเทคโนโลยีของระบบฝังตัว (Embedded System) ไม่ว่าจะเป็นการควบคุมการสร้างละอองไอน้ำ การตรวจวัดความชื้นแบบเรียลไทม์ การตั้งค่าระดับความชื้นที่ต้องการ หรือการเพิ่มฟังก์ชันด้านความปลอดภัย เช่น การปิดเครื่องอัตโนมัติเมื่อระดับน้ำต่ำ การประยุกต์ใช้งานไมโครคอนโทรลเลอร์ เช่นเซอร์และวงจรอิเล็กทรอนิกส์อย่างเหมาะสมนี้จึงเป็นปัจจัยสำคัญที่ทำให้อุปกรณ์สามารถทำงานได้อย่างถูกต้องและมีประสิทธิภาพ

เอกสารที่เกี่ยวข้อง

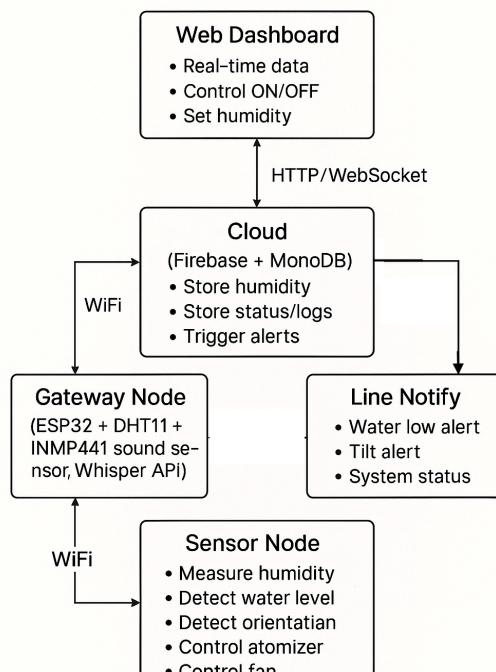
บทความของ REHVA (2021) เรื่อง “Relative humidity in the indoor air – impact on indoor air quality and means of control” ได้อธิบายถึงความสำคัญของความชื้นสัมพัทธ์ (Relative Humidity: RH) ภายในอาคาร โดยระบุว่าค่าความชื้นที่เหมาะสมอยู่ในช่วงประมาณ 40–60% ซึ่งเป็นระดับที่ช่วยลดการแพร่กระจายของเชื้อโรคในอากาศ ลดการระคายเคืองของระบบทางเดินหายใจ และเพิ่มความสบายในการอยู่อาศัย นอกจากนี้ยังชี้ให้เห็นว่าความชื้นต่ำในช่วงฤดูหนาวอาจส่งผลให้ผู้ที่อยู่ภายในห้องเกิดผิวแห้ง ไอแห้ง และอาจทำให้วัสดุภายในอาคารเกิดการเสื่อมสภาพเร็วขึ้น

งานวิจัยดังกล่าวถึงแนวทางการควบคุมความชื้นภายในอาคาร เช่น การใช้เครื่องเพิ่มความชื้น (Humidifier) หรือระบบตรวจความชื้นอัตโนมัติ เพื่อรักษาค่าความชื้นให้เสถียรตามช่วงที่กำหนด ซึ่งมีความเกี่ยวข้องโดยตรงกับโปรเจกต์ของผู้จัดทำที่พัฒนา “เครื่องพ่นไอน้ำควบคุมความชื้นในห้อง” โดยใช้เซนเซอร์ตรวจจับความชื้นและไมโครคอนโทรลเลอร์ในการเปิด–ปิดการพ่นไอน้ำอัตโนมัติตามสภาพแวดล้อมจริง การอ้างอิงงานนี้ช่วยสนับสนุนความสำคัญของการควบคุมความชื้นอย่างถูกต้อง รวมทั้งยืนยันว่าระบบที่พัฒนามีประสิทธิภาพจริงในด้านการปรับปรุงคุณภาพอากาศภายในอาคาร (Indoor Air Quality)

REHVA, “Relative humidity in the indoor air – impact on indoor air quality and means of control,” *REHVA Journal*, 2021

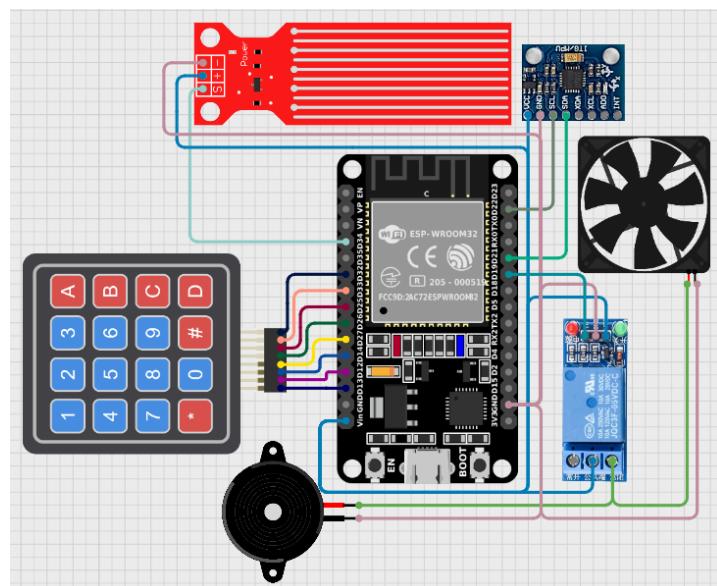
วิธีการออกแบบ

Overview diagram



Overview Diagram

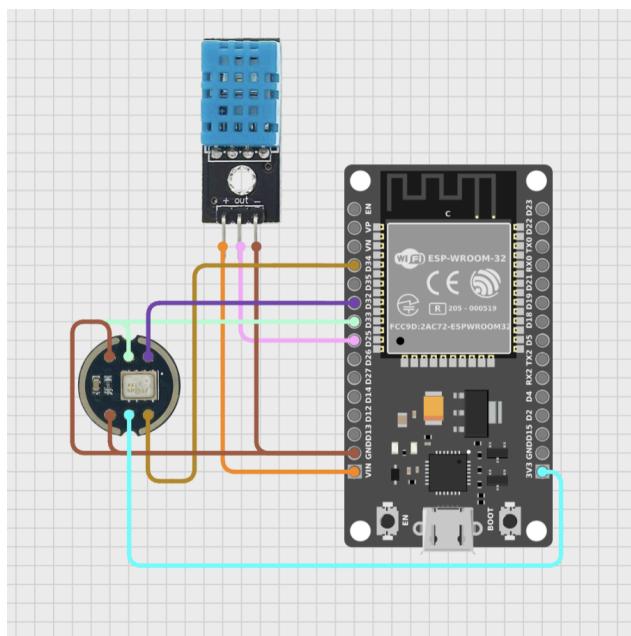
Sensor Node



- 1) Ultrasonic Atomizer อุปกรณ์สร้างไอน้ำความถี่สูง
- 2) 5V Fan ช่วยกระจายไอน้ำสู่ห้อง
- 3) Water Level Module ตรวจระดับน้ำ เพื่อป้องกันเครื่องทำงานต่อเมื่อไม่มีน้ำ
- 4) น้ำ Relay Module KY-019 ใช้ควบคุมการเปิด–ปิดพัดลมและเครื่องทำน้ำ
- 5) Gyroscope (MPU6050) ตรวจสอบการเอียงของเครื่อง เพื่อความปลอดภัย
- 6) Keypad สำหรับผู้ใช้ตั้งค่าความชื้นเป้าหมายหรือโหมดการทำงาน

หน้าที่ของ Sensor Node ได้แก่ วัดความชื้น/อุณหภูมิ ตรวจระดับน้ำเพื่อหยุดการทำงานเมื่อ “น้ำหมด” ตรวจการเอียงของเครื่องเพื่อความปลอดภัย ส่งสัญญาณควบคุมและข้อมูลให้ Gateway Node รับคำสั่งจาก Web และ Cloud เช่น เปิด/ปิดหรือปรับค่าความชื้นเป้าหมาย

Gateway node



- 1) DHT11 Sensor วัดความชื้นและอุณหภูมิอิจจุดเพื่อความแม่นยำ
- 2) INMP441 Sound Sensor ใช้รับคำสั่งเสียง (Speech Recognition)
- 3) ESP32 Gateway ทำหน้าที่สื่อสารกับ Sensor Node และเชื่อม Cloud

หน้าที่หลักของ Gateway Node ประมวลผลสัญญาณเสียง เช่น คำสั่ง “Open / Close” Eco Friendly เชื่อมต่อกับ Whisper API เพื่อทำ Speech to Text ส่งข้อมูลความชื้น/สถานะอุปกรณ์ขึ้น Firebase รับคำสั่งจาก Web และส่งไป Sensor Node

Cloud / Storage

```

https://embedded-curtain-project-default-rtdb.firebaseio.com/
  +-- control
    |   +-- control_state: false
    |   +-- manual_state: false
    |   +-- mode: "manual"
    |   +-- updated_at: 1765161090557
  +-- humidity
    +-- control
      +-- target_humidity: 65
      +-- updated_at: 1765160694047
  +-- schedule
    +-- countdown_sec: 0
    +-- enable: false
    +-- start_time: "02:40"
  
```

1) Firebase Real-time Database

ใช้สำหรับเก็บข้อมูลความชื้น/อุณหภูมิแบบเรียลไทม์ เก็บสถานะของอุปกรณ์เปิด-ปิด ใช้เป็น trigger สำหรับ Line Notification MongoDB ใช้เก็บข้อมูลประวัติ เช่น Log การทำงาน ค่า Humidity/Temp report การตั้งค่าโดยผู้ใช้ข้อมูลย้อนหลังเพื่อข้อเสนอแนะเพื่อวิเคราะห์ Firebase ใช้สำหรับ real-time control

2) MongoDB

ใช้เก็บข้อมูลประวัติ เช่น Log การทำงาน ค่า Humidity/Temp report การตั้งค่าโดยผู้ใช้ ข้อมูลย้อนหลังเพื่อวิเคราะห์

2.1) Control Log สำหรับเก็บข้อมูลที่เคยสั่งการ

Field	Type	Value
isOn	Boolean	false
mode	String	"manual"
scheduleEnabled	Boolean	false
scheduleObj	Object	{}

2.2) Sensor Snapshots สำหรับเก็บข้อมูล Sensor ที่อ่านได้

The screenshot shows the Compass interface with the 'Find' tab selected. On the left, there is a sidebar with a tree view of collections: 'curtain_project' (control_logs, sensor_snapshots), 'test' (speech_logs), and 'test'. The 'sensor_snapshots' collection is currently selected. In the main panel, there is a search bar with the placeholder 'Generate queries from natural language in Compass' and a 'Filter' input field containing the query '{ field: 'value' }'. Below the filter, the document structure is displayed:

```
env : Object
  humidity : 53.1
  temperature : 26.9
tilt : Object
  state : 2
  state_text : "FALL"
water : Object
  percent : 58
  raw : 1838
key : Object
  createdAt : 2025-12-07T17:42:43.178+00:00
```

2.3) Speech Log สำหรับเก็บข้อมูล Speech ที่เคยพูด

The screenshot shows the Compass interface with the 'Find' tab selected. On the left, there is a sidebar with a tree view of collections: 'curtain_project' (control_logs, sensor_snapshots), 'test' (speech_logs), and 'test'. The 'speech_logs' collection is currently selected. In the main panel, there is a search bar with the placeholder 'Generate queries from natural language in Compass' and a 'Filter' input field containing the query '{ field: 'value' }'. Below the filter, the document structure is displayed:

```
uniqueId : "0ac289f6-5ab3-49b8-8ac2-454dd3091fe5"
text : "Hello."
createdAt : 2025-12-07T18:47:22.474+00:00
```



```
_id: ObjectId('6935cbbe86f050c37680b04c')
uniqueId : "39567b65-007a-4959-99fd-ff1c1772af5e"
text : "Peter, you need my help, sir."
createdAt : 2025-12-07T18:47:26.565+00:00
```

Dashboard

The dashboard consists of three main sections:

- Top Section (Sensor Control):** Features "AUTO AUTOMATION" with a target humidity of 50% and an "ENABLE AUTO MODE" button. Below it is a "SCHEDULER" section with start and stop times set to 10:38 and 10:39 respectively, and a "Save Schedule" button.
- Middle Section (System Status):** Labeled "COMMAND CENTER" with "SEC-CODE-A13 // SYSTEM READY". It includes a "SYSTEM POWER" button labeled "STANDBY", "AUTO AUTOMATION" settings, and a message: "We'll be back in the next video."
- Bottom Section (Sensor Feed):** Labeled "SENSOR FEED". It displays real-time sensor data:
 - Temperature: 25.9 °C
 - Humidity: 40.3 %
 - Water Level: 100 %
 - Tilt State: NORMAL

Dashboard แสดงความชื้น, อุณหภูมิ, ระดับน้ำแบบ real-time

ตั้งเวลาเปิด–ปิดเครื่อง ตั้งค่าความชื้นเป้าหมาย ดูสถานะเข็น เครื่องล้ม นำกำลังจะหมด เชื่อมต่อผ่าน WiFi API

ผลการทดสอบระบบ

การทดลอง	การผลลัพธ์
เว็บ	
- การแสดงสถานะ	สามารถดึงข้อมูลจาก cloud มาแสดงผลได้
- เปิด/ปิด เครื่องทำความชื้น	สามารถแสดงผลและส่งคำสั่งไป cloud ได้
- การตั้งโหมดอัตโนมัติ	สามารถเปิด/ปิดโหมด, ตั้งความชื้น, และส่งคำสั่งไป cloud ได้
- การตั้งเวลา	สามารถเปิด/ปิดโหมด, ตั้งเวลา, และส่งคำสั่งไป cloud ได้
Cloud	
- Firebase	สามารถรับ ส่ง และเก็บค่าตามที่กำหนดได้
- Postgres	สามารถรับและเก็บข้อมูล logs ได้
Gateway node	สามารถทำงานที่ตามเป้าหมายได้
- Communication	สามารถรับและส่งค่าไปยัง cloud ได้ สามารถสื่อสารกับ sensor node ได้ สามารถเรียกใช้ api ได้
- Sensors	สามารถอ่านค่าจากเซ็นเซอร์ได้
Sensor node	สามารถทำงานที่ตามเป้าหมายได้
- Communication	สามารถสื่อสารกับ gateway node ได้
- Sensors	สามารถสื่อสารกับเซ็นเซอร์ได้
- Actuators	สามารถควบคุมพัดลมกับเครื่องพ่นไอน้ำผ่าน relay ได้
Speech recognition	สามารถจับคำสั่งได้บางครั้ง ความแม่นยำเมื่อมีเสียงรบกวน ต่ำ
Line notification	สามารถอ่านค่าจาก firebase ได้ สามารถแจ้งเตือนสถานะใน line ได้

สรุปผลและข้อเสนอแนะ

ระบบ Smart Humidifier ที่พัฒนาสามารถทำงานได้ครบตามเป้าหมาย

- 1) ตรวจวัดความชื้นและควบคุมเครื่องเพิ่มความชื้นอัตโนมัติ
- 2) แสดงข้อมูลแบบเรียลไทม์บน Dashboard
- 3) แจ้งเตือนเหตุการณ์ผิดปกติ เช่น เครื่องเสีย น้ำหมด
- 4) รองรับคำสั่งเสียงเพื่อความสะดวก
- 5) มีระบบความปลอดภัยเหมาะสม

โครงการนี้สามารถต่อยอดได้โดยการทำให้ speech recognition เสียงร้องเพิ่มโหมด/ฟังก์ชันอื่นๆ และทำการทดลองที่หลากหลายมากยิ่งขึ้น

ตำแหน่งหน้าที่

Line OA notification

นายพสิษฐ์

ราดาชวสกุล

รหัสประจำตัวนิสิต 6733166921

Embedded & Cloud programing

นายภัทรนันท์

เจริญผล

รหัสประจำตัวนิสิต 6733194421

Frontend ux/ui

นายสิบปีรุจ្យ

แซ่�ชุดดาว

รหัสประจำตัวนิสิต 6733272721

Circuit design

นายศิริวิชญ์

เพ็ชรศรีอุดม

รหัสประจำตัวนิสิต 6733275621