

คะแนน (เต็ม 10)

10/10

โศภณ ตายะ
(ลายมือชื่อ
อ.ที่ปรึกษา)

แบบรายงานความก้าวหน้าในการทำโครงการ

โครงการ การพัฒนากล่องควบคุมเครื่องปรับอากาศแบบแยกส่วนในรูปแบบ IoT

รายงานครั้งที่ 3

1 ภาพรวม (Summary)

งานด้านเทคนิค

ศึกษาองค์ประกอบคอยล์เย็นจากอุปกรณ์จริง เขียนเว็บแอปพลิเคชันหน้า login , Register , Forgot Password , Home , Temperature Status

งาน (Task)	นักศึกษา 1 กัลยกร	นักศึกษา 2 ชัยภัทร	นักศึกษา 3 ธิภาพ	ร้อยละ ความสำเร็จ ตามแผนงาน
งานด้านเทคนิค 1 ศึกษาองค์ประกอบจากคอยล์เย็นจากอุปกรณ์จริง	2 ชั่วโมง	2 ชั่วโมง	1 ชั่วโมง	100
งานด้านเทคนิค 2 เขียนเว็บแอปพลิเคชันหน้าล็อกอินเข้าใช้งาน	2 ชั่วโมง	2 ชั่วโมง	3 ชั่วโมง	100
งานด้านเทคนิค 3 เขียนเว็บแอปพลิเคชันหน้า Register	1 ชั่วโมง	1 ชั่วโมง	3 ชั่วโมง	100
งานด้านเทคนิค 4 เขียนเว็บแอปพลิเคชันหน้า Forgot Password	1 ชั่วโมง	1 ชั่วโมง	3 ชั่วโมง	100
งานด้านเทคนิค 5 เขียนเว็บแอปพลิเคชันหน้า Home	1 ชั่วโมง	2 ชั่วโมง	3 ชั่วโมง	100
งานด้านเทคนิค 6 เขียนเว็บแอปพลิเคชันหน้า Temperature Status	1 ชั่วโมง	1 ชั่วโมง	3 ชั่วโมง	100

งานทั่วไป

ศึกษาค้นคว้าข้อมูลองค์ประกอบและหลักการทำงานของคอยล์เย็น ชนิดของเซ็นเซอร์วัดอุณหภูมิที่ใช้ในระบบเครื่องปรับอากาศและหลักการทำงานของเซ็นเซอร์

งาน (Task)	นักศึกษา 1 กัลยกร	นักศึกษา 2 ชัยภัทร	นักศึกษา 3 ธิภาพ	ร้อยละ ความสำเร็จ ตามแผนงาน
งานทั่วไป 1 ค้นคว้าข้อมูลเรื่อง องค์ประกอบและหลักการทำงานของคอยล์เย็น	3 ชั่วโมง	3 ชั่วโมง	1 ชั่วโมง	100
งานทั่วไป 2 ค้นคว้าข้อมูลเรื่อง ชนิดของเซ็นเซอร์วัดอุณหภูมิที่ใช้ในระบบเครื่องปรับอากาศและหลักการทำงานของเซ็นเซอร์	3 ชั่วโมง	2 ชั่วโมง	1 ชั่วโมง	100

2. ผลสัมฤทธิ์สำคัญในสัปดาห์ที่ผ่านมา (Highlights)

2.1 ทราบรายละเอียดจากการค้นคว้าข้อมูลเรื่อง องค์ประกอบและหลักการทำงานของคอยล์เย็น

คอยล์เย็น เป็นชุดของแอร์ที่ถูกติดตั้งภายในห้องหรืออาคารประกอบด้วยแผงคอยล์เย็นที่ติดกับส่วนท่อน้ำยาแอร์ที่บรรจุน้ำยาแอร์ไหลเวียนภายในส่งต่อไปยังคอมเพรสเซอร์โดยขณะไหลเวียนผ่านท่อจะมีพัดลมคอยดูดอากาศภายในห้องจากด้านล่างเครื่องผ่านท่อ และแผงคอยล์เย็นเพื่อให้เกิดการแลกเปลี่ยนความร้อน และผ่านอากาศเย็นออกมายังช่องแอร์ในด้านปลาย มีส่วนประกอบ ดังนี้

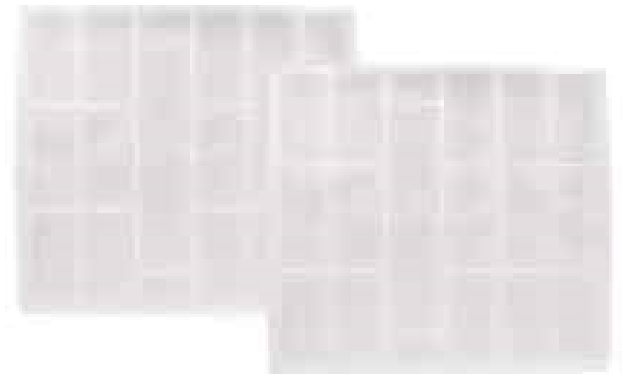
2.1.1 หน้ากากคอยล์เย็น



รูปที่ 1 หน้ากากคอยล์เย็น

หน้ากากคอยล์เย็น มีลักษณะเป็นพลาสติก รูปแบบของหน้ากากจะแตกต่างกันไป ตามรุ่นตามยี่ห้อของบริษัทผู้ผลิต ซึ่งหน้ากากคอยล์เย็นนี้ก็จะมิตัวบานเปิดปิด บางรุ่นก็จะมี Swing แบบซ้ายขวาบางรุ่นก็จะปรับทิศทางลมได้แค่ขึ้นลงเท่านั้น

2.1.2 แผ่นกรองอากาศ / ฟิลเตอร์ (Filter)



รูปที่ 2 แผ่นกรองฟิวเตอร์

ฟิวเตอร์แอร์ หรือแผ่นกรองอากาศ เป็นชิ้นส่วนสำคัญที่อยู่ในชุดคอยล์เย็นของเครื่องปรับอากาศ มีลักษณะเป็นแผ่นตะแกรงพลาสติก ติดอยู่บริเวณใต้ฝาด้านของแอร์ พอเปิดฝาด้านออกก็จะเจอกับแผ่นฟิวเตอร์แอร์ทันที เมื่อแอร์สูบอากาศภายนอกเข้าไปในเครื่อง ฟิวเตอร์จะทำหน้าที่ดักจับฝุ่นละอองในอากาศไม่ให้เข้าไปถึงคอยล์เย็นได้ ทำให้แอร์ทำงานได้เป็นปกติ และอากาศที่แอร์เป่าออกมา ก็จะสะอาดปราศจากฝุ่น

ฟิวเตอร์หน้าแอร์จะมี 2 แบบ ได้แก่

1. ฟิวเตอร์แบบหยาบ : ใช้ดักจับฝุ่นขนาดใหญ่และขนาดกลาง เราสามารถถอดออกมล้างเองได้

2. ฟیلเตอร์แบบยิงประจุ : สามารถกำจัดฝุ่นละอองขนาดเล็กมาก
เชื้อแบคทีเรีย หรือควันในอากาศได้ เหมาะสำหรับคนที่ใส่ใจ
สุขภาพ แต่ไม่สามารถถอดล้างเองได้

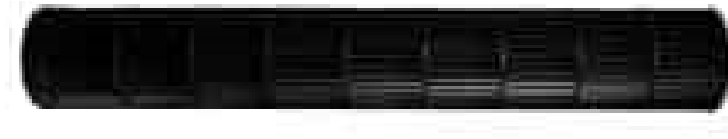
2.1.3 แผงขดท่อคอยล์เย็น (Cooling Coil)



รูปที่ 3 แผงขดท่อคอยล์เย็น (Cooling Coil)

แผงขดท่อคอยล์เย็นคือ ตัวสร้างความเย็น มีรูปร่างเป็นเส้นท่อขดไปตาม
ความยาวของเครื่อง และจะมีแผ่นครีบอลูมิเนียมบาง ๆ หุ้มขดท่อเหล่านั้นอยู่ แผงขด
ท่อจะมองเห็นได้อย่างชัดเจนเมื่อถอดหน้ากากส่งลมหรือหน้ากากรับลมกลับ ของเครื่อง
ออก ที่แผงขดท่อนี้จะมีฝุ่นผงขนาดเล็กที่สามารถผ่านการกรองของฟیلเตอร์เข้ามาได้
เมื่อใช้งานไปนาน ๆ ฝุ่นเหล่านี้จะจับตัวกันหนาขึ้น และอากาศจะไม่สามารถผ่านได้ ซึ่ง
จะทำให้เครื่องปรับอากาศมีผลเช่นเดียวกับฟیلเตอร์ตัน จึงควรมีการล้างทำความสะอาด
สะอาดขดท่อและแผ่นอลูมิเนียมโดยในระยะเวลาในการล้างในรอบหนึ่งปี ควรมีการล้าง
1 ครั้ง

2.1.4 ใบพัดลมคอยล์เย็น (Blower)



รูปที่ 4 ใบพัดลมคอยล์เย็น (Blower)

ใบพัดลมคอยล์เย็นหรือโบลเวอร์ เป็นตัวขับเคลื่อนให้เกิดการเคลื่อนที่ของลม โดยได้กำลังมาจากมอเตอร์ไฟฟ้า ฝุ่นผงขนาดเล็กที่เล็ดลอดมาจากการดักจับของแผงกรองอากาศบางส่วน จะมาจับอยู่ที่ใบพัดลมทำให้ร่องตักลมของใบพัดลมอุดตันไม่สามารถตักลมได้ เติมนที่ การเกิดในลักษณะเช่นนี้จะทำให้ปริมาณลมเย็น ที่ออกไปจากคอยล์เย็นลดลง จึงต้องเสียเวลาในการเดินเครื่องปรับอากาศนานขึ้น เพื่อที่จะให้ได้อุณหภูมิของห้องเท่าเดิม ซึ่งมีผลทำให้เสียค่าไฟฟ้าเพิ่มขึ้น นอกจากฝุ่นที่เกาะตามใบพัดลมจะทำให้พัดลมส่งลมเย็นออกมาได้น้อยแล้ว อาจจะทำให้เกิด เสียงดังที่ตัวชุดคอยล์เย็นขึ้นได้ เนื่องจากฝุ่นที่จับอยู่จะไปเพิ่มน้ำหนักให้กับใบพัด ทำให้ใบพัดเสียการสมดุลในตัวเอง และเมื่อมอเตอร์หมุนจะเกิดการสั่นสะเทือนจากแรงเหวี่ยงและเกิดเสียงดังขึ้นได้ การล้างทำความสะอาดใบพัด ควรล้างไปพร้อมกับการทำความสะอาดแผงคอยล์เย็น

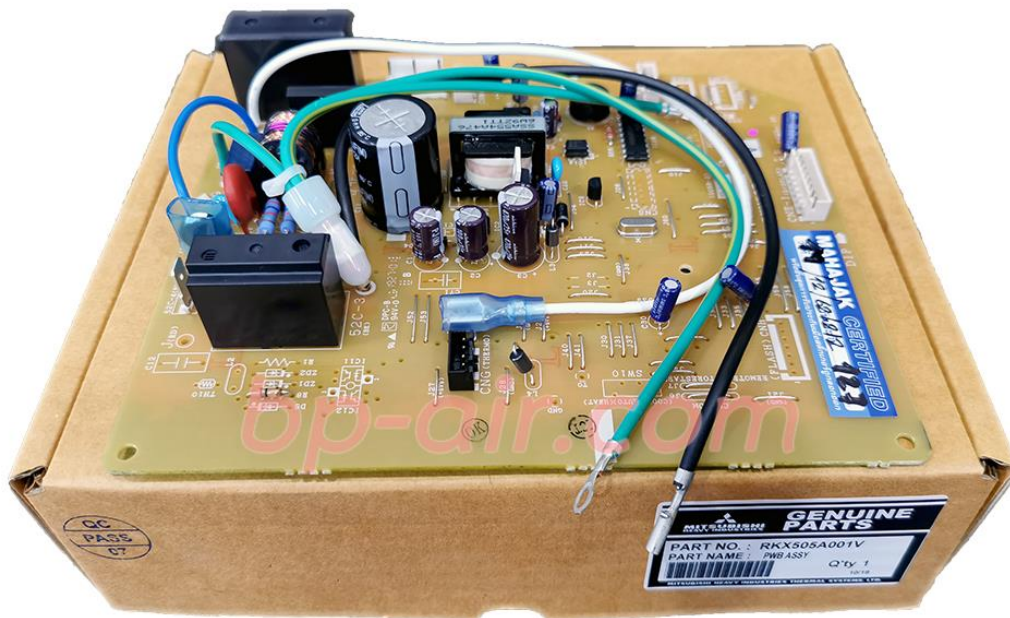
2.1.5 ถาดรองรับน้ำทิ้งและท่อน้ำทิ้ง (Condensate Tray & Drain Line)



รูปที่ 5 ถาดรองรับน้ำทิ้งและท่อน้ำทิ้ง (Condensate Tray & Drain Line)

ถาดรองรับน้ำทิ้งและท่อน้ำทิ้งเป็นอุปกรณ์สำหรับรองรับน้ำที่เกิดจากการกลั่นตัวเป็นหยดน้ำของไอน้ำในอากาศภายในห้อง น้ำที่เกิดขึ้นนี้จะไหลไปรวมกันที่ถาดรองรับน้ำและถูกระบายน้ำทิ้งโดยผ่านทางท่อน้ำทิ้ง ที่ถาดรองรับน้ำทิ้งนี้ถ้าไม่ได้รับการดูแลหรือทำความสะอาดเป็นเวลานาน จะทำให้เกิดเมือกขาวใสคล้ายวุ้น น้ำที่ขังอยู่ในถาดรองรับน้ำทิ้งเป็นเวลานานนี้เมื่อรวมกับฝุ่นละอองต่างๆ ที่เกาะอยู่ตามถาดรับก็อาจจะเป็นแหล่งอาหารหรือเป็นแหล่งสะสมของเชื้อโรค เชื้อรา และทำให้เชื้อโรค เชื้อราเหล่านี้ เจริญเติบโตและแพร่กระจายสู่ผู้ปฏิบัติงาน

2.1.6 แผงวงจรคอยล์เย็น แผงบอร์ดแอร์เป็นชิ้นส่วนอิเล็กทรอนิกส์สำหรับคอยล์เย็น มีหน้าที่คอยสั่งงานให้เครื่องปรับอากาศทำงาน



รูปที่ 6 แผงวงจรคอยล์เย็น

2.1.7 มอเตอร์คอล์ยเย็น



รูปที่ 7 มอเตอร์คอล์ยเย็น

จะใช้คู่กับใบพัดลมโพรงกระรอกมอเตอร์คอล์ยเย็นจะติดประกอบกับตัวใบพัดลมโพรงกระรอก เมื่อมอเตอร์สตาร์ทใบพัดลมโพรงกระรอกก็จะหมุนตามมอเตอร์ ซึ่งขนาดมอเตอร์ภายในตัวเครื่องจะขึ้นอยู่กับขนาดของตัวเครื่องปรับอากาศด้วย ค่าต่างๆ จะไม่เท่ากันและสายแฉีกจะไม่สามารถใช้ร่วมกันได้

2.2 ทราบรายละเอียดจากการค้นคว้าข้อมูลเรื่อง ชนิดของเซ็นเซอร์วัดอุณหภูมิที่ใช้ในระบบเครื่องปรับอากาศและหลักการทำงานของเซ็นเซอร์

เซ็นเซอร์วัดอุณหภูมิที่ใช้ในระบบเครื่องปรับอากาศ (แอร์) มีชนิดต่างๆ มีหน้าที่เฉพาะและติดตั้งในตำแหน่งต่างๆ ภายในระบบแอร์ เพื่อทำงานร่วมกันในการควบคุมอุณหภูมิของห้องและรักษาประสิทธิภาพในการทำความเย็น รายละเอียดของแต่ละเซ็นเซอร์มีดังนี้

2.2.1 เทอร์มิสเตอร์ (Thermistor) ใช้ในการวัดอุณหภูมิห้อง คอล์ยเย็น และคอล์ยร้อน เป็นเซ็นเซอร์ที่ใช้มากที่สุดในระบบแอร์ เพราะราคาถูก ความแม่นยำเพียงพอ และทนทาน



รูปที่ 8 เทอร์มิสเตอร์ (Thermistor)

ชนิด : NTC (Negative Temperature Coefficient)

ทำหน้าที่ : วัดอุณหภูมิของอากาศหรือชิ้นส่วนต่างๆ ภายในแอร์ เช่น อุณหภูมิห้อง อุณหภูมิของคอยล์เย็น และอุณหภูมิของคอยล์ร้อน เพื่อนำไปใช้ควบคุมการทำงานของคอมเพรสเซอร์ พัดลม และอื่นๆ

จุดที่ติดตั้ง :

- อุณหภูมิห้อง (Room Temperature Sensor): ติดตั้งในตัวเครื่องแอร์เพื่อวัดอุณหภูมิของห้องโดยรวม
- คอยล์เย็น (Evaporator Coil Sensor): ติดตั้งใกล้กับคอยล์เย็นเพื่อวัดอุณหภูมิของคอยล์และป้องกันไม่ให้น้ำแข็งเกาะคอยล์
- คอยล์ร้อน (Condenser Temperature Sensor): ติดตั้งใกล้กับคอยล์ร้อนเพื่อวัดอุณหภูมิของสารทำความเย็นขณะผ่านคอยล์ร้อน

หลักการทำงาน : เทอร์มิสเตอร์แบบ NTC มีค่าความต้านทานที่ลดลงเมื่ออุณหภูมิเพิ่มขึ้น โดยเมื่ออุณหภูมิเปลี่ยนไป ค่าความต้านทานของเทอร์มิสเตอร์จะ

เปลี่ยนแปลงตาม ซึ่งระบบควบคุม (Controller) จะอ่านค่าความต้านทานนี้แล้วแปลงเป็นค่าอุณหภูมิ จากนั้นใช้ในการควบคุมการทำงานของระบบ

2.2.2 เทอร์โมสตัท (Thermostat) ใช้ควบคุมการทำงานของคอมเพรสเซอร์ ให้เปิด-ปิดตามอุณหภูมิที่ต้องการ



รูปที่ 9 เทอร์โมสตัท (Thermostat)

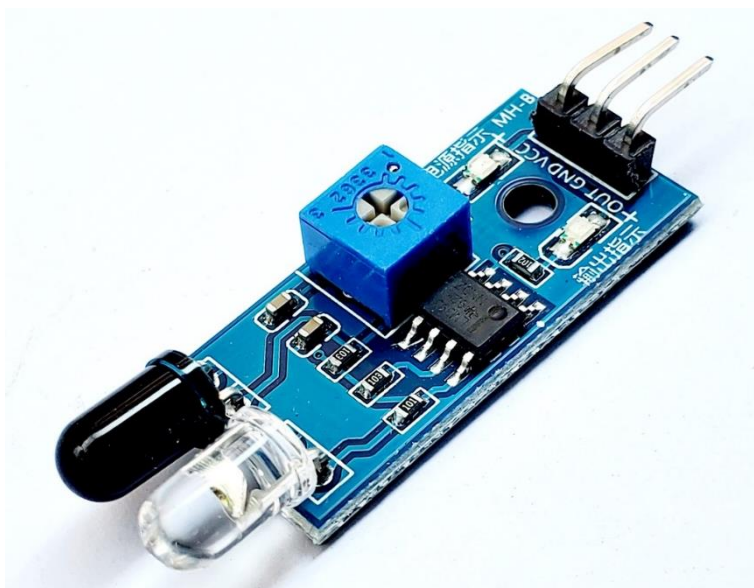
ชนิด : Bimetallic Thermostat หรือ Digital Thermostat

ทำหน้าที่ : ตรวจวัดอุณหภูมิของห้องและควบคุมการทำงานของคอมเพรสเซอร์ให้เปิดหรือปิดตามอุณหภูมิที่ตั้งไว้ ป้องกันไม่ให้อุณหภูมิสูงหรือต่ำเกินไป เพื่อรักษาระดับอุณหภูมิห้องให้เหมาะสม

จุดที่ติดตั้ง : ติดตั้งในตำแหน่งที่สามารถตรวจวัดอุณหภูมิห้องได้อย่างแม่นยำ มักอยู่ในตัวเครื่องแอร์ที่มีการไหลเวียนของอากาศผ่าน

หลักการทำงาน : เทอร์โมสตัทใช้กลไกในการตรวจจับการเปลี่ยนแปลงของอุณหภูมิ เมื่ออุณหภูมิห้องสูงเกินกว่าที่ตั้งไว้ เทอร์โมสตัทจะส่งสัญญาณให้คอมเพรสเซอร์ทำงาน และเมื่ออุณหภูมิถึงระดับที่ตั้งไว้ จะหยุดการทำงานของคอมเพรสเซอร์เพื่อประหยัดพลังงาน

2.2.3 เซ็นเซอร์อินฟราเรด (Infrared Temperature Sensor) ใช้ในกรณีที่ต้องวัดอุณหภูมิในตำแหน่งที่เข้าถึงยาก โดยไม่ต้องสัมผัส



รูปที่ 10 เซ็นเซอร์อินฟราเรด (Infrared Temperature Sensor)

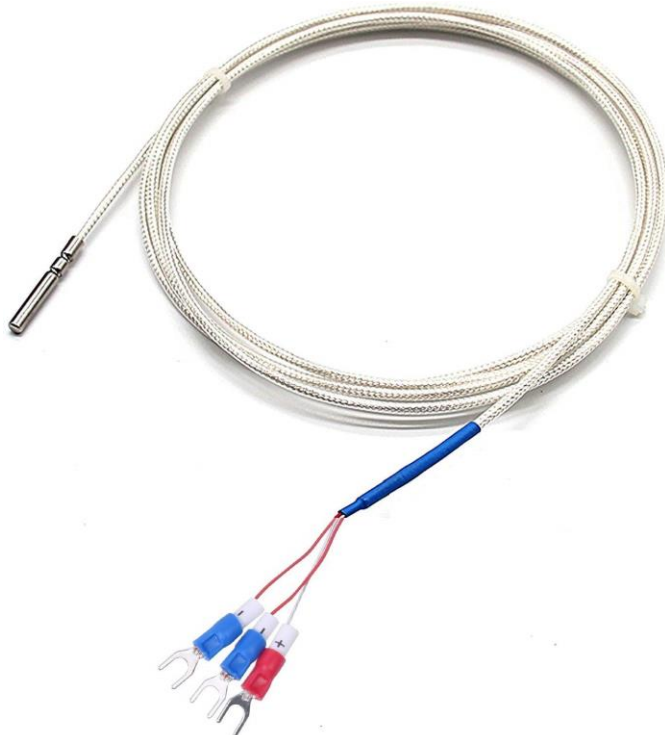
ชนิด : อิเล็กทรอนิกส์

ทำหน้าที่ : ตรวจวัดอุณหภูมิของวัตถุหรือพื้นที่โดยไม่ต้องสัมผัส ใช้ในบางกรณีเพื่อวัดอุณหภูมิของท่อหรือคอยล์ในระบบแอร์ที่ยากต่อการเข้าถึง

จุดที่ติดตั้ง : ติดตั้งในตำแหน่งที่สามารถตรวจจับการแผ่รังสีอินฟราเรดจากคอยล์หรือท่อได้ชัดเจน โดยไม่ต้องสัมผัสโดยตรง

หลักการทำงาน : เซ็นเซอร์อินฟราเรดทำงานโดยการตรวจจับรังสีอินฟราเรดที่ปล่อยออกมาจากวัตถุ แล้วแปลงสัญญาณเป็นข้อมูลอุณหภูมิ เซ็นเซอร์นี้สามารถวัดอุณหภูมิได้โดยไม่ต้องสัมผัสวัตถุ ซึ่งเหมาะกับการวัดอุณหภูมิในส่วนที่เข้าถึงยากหรือต้องการความปลอดภัย

2.2.4 เซ็นเซอร์เพื่อตรวจวัดอุณหภูมิท่อน้ำยา (Pipe Temperature Sensor) ใช้วัดอุณหภูมิของท่อน้ำยาทั้งไหลเข้าและไหลออก เพื่อตรวจสอบการทำงานของระบบ



รูปที่ 11 เซ็นเซอร์เพื่อตรวจวัดอุณหภูมิท่อน้ำยา (Pipe Temperature Sensor)

ชนิด : เทอร์มิสเตอร์หรือ RTD

ทำหน้าที่ : วัดอุณหภูมิของท่อน้ำยาทั้งในท่อไหลเข้าและท่อไหลออกจากคอยล์
เพื่อใช้ในการประเมินประสิทธิภาพการทำงานของคอมเพรสเซอร์และการเปลี่ยนแปลง
ของสารทำความเย็น

จุดที่ติดตั้ง : ติดตั้งตรงท่อที่มีการไหลของน้ำยา เช่น ท่อที่เชื่อมต่อระหว่าง
คอมเพรสเซอร์และคอยล์เย็นหรือคอยล์ร้อน

หลักการทำงาน : เมื่ออุณหภูมิของท่อเปลี่ยนแปลง ความต้านทานของ
เซ็นเซอร์จะเปลี่ยนไปตามอุณหภูมิ เซ็นเซอร์จะแปลงการเปลี่ยนแปลงนี้เป็น
สัญญาณไฟฟ้าเพื่อให้ระบบควบคุมสามารถทำการประเมินและปรับปรุงการทำงานของ
คอมเพรสเซอร์และพัดลมให้เหมาะสม

2.2.5 เซ็นเซอร์น้ำแข็ง (Ice Sensor) ป้องกันการเกิดน้ำแข็งเกาะคอยล์เย็น



รูปที่ 12 เซ็นเซอร์น้ำแข็ง (Ice Sensor)

ชนิด : NTC Thermistor

ทำหน้าที่ : ตรวจจับการเกิดน้ำแข็งเกาะบนคอยล์เย็นและป้องกันการ ทำงานผิดปกติของระบบเมื่อเกิดน้ำแข็ง

จุดที่ติดตั้ง : ติดตั้งที่คอยล์เย็น (Evaporator Coil) เพื่อสามารถตรวจจับการเกิดน้ำแข็งที่อาจส่งผลกระทบต่อประสิทธิภาพการทำงาน

หลักการทำงาน : เมื่อคอยล์เย็นมีอุณหภูมิต่ำเกินไปจนถึงจุดที่เสี่ยงเกิดน้ำแข็ง ค่าความต้านทานของ NTC Thermistor จะเปลี่ยนไป เซ็นเซอร์จะส่งสัญญาณไปยังระบบควบคุม เพื่อสั่งให้หยุดการทำงานของคอมเพรสเซอร์ชั่วคราวหรือเพิ่มการหมุนเวียนอากาศเพื่อป้องกันการเกิดน้ำแข็ง

2.3 ทราบรายละเอียดองค์ประกอบคอยล์เย็นจากอุปกรณ์จริง

จากการศึกษาองค์ประกอบคอยล์เย็นจากอุปกรณ์จริงได้พบว่ามีอุปกรณ์ ดังนี้



รูปที่ 13 องค์ประกอบคอยล์เย็น

จากรูปที่ 13 มีองค์ประกอบ คือ

1. หน้ากากคอยล์เย็น
2. แผ่นฟیلเตอร์กรองฝุ่น
3. แผงคอยล์เย็น
4. บานสวิงแอร์



รูปที่ 14 องค์ประกอบคอยล์เย็น

จากรูปที่ 14 มอเตอร์ประกอบ คือ

1. ใบพัดลมคอล์ยเย็น
2. มอเตอร์คอล์ยเย็น



รูปที่ 15 มอเตอร์บานสวิง

จากรูปที่ 15 มอเตอร์บานสวิงซึ่งเป็นมอเตอร์แบบสเต็ปมอเตอร์

หลักการทำงานหลักการทำงานของสเต็ปมอเตอร์

- โครงสร้างพื้นฐาน : สเต็ปมอเตอร์ประกอบด้วยสองส่วนหลัก คือ โรเตอร์ (Rotor) และสเตเตอร์ (Stator) โรเตอร์จะเป็นแม่เหล็กถาวรหรือแม่เหล็กแบบมีขั้วฟัน (เฟอไรต์) และสเตเตอร์มีขดลวดที่ถูกจัดเรียงในลักษณะเป็นเฟส

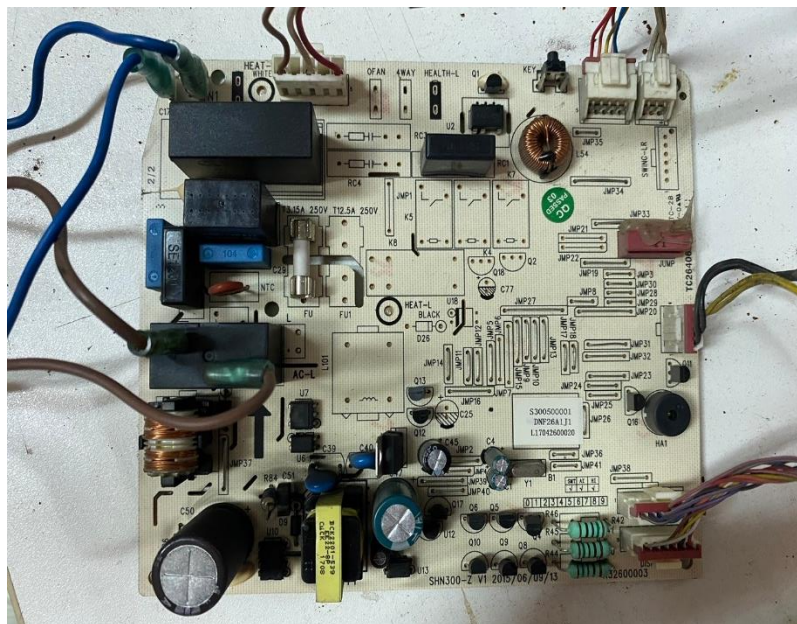
- การจ่ายไฟและการสร้างสนามแม่เหล็ก : เมื่อมีการจ่ายไฟฟ้าเข้าไปในขดลวดของสเตเตอร์ จะเกิดสนามแม่เหล็กที่ดึงดูดหรือผลักดันโรเตอร์ให้หมุนไปในทิศทางที่กำหนด สเต็ปมอเตอร์จะหมุนทีละ "ขั้น" ซึ่งจำนวนขั้นต่อการหมุนหนึ่งรอบจะขึ้นอยู่กับจำนวนขดลวดและวิธีการจัดเรียงของแม่เหล็กและขดลวด

- การทำงานเป็นขั้น : การหมุนของสเต็ปมอเตอร์เกิดจากการสลับการจ่ายไฟฟ้าไปยังขดลวดต่าง ๆ ในลำดับที่กำหนด ซึ่งเรียกว่า "ขั้น" หรือ "สเต็ป" โรเตอร์จะเคลื่อนไปยังตำแหน่งถัดไปเมื่อมีการสลับการจ่ายไฟ ทำให้สามารถควบคุมมุมการหมุนได้อย่างละเอียดและแม่นยำ

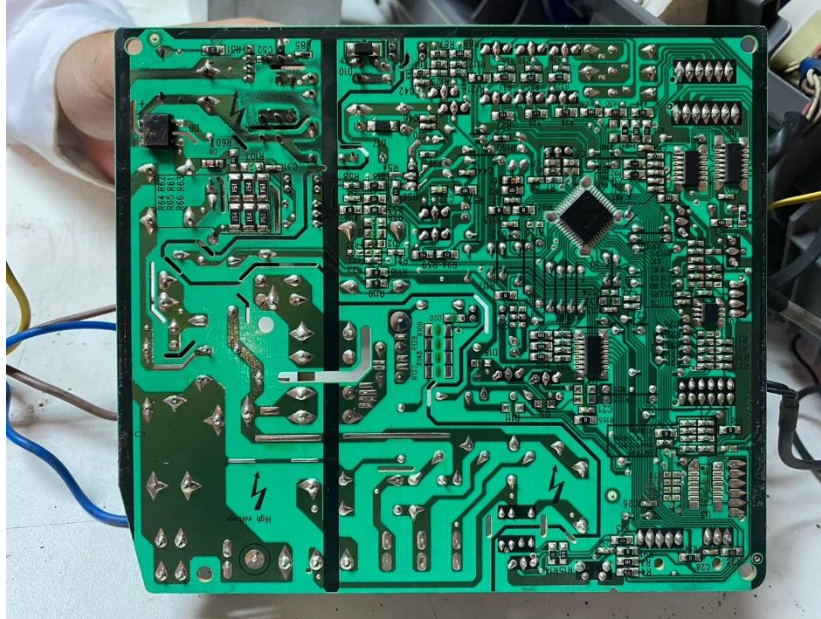
- ประเภทของสเต็ปมอเตอร์ : Unipolar Stepper Motor: มีขดลวดหลายชุดที่มีขั้วกลางที่ใช้ร่วมกัน ทำให้การควบคุมทำได้ง่าย Bipolar Stepper Motor: ใช้ขดลวดสองชุด โดยไม่มีขั้วกลาง การควบคุมจะซับซ้อนขึ้นเล็กน้อยแต่มีแรงบิดที่สูงกว่า Hybrid Stepper Motor: ผสมผสานคุณสมบัติของแม่เหล็กถาวรและแบบมีขั้วฟัน เพื่อเพิ่มความแม่นยำและประสิทธิภาพในการทำงาน

- การควบคุมการหมุน : สเต็ปมอเตอร์สามารถควบคุมการหมุนได้โดยใช้ไดรเวอร์ (Driver) ที่ควบคุมการจ่ายไฟฟ้าไปยังขดลวดในลำดับที่ถูกต้อง ไดรเวอร์สามารถรับคำสั่งจากไมโครคอนโทรลเลอร์เพื่อควบคุมทิศทางและจำนวนสเต็ปที่ต้องการ สเต็ปมอเตอร์สามารถหมุนเป็นมุมที่แน่นอนได้ เช่น หมุน 1.8 องศาต่อสเต็ป (สำหรับสเต็ปมอเตอร์ที่มี 200 สเต็ปต่อรอบ)

- การใช้ไมโครสเต็ป (Microstepping) : ไมโครสเต็ปเป็นเทคนิคที่ใช้เพื่อเพิ่มความละเอียดในการหมุน โดยแบ่งการหมุนแต่ละสเต็ปออกเป็นหลายขั้นย่อย ทำให้สเต็ปมอเตอร์สามารถหมุนได้อย่างนุ่มนวลและแม่นยำยิ่งขึ้น



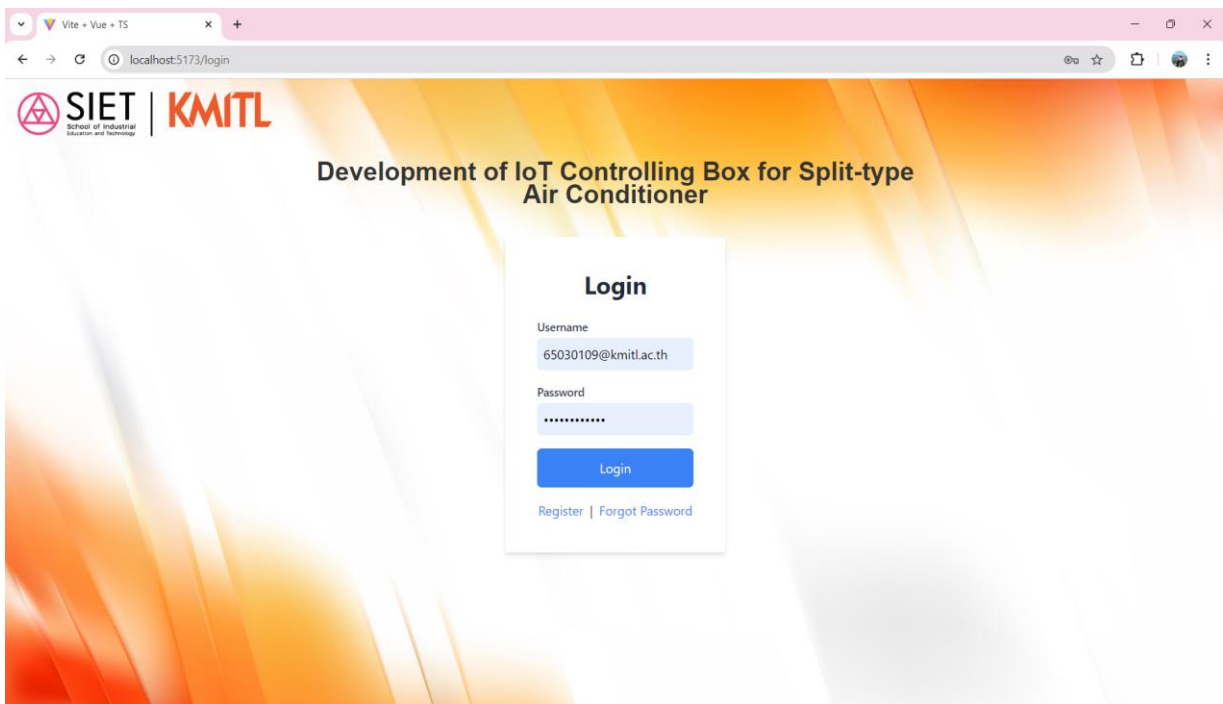
รูปที่ 16 ด้านหน้าแผงควบคุมคอยล์เย็น



รูปที่ 17 ด้านหลังแผงควบคุมคอยล์เย็น

2.4 สร้างหน้าเว็บแอปพลิเคชัน ดังนี้

2.4.1 หน้าล็อกอินใช้งาน



รูปที่ 18 หน้าล็อกอินใช้งาน

หน้าล็อกอินเข้าใช้งานหน้านี้เป็นหน้าล็อกอินของระบบที่พัฒนาเพื่อการควบคุมกล่อง IoT สำหรับเครื่องปรับอากาศแบบแยกส่วน (Split-type Air Conditioner) ซึ่งถูกพัฒนาขึ้นภายใต้ชื่อโปรเจกต์โดยสถาบันการศึกษา SIET (School of Industrial Education and Technology) และ KMITL (King Mongkut's Institute of Technology Ladkrabang) หน้าล็อกอินนี้จะใช้สำหรับการเข้าถึงระบบที่เกี่ยวข้องกับการพัฒนาและควบคุมกล่อง IoT สำหรับเครื่องปรับอากาศแบบแยกส่วน

2.4.1.1 องค์ประกอบของหน้า

1. โลโก้ อยู่ด้านซ้ายบน มีโลโก้ของ SIET และ KMITL แสดงสถาบันที่เกี่ยวข้องกับการพัฒนาโปรเจกต์นี้
2. ชื่อโปรเจกต์ Development of IoT Controlling Box for Split-type Air Conditioner ซึ่งเป็นชื่อของโปรเจกต์พัฒนาเพื่อควบคุมกล่อง IoT สำหรับเครื่องปรับอากาศแบบแยกส่วน
3. ฟอर्मล็อกอิน มีฟิลด์กรอกข้อมูลผู้ใช้ (Username) และรหัสผ่าน (Password) และปุ่ม Login เพื่อเข้าสู่ระบบ
4. ลิงก์เพิ่มเติม มีตัวเลือกให้ผู้ใช้สมัครสมาชิกใหม่ (Register) หรือกู้คืนรหัสผ่าน (Forgot Password)

2.4.1.2 รายละเอียดคำสั่งหน้าเว็บ Login

ในโค้ดนี้เป็นหน้าเว็บแบบเทมเพลตที่ใช้ Vue 3 และ TypeScript โดยโค้ดนี้ถูกออกแบบเพื่อใช้เป็นหน้า Login สำหรับโปรเจกต์ที่เกี่ยวข้องกับการพัฒนา IoT Controlling Box สำหรับเครื่องปรับอากาศแบบ Split-type มีองค์ประกอบและการทำงาน ดังนี้

1. โครงสร้างของหน้าเว็บ (Template) หน้าเว็บนี้ถูกจัดให้อยู่ใน div หลักซึ่งเป็นโครงสร้างที่ใช้จัดการการแสดงผลของเนื้อหา โดยใช้คลาส flex justify-center items-center h-screen bg-background relative เพื่อกำหนดการจัดตำแหน่งให้อยู่

ตรงกลางของหน้าจอ Logo อยู่มุมซ้ายบนของหน้าเว็บ โดยมีการแทรกโลโก้จากลิงก์ https://siet.kmitl.ac.th/sites/siet/themes/kmitl_siet/logo.svg ในส่วนของข้อความตรงกลาง แสดงข้อความหัวเรื่องว่า "Development of IoT Controlling Box for Split-type Air Conditioner" อยู่ตรงกลางของหน้าเว็บ ซึ่งใช้คลาส absolute top-4 left-1/2 เพื่อจัดตำแหน่งกลางและพอนต์ใหญ่สำหรับเน้นข้อความ กล่อง Login กล่องสีขาวตรงกลางของหน้าเว็บประกอบด้วย หัวข้อ Login ขนาดใหญ่ แบบฟอร์มสำหรับใส่ Username และ Password ปุ่มสำหรับยืนยันการ Login และมีการตรวจสอบว่าผู้ใช้ใส่ข้อมูลครบหรือไม่ ลิงก์สำหรับ Register และ Forgot Password ซึ่งจะนำไปสู่หน้าอื่นที่เตรียมไว้

2. การทำงานของระบบ (Script Setup) โค้ดใช้การทำงานผ่าน script setup ที่ใช้ TypeScript เพื่อจัดการสถานะและฟังก์ชันต่างๆ โดยใช้ ref จาก Vue เพื่อติดตามสถานะของข้อมูล Username และ Password ที่ผู้ใช้งานกรอกเข้ามา ส่วน Methods

2.1 login() : ฟังก์ชันที่ทำงานเมื่อผู้ใช้งานกดปุ่ม Login โดยใช้ฟังก์ชัน signInWithEmailAndPassword() จาก Firebase เพื่อยืนยันตัวตน หากเข้าสู่ระบบสำเร็จจะถูกเปลี่ยนหน้าไปยัง /dashboard หากไม่สำเร็จจะแสดงข้อความเตือน

2.2 openRegisterModal() : เมื่อผู้ใช้งานกดปุ่ม Register ระบบจะนำไปยังหน้า /register

2.3 openForgotPasswordModal() : เมื่อผู้ใช้งานกด Forgot Password ระบบจะนำไปยังหน้า /forgot-password

3. การกำหนดสไตล์ (CSS) ส่วนของสไตล์เป็นการปรับแต่งให้หน้าเว็บดูสวยงามขึ้น พื้นหลัง ตั้งค่า background-image เป็นรูปภาพที่ถูกกำหนดลิงก์ไว้ (<https://files.123freevectors.com/wp-content/original/107062-light-orange-abstract.jpg>) โลโก้ ปรับขนาดและตำแหน่งของโลโก้ให้อยู่มุมบนซ้าย และมีขนาดที่

ใหญ่ขึ้น (ความกว้าง 20rem) กล่อง Login ตั้งค่าให้กล่อง Login มีขอบมน (rounded) และมีเงา (shadow) เพื่อให้ดูมีมิติและสะอาดตา ปุ่มและฟอร์ม: มีการกำหนดสีและลักษณะของปุ่มให้สามารถเปลี่ยนสีเมื่อ hover (bg-blue-500 จะเปลี่ยนเป็น bg-blue-700 เมื่อ hover) รวมถึงปรับแต่งขอบของ input ให้เปลี่ยนสีเมื่อถูกคลิก (focus)

4. การจัดการฟอร์ม ฟอร์มใช้ v-model ในการเชื่อมต่อข้อมูลจากอินพุตกับตัวแปร username และ password ฟังก์ชัน @submit.prevent="login" ถูกใช้เพื่อหยุดการรีเฟรชหน้าเว็บและเรียกใช้ฟังก์ชัน login() แทน

5. ระบบเตือน มีการตั้งค่าการแสดงความเตือน Please enter both username and password ซึ่งจะถูกแสดงเมื่อการ login ล้มเหลว เช่น กรณีผู้ใช้กรอกข้อมูลไม่ครบหรือกรอกผิด

```
popku-master - LoginForm.vue
V LoginForm.vue x
1 <template>
2   <div class="flex justify-center items-center h-screen bg-background relative">
3     <!-- โลโก้มหาวิทยาลัย -->
4     
5
6     <!-- ข้อความที่อยู่กลาง -->
7     <h3 class="absolute top-4 left-1/2 transform -translate-x-1/2 text-lg font-semibold text-gray-800 text-center">
8       Development of IoT Controlling Box for Split-type Air Conditioner
9     </h3>
10
11     <!-- กล่อง Login -->
12     <div class="bg-white p-10 rounded shadow-md max-w-lg w-full sm:w-auto flex flex-col items-center">
13       <h2 class="text-3xl mb-6 font-bold text-gray-800">Login</h2>
14       <form @submit.prevent="login" class="space-y-4 w-full">
15         <div class="w-full">
16           <label for="username" class="block text-sm font-medium text-gray-700">Username</label>
17           <input v-model="username" type="text" id="username" class="mt-1 p-2 w-full border-gray-300 rounded-md">
18         </div>
19         <div class="w-full">
20           <label for="password" class="block text-sm font-medium text-gray-700">Password</label>
21           <input v-model="password" type="password" id="password" class="mt-1 p-2 w-full border-gray-300 rounded-md">
22         </div>
23         <button type="submit" class="w-full bg-blue-500 hover:bg-blue-700 text-white p-3 rounded-md transition duration-300">Login</button>
24         <p v-if="showWarning" class="text-red-500 text-sm mt-2">Please enter both username and password.</p>
25       </form>
26       <div class="text-center mt-4">
27         <button @click="openRegisterModal" class="text-blue-500 cursor-pointer">Register</button>
28         <span class="mx-2 text-gray-700">|</span>
29         <button @click="openForgotPasswordModal" class="text-blue-500 cursor-pointer">Forgot Password</button>
30       </div>
31     </div>
32   </div>
33 </template>
34
```

รูปที่ 19 แสดงโค้ดหน้าลือคอิน

```

<script setup lang="ts">
import { ref } from "vue";
import { useRouter } from "vue-router";
import { getAuth, signInWithEmailAndPassword } from "firebase/auth";

// Properties
const router = useRouter();
const username = ref( value: "");
const password = ref( value: "");
const showWarning = ref( value: false);

// Methods
function login() { Show usages
  const auth = getAuth();

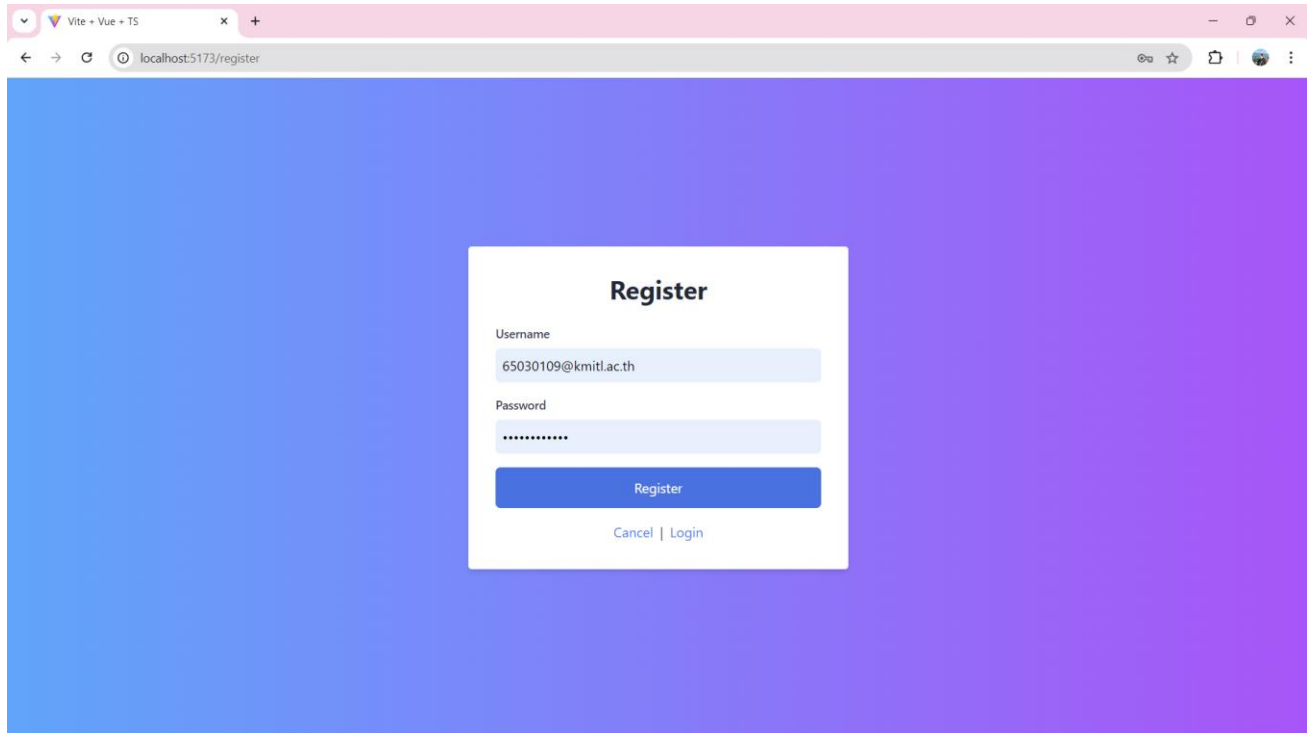
  signInWithEmailAndPassword(auth, username.value, password.value)
    .then(() => {
      // Signed in successfully, navigate to LedControlView.vue
      router.push("/dashboard");
    })
    .catch((error) => {
      // Display error message if login fails
      showWarning.value = true;
      console.error("Login error:", error.code, error.message);
    });
}

function openRegisterModal() { Show usages
  // Navigate to RegisterFormView.vue when Register button is clicked
  router.push("/register");
}

```

รูปที่ 20 แสดงการเชื่อมต่อ Firebase หน้า Login

2.4.2 หน้า Register



รูปที่ 21 หน้า Register

หน้าลงทะเบียน (Register) ในเว็บแอปพลิเคชันหนึ่ง โดยมีองค์ประกอบหลักคือ ชื่อแบบฟอร์ม "Register" หรือ ลงทะเบียน ช่องกรอกข้อมูล ช่องกรอกสำหรับ Username (ชื่อผู้ใช้งาน) ช่องกรอกสำหรับ Password (รหัสผ่าน) ปุ่ม Register สำหรับส่งข้อมูลการลงทะเบียน ลิงก์ Cancel | Login อยู่ด้านล่างปุ่มสำหรับยกเลิกหรือกลับไปหน้าล็อกอิน

ซึ่งโค้ดนี้สร้างส่วนประกอบ (component) ของฟอร์มลงทะเบียนในเว็บแอปพลิเคชันที่ใช้ Vue.js ร่วมกับ Firebase Authentication เพื่อการจัดการผู้ใช้ รายละเอียดของแต่ละส่วน ดังนี้

1. Template (HTML) องค์ประกอบของ UI ถูกสร้างขึ้นโดยใช้ Tailwind CSS ซึ่งมีลักษณะคือพื้นหลัง: bg-gradient-to-r from-blue-400 to-purple-500 ใช้สำหรับสร้างพื้นหลังไล่เฉดสีจากฟ้าไปม่วง ฟอร์มการลงทะเบียนที่อยู่ใน div ที่มีการจัดวางตรงกลางของหน้าจอ (flex justify-center items-center h-screen) ภายในฟอร์มมีสองช่องกรอก ช่องกรอก Username (input ที่มี v-model="registerUsername") สำหรับรับค่าอีเมลผู้ใช้ ช่องกรอก

Password (input ที่มี v-model="registerPassword") สำหรับรหัสผ่าน ปุ่ม Register ส่งฟอร์มลงทะเบียนและปุ่ม Cancel กับ Login จะเรียกฟังก์ชันเพื่อเปลี่ยนหน้า ปุ่ม Register มีเอฟเฟกต์ hover (hover:bg-blue-700) ที่จะเปลี่ยนสีเมื่อเมาส์ชี้ และมีการเปลี่ยนเส้นทางผู้ใช้ไปที่หน้าล็อกอินหลังจากลงทะเบียนเสร็จ

2. Script (JavaScript/TypeScript) ในส่วนนี้ใช้ Vue Composition API ผ่านฟังก์ชัน setup() มีการนำเข้า ref เพื่อสร้างตัวแปรที่เชื่อมกับการกรอกข้อมูลของผู้ใช้ (เช่น registerUsername, registerPassword) useRouter เพื่อเปลี่ยนเส้นทาง (routing) ไปที่หน้าล็อกอินหลังจากลงทะเบียนเสร็จ Firebase Authentication (createUserWithEmailAndPassword และ getAuth) สำหรับการสร้างบัญชีผู้ใช้ใหม่ใน Firebase ตัวแปรหลัก registerUsername และ registerPassword เป็นตัวแปรที่ใช้สำหรับเก็บค่าจากฟิลด์กรอกข้อมูลในฟอร์ม ฟังก์ชัน registerUser ฟังก์ชันนี้ถูกเรียกเมื่อผู้ใช้คลิกปุ่ม "Register" เช็คว่าผู้ใช้ปัจจุบันยังไม่ได้ล็อกอิน (currentUser เป็น null) สร้างผู้ใช้ใหม่ใน Firebase ด้วยข้อมูลที่ได้จากฟิลด์กรอก (createUserWithEmailAndPassword) หลังจากการสร้างบัญชีเสร็จสิ้น จะออกจากระบบ (auth.signOut()) และรอ 1.5 วินาที จากนั้นเปลี่ยนเส้นทางไปยังหน้า /login ฟังก์ชัน closeRegisterModal และ openRegisterModal ทั้งสองฟังก์ชันนี้ทำหน้าที่เปลี่ยนหน้าไปที่ /login เมื่อผู้ใช้คลิกที่ปุ่ม Cancel หรือ Login

3. Styles (CSS) ส่วนนี้ใช้ CSS ที่ถูกกำหนดเฉพาะสำหรับฟอร์มและปุ่มการโฟกัสที่ช่องกรอกข้อมูล (input[type="text"]:focus, input[type="password"]:focus) จะเปลี่ยนสีขอบเป็นสีฟ้า (#4a90e2) ปุ่มมีการเปลี่ยน opacity เมื่อเมาส์ชี้ (button:hover { opacity: 0.8; }) ลิงก์หรือข้อความที่มีการ hover จะมีเอฟเฟกต์ขีดเส้นใต้ (a:hover { text-decoration: underline; })

```

1  <template>
2  <div class="flex justify-center items-center h-screen bg-gradient-to-r from-blue-400 to-purple-500">
3    <div class="bg-white p-8 rounded shadow-md max-w-md w-full">
4      <h2 class="text-3xl mb-6 text-center font-bold text-gray-800">Register</h2>
5      <form @submit.prevent="registerUser" class="space-y-4">
6        <div class="flex flex-col">
7          <label for="registerUsername" class="block text-sm font-medium text-gray-700 mb-1">Username</label>
8          <input v-model="registerUsername" type="text" id="registerUsername" class="mt-1 p-2 block w-full border-gray-300 rounded-md">
9        </div>
10       <div class="flex flex-col">
11         <label for="registerPassword" class="block text-sm font-medium text-gray-700 mb-1">Password</label>
12         <input v-model="registerPassword" type="password" id="registerPassword" class="mt-1 p-2 block w-full border-gray-300 rounded-md">
13       </div>
14       <button type="submit" class="w-full bg-blue-500 hover:bg-blue-700 text-white p-3 rounded-md transition duration-300">Register</button>
15     </form>
16     <div class="text-center mt-4">
17       <button @click="closeRegisterModal" class="text-blue-500 cursor-pointer">Cancel</button>
18       <span class="mx-2 text-gray-700">|</span>
19       <button @click="openRegisterModal" class="text-blue-500 cursor-pointer">Login</button>
20     </div>
21   </div>
22 </div>
23 </template>
24
25 <script lang="ts">

```

รูปที่ 22 แสดงโค้ดหน้า Register

```

> import ... // Import createUserWithEmailAndPassword function

export default { Show usages
  setup() {
    const router = useRouter();
    const registerUsername = ref( value: "");
    const registerPassword = ref( value: "");

    async function registerUser() { Show usages
      try {
        const auth = getAuth();
        const currentUser = auth.currentUser;
        if (!currentUser) {
          await createUserWithEmailAndPassword(auth, registerUsername.value, registerPassword.value);
          await auth.signOut();
          setTimeout( handler: () => {
            router.push("/login");
          }, timeout: 1500);
        }
      } catch (error) {
        console.error(error);
      }
    }

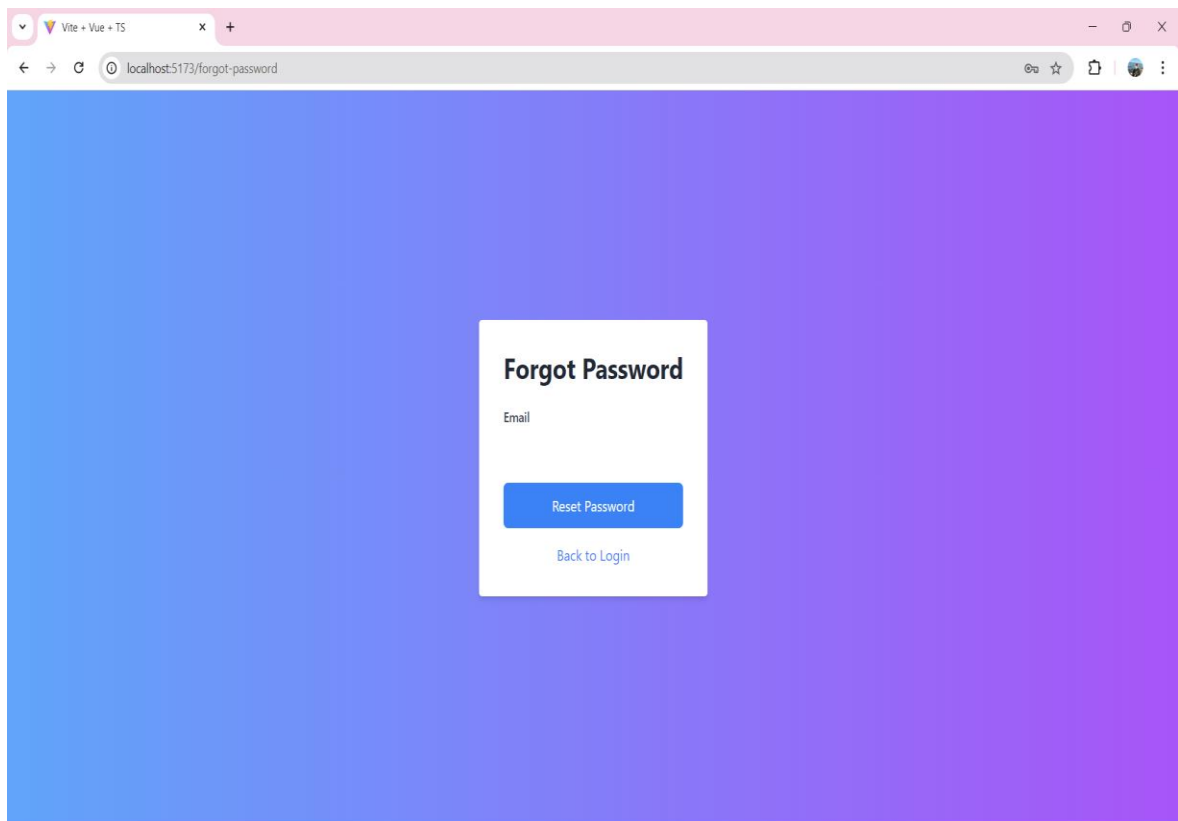
    function closeRegisterModal() { Show usages
      router.push("/login");
    }

    function openRegisterModal() { Show usages
      router.push("/login");
    }
  }
}

```

รูปที่ 23 แสดงการเชื่อมต่อ Firebase หน้า Register

2.4.3 หน้า Forgot Password



รูปที่ 24 แสดงหน้า Forgot Password

หน้านี้เป็นหน้าสำหรับ "ลืมรหัสผ่าน" ซึ่งมักจะใช้ในเว็บไซต์หรือแอปพลิเคชันเพื่อให้ผู้ใช้สามารถรีเซ็ตรหัสผ่านได้หากลืมรหัสผ่าน โค้ดนี้สร้างหน้า "Forgot Password" สำหรับการรีเซ็ตรหัสผ่านด้วย Vue.js และ Firebase โดยใช้ฟอร์มกรอกอีเมลของผู้ใช้ แล้วเรียกใช้ Firebase เพื่อส่งอีเมลรีเซ็ตรหัสผ่าน ถ้าเกิดข้อผิดพลาดจะแสดงข้อความข้อผิดพลาด ซึ่งองค์ประกอบหลักคือ หัวข้อ ("Forgot Password") แสดงจุดประสงค์ของหน้านี้ว่าเป็นการกู้คืนรหัสผ่านช่องใส่อีเมล โดยให้ผู้ใช้กรอกอีเมลที่เชื่อมโยงกับบัญชีของตน ปุ่ม "Reset Password" เมื่อกรอกอีเมลแล้ว ผู้ใช้สามารถกดปุ่มนี้เพื่อเริ่มกระบวนการรีเซ็ตรหัสผ่านโดยจะต้องยืนยันตัวตนผ่าน email ลิงก์ "Back to Login" ให้ผู้ใช้กลับไปยังหน้าเข้าสู่ระบบหากจำรหัสผ่านได้ ซึ่งประกอบด้วยดังนี้

2.4.3.1 Template (โครงสร้าง HTML)

`<div class="flex justify-center items-center h-screen bg-gradient-to-r from-blue-400 to-purple-500">` กำหนดพื้นหลังเป็นไล่สีจากสีฟ้าไปสีม่วงและจัดตำแหน่งให้หน้าอยู่ตรงกลางของจอ `<div class="bg-white p-8 rounded shadow-md max-w-md w-full sm:w-auto flex flex-col items-center">` เป็นกล่องพื้นหลังสีขาวที่มีขอบโค้งและเงา ขนาดสูงสุดไม่เกิน 640px (max-width) โดยใช้ Flexbox จัดให้เนื้อหาอยู่ตรงกลาง `<h2>` หัวข้อ "Forgot Password" แสดงอยู่ด้านบนด้วยขนาดตัวอักษรใหญ่ (text-3xl) `<form>` φόρμ สำหรับใส่อีเมลและปุ่มกดส่งคำขอรีเซ็ตรหัสผ่าน `<input v-model="email" type="email">`: ช่องกรอกอีเมล ใช้ v-model เพื่อเชื่อมข้อมูลในตัวแปร email `<button type="submit">Reset Password</button>`: ปุ่มสำหรับส่งฟอร์มเพื่อรีเซ็ตรหัสผ่าน `<p v-if="showError">`: แสดงข้อความข้อผิดพลาดหากมีการเกิดข้อผิดพลาดในการส่งคำขอ `<button @click="goToLogin" class="mt-4 text-blue-500 cursor-pointer">Back to Login</button>`: ปุ่มที่ใช้สำหรับนำผู้ใช้กลับไปหน้าจอเข้าสู่ระบบ

2.4.3.2 Script (โค้ด JavaScript)

import { ref } from "vue"; นำเข้า ref จาก Vue เพื่อสร้างตัวแปร reactive import { getAuth, sendPasswordResetEmail } from "firebase/auth"; นำเข้า getAuth และ sendPasswordResetEmail จาก Firebase สำหรับการทำงานเกี่ยวกับการรีเซ็ตรหัสผ่าน ตัวแปร email, showError, errorMessage: ใช้ ref เพื่อสร้างตัวแปรที่สามารถปรับปรุ่ค่าและแสดงผลแบบ reactive ฟังก์ชัน resetPassword: ฟังก์ชันสำหรับรีเซ็ตรหัสผ่าน โดยใช้ sendPasswordResetEmail จาก Firebase เพื่อส่งอีเมลรีเซ็ตรหัสผ่าน และถ้าเกิดข้อผิดพลาด

จะแสดงข้อความผ่านตัวแปร errorMessage ฟังก์ชัน goToLogin: ใช้สำหรับนำผู้ใช้กลับไป
หน้าเข้าสู่ระบบ (/login)

```
1 <template>
2   <div class="flex justify-center items-center h-screen bg-gradient-to-r from-blue-400 to-purple-500">
3     <div class="bg-white p-8 rounded shadow-md max-w-md w-full sm:w-auto flex flex-col items-center">
4       <h2 class="text-3xl mb-6 font-bold text-gray-800">Forgot Password</h2>
5       <form @submit.prevent="resetPassword" class="space-y-4 w-full">
6         <div class="w-full">
7           <label for="email" class="block text-sm font-medium text-gray-700">Email</label>
8           <input v-model="email" type="email" id="email" class="mt-1 p-2 w-full border-gray-300 rounded-md">
9         </div>
10        <button type="submit" class="w-full bg-blue-500 hover:bg-blue-700 text-white p-3 rounded-md transition duration-300">Reset Password</button>
11        <p v-if="showError" class="text-red-500 text-sm mt-2">{{ errorMessage }}</p>
12      </form>
13      <button @click="goToLogin" class="mt-4 text-blue-500 cursor-pointer">Back to Login</button>
14    </div>
15  </div>
16 </template>
17
18 <script setup lang="ts">
```

รูปที่ 25 แสดงโค้ดหน้า Forgot Password

```
import ...

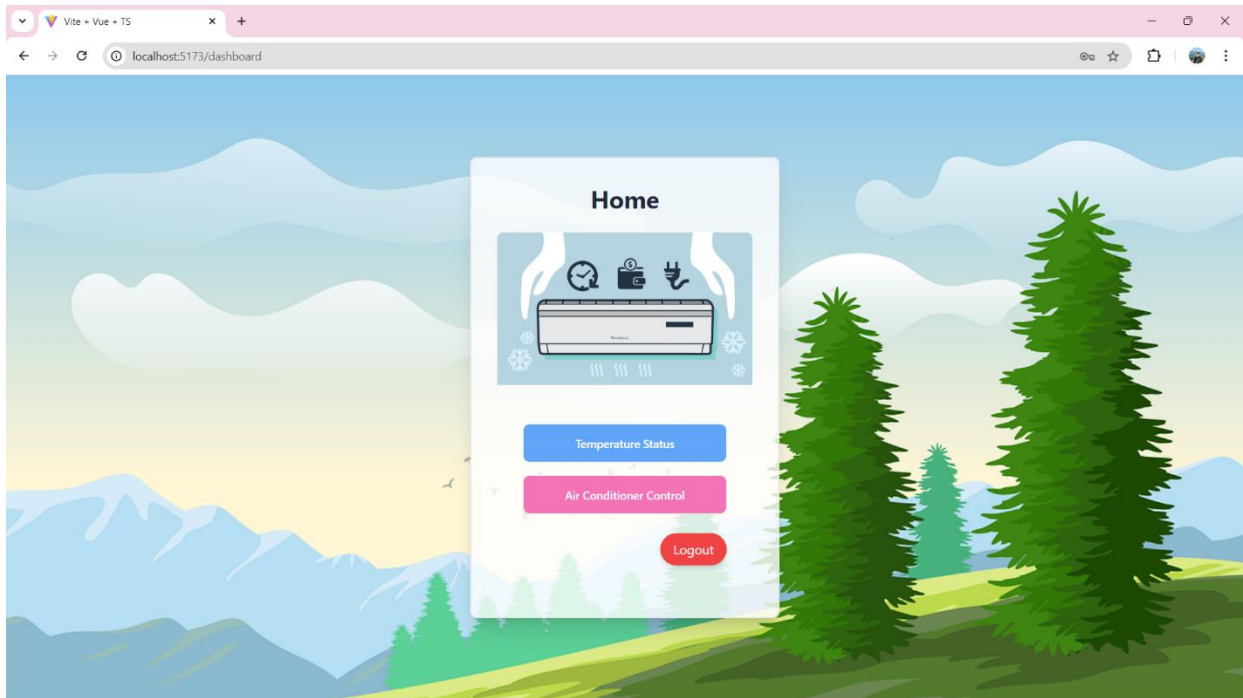
const email = ref( value: "");
const showError = ref( value: false);
const errorMessage = ref( value: "");

async function resetPassword() { Show usages
  try {
    const auth = getAuth();
    await sendPasswordResetEmail(auth, email.value);
    // Password reset email sent successfully
    // You can add a success message here if needed
  } catch (error) {
    // Check if error is an instance of Error
    if (error instanceof Error) {
      showError.value = true;
      errorMessage.value = error.message;
    } else {
      // If error is not an instance of Error, handle it accordingly
      console.error("Unknown error occurred:", error);
    }
  }
}

function goToLogin() { Show usages
  router.push('/login');
}
```

รูปที่ 26 แสดงการเชื่อมต่อ Firebase หน้า Forgot Password

2.4.4 หน้า Home



รูปที่ 27 แสดงหน้า Home

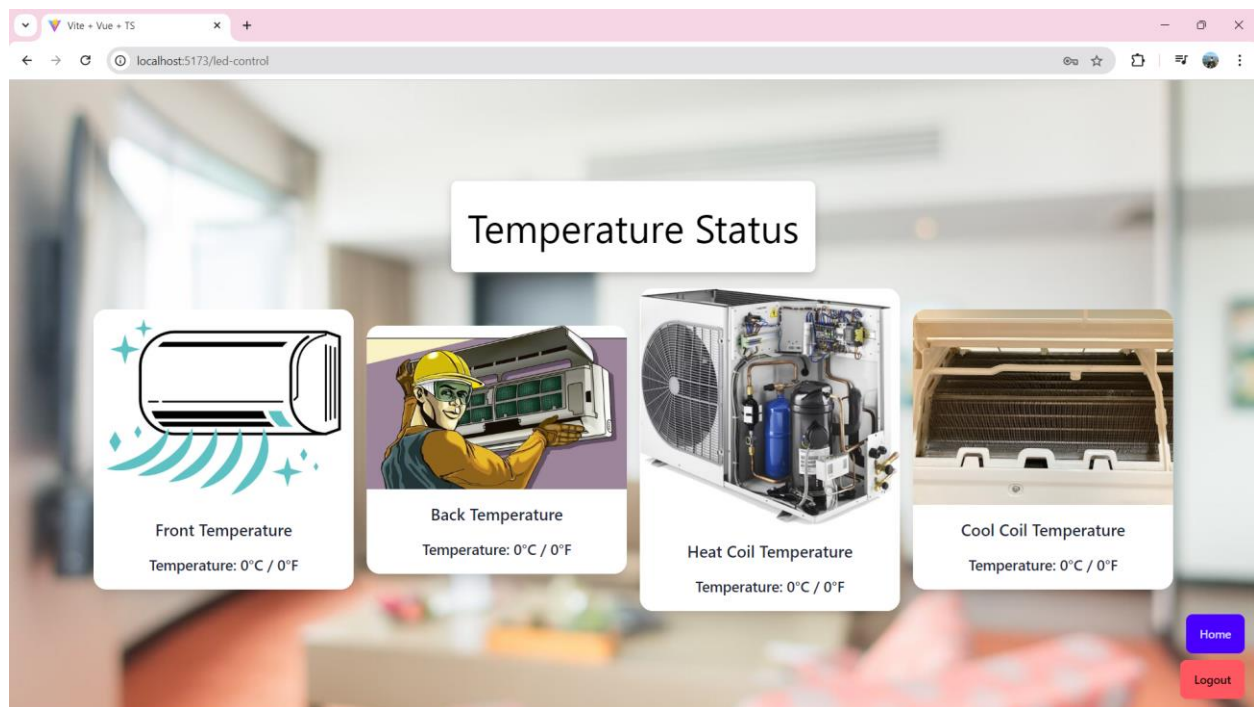
หน้าจอโฮมของระบบควบคุมแอร์ ซึ่งประกอบด้วยองค์ประกอบต่าง ๆ คือ หัวข้อ "Home" - แสดงอยู่ด้านบนสุดของกล่องแสดงข้อมูล เป็นการบอกว่าเป็นหน้าแรกของระบบ ปุ่ม "Temperature Status" (ปุ่มสีฟ้า) - เป็นปุ่มที่ใช้ตรวจสอบสถานะของอุณหภูมิของจุดต่างๆ ซึ่ง ปุ่ม "Air Conditioner Control" (ปุ่มสีชมพู) สำหรับการควบคุมแอร์โดยตรง เช่น การปรับอุณหภูมิหรือโหมดต่าง ๆ ปุ่ม "Logout" (ปุ่มสีแดง) - ใช้สำหรับออกจากระบบ

```
Dashboard.vue
1 <template>
2   <div class="flex justify-center items-center h-screen bg-snow">
3     <div class="card w-96 bg-opacity-80 bg-white shadow-xl border-2 border-gray-200 rounded-lg p-8">
4       <h1 class="text-3xl font-bold text-gray-800 mb-6 text-center">Home</h1>
5
6       <figure>
7         
8       </figure>
9       <div class="card-body">
10        <div class="flex flex-col space-y-4 mt-4">
11          <router-link to="/led-control" class="btn bg-blue-400 hover:bg-blue-600 text-white">Temperature Status</router-link>
12          <router-link to="/air-conditioner" class="btn bg-pink-400 hover:bg-pink-600 text-white">Air Conditioner Control</router-link>
13        </div>
14        <div class="flex justify-end mt-4">
15          <button @click="logout" class="bg-red-500 hover:bg-red-700 text-white px-4 py-2 rounded-full shadow-md">
16            Logout
17          </button>
18        </div>
19      </div>
20    </div>
21  </div>
22 </template>
```

รูปที่ 28 แสดงโค้ดหน้า

ซึ่งโค้ดนี้เป็นโค้ด Vue.js ที่สร้างหน้าแดชบอร์ดสำหรับควบคุมแอร์และเช็คสถานะอุณหภูมิ โค้ดถูกเขียนในรูปแบบ TypeScript และใช้ Firebase Authentication สำหรับการเข้าสู่ระบบและออกจากระบบ รายละเอียดของโค้ดประกอบไปด้วย โครงสร้างของหน้าแดชบอร์ด ใช้การจัดตำแหน่งด้วย flexbox เพื่อให้หน้าเว็บอยู่กึ่งกลางหน้าจอ (justify-center, items-center) และทำให้เต็มความสูงของหน้าจอ (h-screen) Card กล่อง div ที่ใช้สำหรับแสดงเนื้อหา ถูกจัดการให้มีเงา (shadow-xl), ขอบโค้งมน (rounded-lg), และมีพื้นที่ขอบ (border-2) หัวข้อ "Home" ถูกกำหนดให้เป็นหัวข้อหลักที่มีขนาดตัวอักษรใหญ่ (text-3xl) และเป็นตัวหนา (font-bold) รูปภาพ ใช้ img แสดงภาพของแผงควบคุมที่เป็นภาพเคลื่อนไหว (gif) และมีขอบโค้งมนบน (rounded-t-lg) ปุ่มควบคุม Temperature Status และ Air Conditioner Control ใช้ router-link สำหรับการเปลี่ยนเส้นทางไปยังหน้าอื่นภายในแอป เช่นหน้า /led-control และ /air-conditioner ปุ่มออกจากระบบ (Logout) ใช้ button พร้อมกับ @click ที่ผูกกับฟังก์ชัน logout

2.4.5 หน้า Temperature Status



รูปที่ 29 แสดงหน้า Temperature Status

หน้าเว็บที่แสดงสถานะอุณหภูมิ โดยมีส่วนประกอบหลักดังนี้ หัวข้อ "Temperature Status" ซึ่งหมายถึงสถานะของอุณหภูมิของส่วนต่าง ๆ ของช่องแสดงผลลัพธ์อุณหภูมิ 4 ช่องที่แสดงข้อมูลอุณหภูมิของแต่ละ

ส่วนคือ Front Temperature (อุณหภูมิด้านหน้า): มีสัญลักษณ์แอร์ที่กำลังปล่อยอากาศออกมา แสดงว่าเป็นการ
วัดอุณหภูมิที่ด้านหน้าของเครื่อง ซึ่งแสดง °C / °F Back Temperature (อุณหภูมิด้านหลัง): มีภาพการ์ตูนของ
ช่างกำลังทำงานกับแอร์ แสดงอุณหภูมิที่ด้านหลัง ซึ่งแสดง °C / °F Heat Coil Temperature (อุณหภูมิของ
คอยล์ความร้อน): มีภาพของเครื่องทำความเย็น แสดงอุณหภูมิของคอยล์ที่ใช้สำหรับทำความร้อน ซึ่งแสดง °C / °F
Cool Coil Temperature (อุณหภูมิของคอยล์เย็น): มีภาพคอยล์ในเครื่องปรับอากาศ แสดงอุณหภูมิของคอยล์ที่
ใช้ทำความเย็น ซึ่งแสดง °C / °F ปุ่ม Home และ Logout ที่มุมล่างขวา

```
<template>
<div class="full-background">
  <div class="title-container">
    <h1 class="title">Temperature Status</h1> <!-- Wrapped title in a div -->
  </div>
  <div class="flex flex-wrap justify-center items-center space-x-4">
    <div class="card card-compact w-80 bg-base-100 shadow-xl mb-4">
      <figure>
        
      </figure>
      <div class="card-body flex flex-col items-center justify-center text-center">
        <h2 class="card-title">Front Temperature</h2>
        <div class="mt-1">
          <h3 class="text-lg font-semibold text-center">Temperature: {{ sensorTemperatureCelsius }}°C / {{ sensorTemperatureFahrenheit }}°F</h3>
        </div>
      </div>
    </div>
    <div class="card card-compact w-80 bg-base-100 shadow-xl mb-4">
      <figure>
        
      </figure>
      <div class="card-body flex flex-col items-center justify-center text-center">
        <h2 class="card-title">Back Temperature</h2>
        <div class="mt-1">
          <h3 class="text-lg font-semibold text-center">Temperature: {{ sensorTemperatureCelsius }}°C / {{ sensorTemperatureFahrenheit }}°F</h3>
        </div>
      </div>
    </div>
  </div>
</div>
```

รูปที่ 30 แสดงโค้ดหน้า Temperature Status

```

</div>
<div class="card card-compact w-80 bg-base-100 shadow-xl mb-4">
  <figure>
    
  </figure>
  <div class="card-body flex flex-col items-center justify-center text-center">
    <h2 class="card-title">Heat Coil Temperature</h2>
    <div class="mt-1">
      <h3 class="text-lg font-semibold text-center">Temperature: {{ sensorTemperatureCelsius }}°C / {{ sensorTemperatureFahrenheit }}°F</h3>
    </div>
  </div>
</div>
<div class="card card-compact w-80 bg-base-100 shadow-xl mb-4">
  <figure>
    
  </figure>
  <div class="card-body flex flex-col items-center justify-center text-center">
    <h2 class="card-title">Cool Coil Temperature</h2>
    <div class="mt-1">
      <h3 class="text-lg font-semibold text-center">Temperature: {{ sensorTemperatureCelsius }}°C / {{ sensorTemperatureFahrenheit }}°F</h3>
    </div>
  </div>
</div>
</div>
<div class="fixed bottom-4 right-4 flex flex-col items-end space-y-2">
  <button @click="goToDashboard" class="btn btn-primary">Home</button>
  <button @click="logout" class="btn btn-error">Logout</button>
</div>
</div>
</template>

```

รูปที่ 31 แสดงโค้ดหน้า Temperature Status

โค้ดนี้เป็นหน้า Vue.js ที่ออกแบบมาเพื่อแสดงสถานะอุณหภูมิของส่วนต่าง ๆ ของเครื่องปรับอากาศ ซึ่งแสดงหน้าเว็บที่เกี่ยวกับสถานะอุณหภูมิ โดยรายละเอียดของโค้ดมีดังนี้ คือ องค์ประกอบของหน้า มีการใช้โครงสร้าง template ซึ่งเป็นการสร้างโครงร่างของหน้าเว็บใน Vue.js หน้าเว็บนี้มีพื้นหลังแบบเต็ม (full-background) และมีหัวข้อหลัก "Temperature Status" ที่ห่อหุ้มด้วย div (title-container) เพื่อความสวยงามและการจัดตำแหน่งการแสดงผลสถานะอุณหภูมิการ์ด จำนวน 4 ใบสำหรับแสดงสถานะอุณหภูมิของอุปกรณ์ 4 ส่วน ได้แก่ Front Temperature Back Temperature Heat Coil Temperature Cool Coil Temperature แต่ละกล่องประกอบด้วย รูปภาพที่เกี่ยวข้องกับแต่ละส่วนของเครื่องปรับอากาศ หัวข้อย่อยบอกชื่อส่วน (เช่น Front Temperature) อุณหภูมิที่จะแสดงในรูปแบบทั้งองศาเซลเซียส (sensorTemperatureCelsius) และฟาเรนไฮต์ (sensorTemperatureFahrenheit) ซึ่งค่าจะถูกเชื่อมโยงกับข้อมูลจาก Vue.js data model การจัดตำแหน่งใช้ flex เพื่อจัดเรียงการ์ดในรูปแบบยืดหยุ่น flex-wrap ช่วยให้การการ์ดห่อเมื่อพื้นที่ไม่พอ justify-center และ items-center ใช้เพื่อจัดกึ่งกลางการ์ดในแนวนอนและแนวตั้ง มีการเว้นช่องว่างระหว่างการ์ด (space-x-4) เพื่อให้มีระยะห่างที่เหมาะสม ทุกการ์ดถูกตกแต่งด้วย card card-compact w-80 bg-base-100 shadow-xl mb-4 ซึ่งทำให้ดูมีระเบียบและมีเงา ปุ่ม "Home" และ "Logout" ซึ่งการทำงานกับ Vue.js ตัวแปร

sensorTemperatureCelsius และ sensorTemperatureFahrenheit แสดงอุณหภูมิที่อัปเดตจากข้อมูลเซ็นเซอร์ซึ่งมาจาก Vue.js data model หรือ state management การคลิกปุ่ม "Home" และ "Logout" จะทำงานผ่าน Vue methods ที่เกี่ยวข้อง (goToDashboard และ logout)

3. รายละเอียดในการปฏิบัติงาน (Details)

3.1 แยกกันศึกษาตามหัวข้อที่กำหนด ดังนี้

- ศึกษาองค์ประกอบและหลักการทำงานของคอยล์เย็น
- ชนิดของเซ็นเซอร์วัดอุณหภูมิที่ใช้ในระบบเครื่องปรับอากาศและหลักการทำงานของเซ็นเซอร์

3.2 แยกกันทำงานตามหัวข้อที่กำหนด ดังนี้

3.2.1 ศึกษาองค์ประกอบจากคอยล์เย็นจากอุปกรณ์จริง

- ลงมือแกะคอยล์เย็น
- ศึกษาและสืบค้นข้อมูลเกี่ยวกับอุปกรณ์แต่ละตัว รวมทั้งศึกษาหลักการทำงานของอุปกรณ์

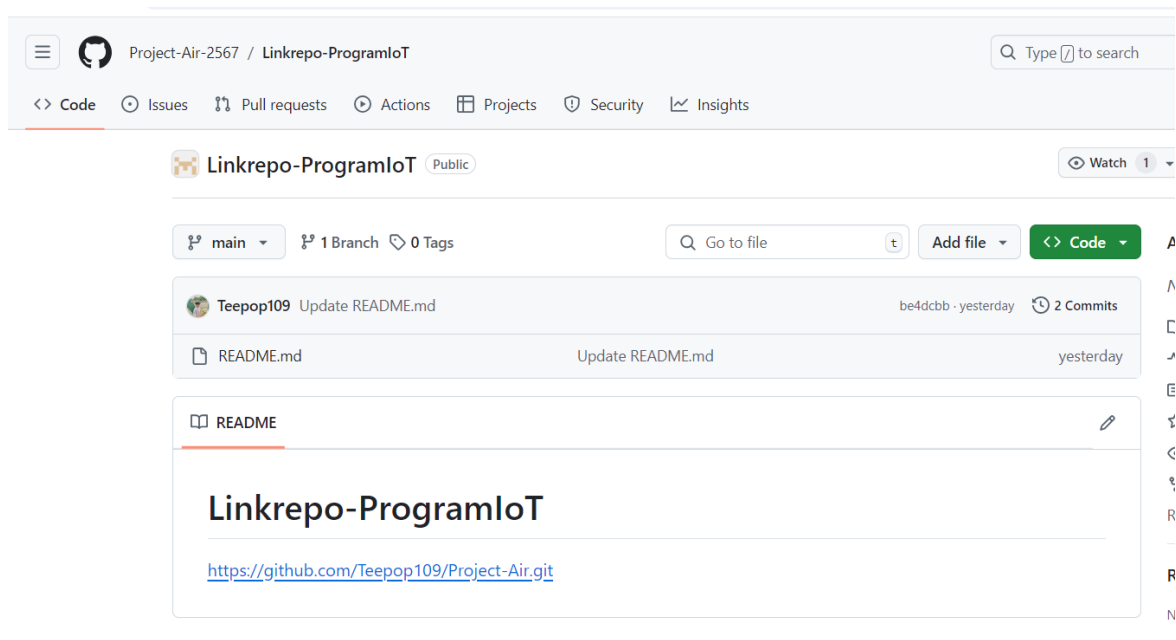


รูปที่ 32 งานเทคนิคศึกษาองค์ประกอบและหลักการทำงานของคอยล์เย็นจากอุปกรณ์จริง



รูปที่ 33 งานเทคนิคศึกษาองค์ประกอบและหลักการทำงานของคอลย์เย็นจากอุปกรณ์จริง

3.2.2 เขียนเว็บแอปพลิเคชันหน้าล็คอิน , Register Forgot , Password , Home , Temperature Status



รูปที่ 34 หน้า Repositories GitHub

Teepop109 Initial commit		5bf5394 · yesterday	🕒 1 Commit
📁 .vscode	Initial commit	yesterday	
📁 my-vite-app/vite-project	Initial commit	yesterday	
📁 popku-master	Initial commit	yesterday	
📁 public	Initial commit	yesterday	
📁 src	Initial commit	yesterday	
📄 .gitignore	Initial commit	yesterday	
📄 README.md	Initial commit	yesterday	
📄 index.html	Initial commit	yesterday	
📄 package-lock.json	Initial commit	yesterday	
📄 package.json	Initial commit	yesterday	
📄 postcss.config.js	Initial commit	yesterday	
📄 tailwind.config.js	Initial commit	yesterday	
📄 tsconfig.json	Initial commit	yesterday	
📄 tsconfig.node.json	Initial commit	yesterday	
📄 vite.config.ts	Initial commit	yesterday	

รูปที่ 35 หน้า Repositories GitHub ที่ใช้ในการทำ web application

4. แผนการปฏิบัติงานในสัปดาห์ต่อไป (Plans)

4.1 วัดค่าอุณหภูมิจาก sensor

4.2 เขียน web application ให้สามารถรับค่าอุณหภูมิจากบอร์ดได้

5. บันทึกเพิ่มเติม (Open Issues)

ลงชื่อ โกศล ธรรม อาจารย์ที่ปรึกษา

วันที่ 29 กันยายน 2567

เอกสารอ้างอิง

ช่างบ่ม channel. (2564, มกราคม 24). [ช่างแอร์ EP.13] FCU : Fan coil unit คอยล์เย็นแอร์บ้าน มีอุปกรณ์อะไรบ้าง "ช่างบ่ม" [วิดีโอ]. Youtube. <https://youtu.be/5qlA6iFX3X8?si=PoeuPFeyg5qifTtY>

สาระเรื่องแอร์บ้าน.(2566). คอยล์เย็นแอร์ คืออะไร มีอะไรที่ขึ้นไหนบ้าง. สืบค้นเมื่อ 24 กันยายน 2567. จาก <https://www.okarhaiair.com/what-aircon/>

Mono tool professional.(2564). ส่วนประกอบของคอยล์เย็นแอร์มีอะไรบ้าง. สืบค้นเมื่อ 24 กันยายน 2567. จาก <https://www.monotoolthailand.com/content/9776/-ingredient>

นายสุพรรณ ผ่องผาง.(2563). เทอร์มิสเตอร์ (Thermistor) คือเซ็นเซอร์วัดอุณหภูมิ. สืบค้นเมื่อวันที่ 26 กันยายน 2567. จาก

<https://www.neonics.co.th/thermometers/%E0%B9%80%E0%B8%97%E0%B8%AD%E0%B8%A3%E0%B9%8C%E0%B8%A1%E0%B8%B4%E0%B8%AA%E0%B9%80%E0%B8%95%E0%B8%AD%E0%B8%A3%E0%B9%8C-thermistor-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B9%80%E0%B8%8B%E0%B9%87%E0%B8%99%E0%B9%80%E0%B8%8B%E0%B8%AD%E0%B8%A3%E0%B9%8C%E0%B8%A7%E0%B8%B1%E0%B8%94%E0%B8%AD%E0%B8%B8%E0%B8%93%E0%B8%AB%E0%B8%A0%E0%B8%B9%E0%B8%A1%E0%B8%B4.html>

Sangchaigroup.(2564). Thermostat คืออะไร.สืบค้นเมื่อวันที่ 27 กันยายน 2567. จาก <https://sangchaigroup.com/thermostat-vs-temperature-controller/>

สำนักงานใหญ่และศูนย์โลจิสติกส์.(2566). เซ็นเซอร์ตรวจจับวัตถุ คือ?.สืบค้นเมื่อวันที่ 27 กันยายน 2567. จาก <https://www.sumipol.com/knowledge/object-detection-sensor/>

บริษัท นิโอนิคส์ จำกัด.(2558). รู้จักเซ็นเซอร์อุณหภูมิ (Temperature sensor). สืบค้นเมื่อวันที่ 27 กันยายน 2567. จาก <https://www.tools.in.th/temperature/thermometer-sensor/>

ความรู้จากช่างแอร์.(2560).อาการ สายเซ็นเซอร์อุณหภูมิเสีย. สืบค้นเมื่อวันที่ 28 กันยายน 2567.จาก

<https://www.okarhaiair.com/air-sensor/>

ภาคทฤษฎีมอเตอร์สวิตแอร์ [วิดีโอออนไลน์].//สืบค้นเมื่อ/28 กันยายน 2567,//จาก/

<https://www.youtube.com/watch?v=bBq1-EwFGv0>