



ระบบตรวจวัดอุณหภูมิและความชื้นภายในห้องเรียน
Classroom Temperature and Humidity Monitoring System

ผู้จัดทำ
วิภาวี พันธุ์เสถียร
ปณิดา กองแสน

โครงการระดับประกาศนียบัตรวิชาชีพชั้นสูงนี้เป็นส่วนหนึ่งของการศึกษา
ตามหลักสูตรเทคโนโลยีคอมพิวเตอร์ ประเภทวิชาอุตสาหกรรม
สาขาวิชาเทคโนโลยีคอมพิวเตอร์ ปีพุทธศักราช 2567

ระบบตรวจวัดอุณหภูมิและความชื้นภายในห้องเรียน
Classroom Temperature and Humidity Monitoring System

ผู้จัดทำ
วิภาวี พันธุ์เสถียร
ปณิดา กองแสน

โครงการระดับประกาศนียบัตรวิชาชีพชั้นสูงนี้เป็นส่วนหนึ่งของการศึกษา
ตามหลักสูตรเทคโนโลยีคอมพิวเตอร์ ประเภทวิชาอุตสาหกรรม
สาขาวิชาเทคโนโลยีคอมพิวเตอร์ ปีพุทธศักราช 2567

บทคัดย่อ

ชื่อเรื่อง	ระบบตรวจวัดอุณหภูมิและความชื้นภายในห้องเรียน
ผู้จัดทำ	นางสาววิภาวี พันธุ์เสถียร นางสาวปณิดา กองแสน
อาจารย์ที่ปรึกษา	อาจารย์นพพร ไกรรักษ์
ระดับการศึกษา	ประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.)
สาขาวิชา	เทคโนโลยีคอมพิวเตอร์
ปีการศึกษา	พุทธศักราช 2568

โครงการนี้มีวัตถุประสงค์เพื่อพัฒนาระบบตรวจวัดอุณหภูมิและความชื้นภายในห้องเรียน โดยใช้เซ็นเซอร์เอสคูนิย่เก่า ซึ่งเชื่อมต่อผ่านเครือข่ายภายใน หรือแลน และดึงข้อมูลจากอุปกรณ์โดยตรงผ่านโปรแกรมไพธอนโดยใช้ไลบรารี ไทน์ทูย่า ระบบสามารถแสดงผลแบบ Real-Time ผ่าน Website ที่พัฒนาขึ้นเอง ทำให้สามารถติดตามสภาพแวดล้อมได้อย่างแม่นยำและสะดวกสบายยิ่งขึ้น

การพัฒนาระบบนี้เน้นการประยุกต์ใช้เทคโนโลยี อินเทอร์เน็ตของสรรพสิ่งหรือไอโอที เพื่อเชื่อมต่อและจัดการข้อมูลจากเซ็นเซอร์โดยตรง โดยไม่ต้องพึ่งพาระบบคลาวด์ภายนอก ทั้งยังสามารถขยายไปควบคุมอุปกรณ์ไฟฟ้าในอนาคต เช่น การเปิด – ปิดพัดลมหรือเครื่องปรับอากาศอัตโนมัติเมื่ออุณหภูมิเกินค่าที่กำหนด

Abstract

Project	Classroom Temperature and Humidity Monitoring System
Students	Miss Wipawee Phansathian Miss Panida Kongsan
Advisor	Mr. Nopphon Kairak
Education Level	Higher Vocational Certificate
Field of Study	Computer Technology
Academic Year	2025 (Buddhist Era 2568)

This project aims to develop a Classroom Temperature and Humidity Monitoring System using the S09 sensor. The sensor connects via a local network (LAN), and data is retrieved directly using Python and the TinyTuya library. The system displays real-time data on a custom-developed website, allowing users to monitor environmental conditions efficiently and accurately.

The system applies Internet of Things (IoT) technology to manage sensor data without relying on external cloud platforms. It also offers potential for future expansion, such as integrating automatic control of electrical devices like fans or air conditioners when the temperature exceeds a predefined threshold.

กิตติกรรมประกาศ

โครงการ ระบบตรวจวัดอุณหภูมิและความชื้นภายในห้องเรียน นี้สำเร็จลุล่วงด้วยการสนับสนุนและคำแนะนำอันมีค่าจากหลายฝ่าย ข้าพเจ้าขอขอบพระคุณ อาจารย์ที่ปรึกษา อาจารย์นพพร ไกรรักษ์และคณะอาจารย์ในแผนกวิชาช่างเทคนิคคอมพิวเตอร์ที่ได้ให้คำแนะนำช่วยเหลือ และชี้แนะแนวทางการทำโครงการอย่างใกล้ชิด รวมถึงให้คำปรึกษาตลอดระยะเวลาการดำเนินงาน ขอขอบคุณ สถาบันวิทยาลัยเทคนิคนครสวรรค์ ที่ได้จัดเตรียมทรัพยากรและอุปกรณ์ต่างๆ สำหรับการทดลองในโครงการครั้งนี้ และขอขอบคุณเพื่อนร่วมชั้นทุกท่านที่ได้ช่วยเหลือและให้กำลังใจในระหว่างการทำงาน นอกจากนี้ ข้าพเจ้ายังขอขอบคุณ ครอบครัว ที่ให้การสนับสนุนด้านกำลังใจและความเข้าใจ ซึ่งเป็นแรงผลักดันสำคัญที่ทำให้โครงการนี้สำเร็จลุล่วง ขอขอบคุณทุกท่านอีกครั้งสำหรับความช่วยเหลือและการสนับสนุนที่มีค่า ทำให้ข้าพเจ้าสามารถบรรลุเป้าหมายของโครงการได้ตามที่ตั้งใจไว้

คำนำ

ในปัจจุบัน เทคโนโลยีมีบทบาทสำคัญอย่างยิ่งในการพัฒนาคุณภาพชีวิต โดยเฉพาะอย่างยิ่งในด้านการตรวจสอบและควบคุมสภาพแวดล้อม เทคโนโลยี Internet of Things (IoT) ได้เข้ามามีบทบาทในการเชื่อมโยงอุปกรณ์ต่าง ๆ ให้สามารถสื่อสารและแลกเปลี่ยนข้อมูลกันได้แบบ Real-Time ส่งผลให้การเฝ้าระวังสภาพแวดล้อมเป็นไปอย่างมีประสิทธิภาพและแม่นยำมากยิ่งขึ้น

โครงการเรื่องระบบตรวจวัดอุณหภูมิและความชื้นภายในห้องเรียน จัดทำขึ้นเพื่อศึกษาและพัฒนาาระบบที่สามารถตรวจวัดค่าอุณหภูมิและความชื้นจากเซ็นเซอร์ S09 และส่งข้อมูลแบบ Real-Time ผ่านแพลตฟอร์ม Tuya Cloud โดยแสดงผลผ่านแอปพลิเคชัน Smart Life และสามารถนำข้อมูลไปแสดงผลบน Website หรือระบบอื่น ๆ ได้อย่างสะดวก ผู้จัดทำมีความมุ่งหวังว่า ระบบนี้จะสามารถนำไปประยุกต์ใช้ในการควบคุมอุปกรณ์ไฟฟ้าอัตโนมัติ เช่น พัดลมหรือเครื่องปรับอากาศ เพื่อให้การจัดการสภาพแวดล้อมในพื้นที่ต่าง ๆ มีประสิทธิภาพมากยิ่งขึ้น

ผู้จัดทำหวังเป็นอย่างยิ่งว่า โครงการนี้จะเป็นประโยชน์แก่ผู้สนใจ และสามารถนำไปประยุกต์ใช้หรือต่อยอดในอนาคต หากมีข้อผิดพลาดประการใด ผู้จัดทำขอน้อมรับไว้เพื่อการปรับปรุงและพัฒนาในโอกาสต่อไป

สารบัญ

บทที่	หน้า
บทคัดย่อภาษาไทย	(1)
บทคัดย่อภาษาอังกฤษ	(2)
กิตติกรรมประกาศ	(3)
คำนำ	(4)
สารบัญ	(5)
สารบัญตาราง	(7)
สารบัญภาพ	(8)
1 บทนำ	1
ความเป็นมาและความสำคัญของโครงการ	1
วัตถุประสงค์ของโครงการ	1
ขอบเขตของโครงการ	1
ข้อจำกัดของโครงการ	3
สมมติฐานของโครงการ	3
คำจำกัดความที่ใช้ในโครงการ	3
ประโยชน์ที่คาดว่าจะได้รับ	3
2 แนวคิดทฤษฎีและโครงการที่เกี่ยวข้อง	4
เซ็นเซอร์วัดอุณหภูมิและความชื้น S09	4
การเชื่อมต่อและการสื่อสารผ่าน LAN	4
ระบบ Internet of Things (IoT)	4
Web Application และ Flask Framework	5
โครงการที่เกี่ยวข้อง	5
แผนภาพแนวคิดระบบ	5
โปรแกรมที่ใช้ในการพัฒนาระบบ	6
3 วิธีการดำเนินโครงการ	11
ศึกษาปัญหาและความต้องการ	11
กำหนดขอบเขตของโครงการ	11
ศึกษาทฤษฎีและเครื่องมือที่เกี่ยวข้อง	11
ออกแบบระบบและโครงสร้าง Website	12
พัฒนาระบบและเชื่อมต่อกับอุปกรณ์	12
การพัฒนา Website และ Web API	13
ติดตั้งและทดสอบระบบจริง	13
ปรับปรุงระบบตามข้อเสนอแนะ	14

สารบัญ (ต่อ)

บทที่	หน้า
จัดทำเอกสารและนำเสนองาน	14
4 ผลการทดสอบและประเมินโครงการ	15
ผลลัพธ์เชิงเทคนิค	15
การแสดงผลข้อมูลและวิเคราะห์ผลเบื้องต้น	15
การทดสอบประสิทธิภาพของระบบ	16
ผลการประเมินระบบ	17
5 สรุปผล ปัญหา และข้อเสนอแนะ	18
สรุปผลของโครงการ	18
ปัญหาและอุปสรรคที่พบ	18
ข้อเสนอแนะในการพัฒนาโครงการต่อไป	19
บทสรุปโดยรวม	20
บรรณานุกรม	21
ภาคผนวก	
ภาคผนวก ก ตัวอย่างโค้ด	22
ภาคผนวก ข แบบสอบถาม	23
ภาคผนวก ค การใช้ระบบตรวจวัดอุณหภูมิและความชื้นภายในห้องเรียน	24
ประวัติผู้จัดทำโครงการ	26

สารบัญตาราง

ตารางที่	หน้า
1.1 ตารางแสดงระยะเวลาในการศึกษาโครงการ	2
3.1 รายการอุปกรณ์และเครื่องมือที่ใช้ในการพัฒนาระบบ	12
4.1 ตารางแสดงการเปลี่ยนแปลงอุณหภูมิและความชื้นจากการทดสอบระบบ	16
4.2 แสดงผลการทดสอบความเสถียรของระบบ	16
4.3 ผลการประเมินด้านการออกแบบระบบและการติดตั้ง	17
4.4 ผลการประเมินด้านการใช้งานระบบโดยผู้ใช	17
5.1 ปัญหาที่พบและแนวทางการแก้ไข	19

สารบัญภาพ

ภาพที่	หน้า
2.1 แนวคิดระบบ	5
2.2 Smart Life	6
2.3 Tuya Cloud	6
2.4 Visual Studio Code	7
2.5 Python	7
2.6 Flask (Python Framework)	8
2.7 Google Chrome	8
2.8 HTML / CSS / JavaScript	9
2.9 ChatGPT	9
2.10 Advanced IP Scanner	10
3.1 ตัวอย่างโค้ดสำหรับดึงข้อมูล	13
3.2 ติดตั้งและทดสอบระบบ	14

บทที่ 1

บทนำ

ความเป็นมาและความสำคัญของโครงการ

ในปัจจุบัน สถานศึกษาหลายแห่งยังไม่มีระบบที่สามารถติดตามและควบคุมอุณหภูมิภายในห้องเรียนได้อย่างมีประสิทธิภาพ โดยเฉพาะห้องเรียนคอมพิวเตอร์ ซึ่งเป็นพื้นที่ที่มีการใช้งานอุปกรณ์อิเล็กทรอนิกส์จำนวนมาก เช่น คอมพิวเตอร์ โปรเจคเตอร์ และอุปกรณ์ต่อพ่วงต่าง ๆ ส่งผลให้เกิดความร้อนสะสมภายในห้อง หากไม่ได้รับการจัดการที่เหมาะสม อาจทำให้อุณหภูมิภายในห้องสูงเกินไปจนส่งผลกระทบต่อประสิทธิภาพของอุปกรณ์ และลดอายุการใช้งานของเครื่องมือเหล่านั้น

นอกจากนี้ อุณหภูมิที่ไม่เหมาะสมยังส่งผลต่อความรู้สึกสบายของผู้ใช้งาน เช่น นักเรียนและครู ซึ่งอาจทำให้สมาธิในการเรียนลดลง และเกิดผลกระทบต่อสุขภาพในระยะยาว โดยอุณหภูมิที่เหมาะสมควรอยู่ในช่วง 20 – 25 องศาเซลเซียส เพื่อเอื้อต่อการเรียนรู้และการทำงานอย่างมีประสิทธิภาพ

ด้วยเหตุนี้จึงเกิดแนวคิดในการพัฒนาระบบตรวจวัดอุณหภูมิและความชื้นภายในห้องเรียน โดยใช้เซ็นเซอร์ S09 ที่สามารถดึงข้อมูลได้โดยตรงผ่านเครือข่ายท้องถิ่น (LAN) ด้วยโปรแกรม Python และ Library TinyTuya โดยไม่ต้องพึ่งพา Cloud ภายนอก ระบบสามารถแสดงผลข้อมูลแบบ Real-Time ผ่าน Website ที่พัฒนาขึ้น ทำให้ผู้ใช้งานสามารถตรวจสอบค่าอุณหภูมิและความชื้นได้อย่างสะดวก และสามารถปรับการใช้งานอุปกรณ์ไฟฟ้าภายในห้องเรียนให้เหมาะสมตามสภาพอากาศที่ตรวจวัดได้ในขณะนั้น

วัตถุประสงค์ของโครงการ

1. เพื่อออกแบบและพัฒนาระบบตรวจวัดอุณหภูมิและความชื้นภายในห้องเรียนที่สามารถแสดงผลแบบ Real-Time ผ่าน Website
2. เพื่อประเมินประสิทธิภาพการทำงานของระบบต้นแบบ
3. เพื่อสร้างต้นแบบระบบที่สามารถขยายผลไปใช้งานในห้องเรียนอื่น ๆ หรืออาคารเรียนทั้งหมดได้

ขอบเขตของโครงการ

1.3.1 ขอบเขตของโครงการ

ตัวแปรต้น (Independent Variable)

ระบบตรวจวัดอุณหภูมิและความชื้นภายในห้องเรียน ซึ่งประกอบด้วยเซ็นเซอร์ S09 เชื่อมต่อผ่านเครือข่ายภายใน และแสดงผลผ่าน Website แบบ Real-Time โดยไม่ใช้ฐานข้อมูลหรือคลาวด์สำหรับเก็บข้อมูลย้อนหลัง

ตัวแปรตาม (Dependent Variable)

1. ค่าข้อมูลอุณหภูมิ (°C) และความชื้นสัมพัทธ์ (%RH) ที่แสดงผลจากระบบ
2. ความพึงพอใจของผู้ใช้งานต่อระบบในด้านต่าง ๆ เช่น ความแม่นยำ ความสะดวกในการใช้งาน และความสวยงามของ Website

1.3.2 ประชากรและกลุ่มตัวอย่าง

ประชากร ได้แก่ นักเรียนและนักศึกษาวิทยาลัยเทคนิคนครสวรรค์ แผนกวิชาช่างเทคนิคคอมพิวเตอร์ ปีการศึกษา 2568

กลุ่มตัวอย่าง คือ นักศึกษาระดับประกาศนียบัตรวิชาชีพชั้นสูงที่ 2 ห้อง 3 แผนกวิชาช่างเทคนิคคอมพิวเตอร์ วิทยาลัยเทคนิคนครสวรรค์ จำนวน 18 คน ปีการศึกษา 2568

โดยใช้วิธีการเลือกแบบ การสุ่มแบบเจาะจง (Purposive Sampling) จากผู้ที่ใช้ห้องเรียน 1742 เป็นประจำ

1.3.3 ระยะเวลาในการศึกษาโครงการ

โครงการนี้ดำเนินการในช่วง ภาคเรียนที่ 1 ปีการศึกษา 2568 โดยแผนการดำเนินงานมีรายละเอียดดังนี้

ตารางที่ 1.1 ตารางแสดงระยะเวลาในการศึกษาโครงการ

กิจกรรม	พ.ค.	มิ.ย.	ก.ค.	ส.ค.
1. ศึกษาปัญหาและความต้องการ	✓			
2. กำหนดขอบเขตโครงการ	✓			
3. ศึกษาเทคโนโลยีที่เกี่ยวข้อง (Python, Tuya API, HTML/CSS/JS)		✓		
4. ออกแบบระบบและ Website		✓	✓	
5. พัฒนาระบบและเชื่อมต่อ Tuya Cloud			✓	✓
6. ติดตั้งและทดสอบระบบในห้อง 1742				✓
7. ปรับปรุงระบบตามข้อเสนอแนะ				✓
8. จัดทำคู่มือและเอกสารโครงการ				✓
9. พบอาจารย์ที่ปรึกษา	✓	✓	✓	✓

จากตารางที่ 1.1 พบว่าโครงการใช้เวลาดำเนินการ 6 เดือน โดยแบ่งเป็นขั้นตอนต่าง ๆ ตั้งแต่การศึกษาข้อมูล ออกแบบระบบ พัฒนา ทดสอบ และจัดทำเอกสาร โดยมีการพบอาจารย์ที่ปรึกษาอย่างต่อเนื่องตลอดโครงการ

1.4 ข้อจำกัดของโครงการ

1. ระบบต้นแบบสามารถวัดและแสดงผลได้เฉพาะ อุณหภูมิและความชื้น โดยยังไม่รองรับการตรวจวัดข้อมูลประเภทอื่น เช่น ฝุ่น PM2.5 หรือแสงสว่าง
2. ระบบสามารถทำงานได้เฉพาะภายในเครือข่ายเดียวกับอุปกรณ์เซ็นเซอร์ (LAN)
3. ระบบเวอร์ชันปัจจุบันยังไม่มี การเชื่อมต่อกับ อุปกรณ์ควบคุมอัตโนมัติ เช่น พัดลม หรือเครื่องปรับอากาศ
4. การดำเนินโครงการจำกัดอยู่ภายในพื้นที่ ห้องเรียน 1742 แผนกคอมพิวเตอร์ของวิทยาลัยเทคนิคนครสวรรค์เท่านั้น

1.5 สมมติฐานของโครงการ

1. ระบบสามารถวัดและแสดงผลค่าอุณหภูมิและความชื้นได้อย่างถูกต้อง และแสดงผลแบบ Real-Time
2. ผู้ใช้งานสามารถเข้าถึงข้อมูลผ่าน Website ได้อย่างสะดวก และเข้าใจข้อมูลที่แสดงผลได้ง่าย
3. ผู้ใช้งานมีความพึงพอใจต่อระบบในระดับ มาก (ระดับ 4) ทั้งด้านความสะดวก ความแม่นยำ และความสวยงามของหน้าตา Website

1.6 คำจำกัดความที่ใช้ในโครงการ

เซ็นเซอร์ S09 คือ อุปกรณ์สำหรับตรวจวัดอุณหภูมิและความชื้น ที่สามารถเชื่อมต่อผ่านแอป Smart Life และดึงข้อมูลผ่าน LAN

TinyTuya คือ ไลบรารีภาษา Python สำหรับเชื่อมต่อกับอุปกรณ์ Tuya ภายในเครือข่าย LAN (Local Area Network) คือ เครือข่ายภายในที่ใช้เชื่อมต่อ Server กับอุปกรณ์

Website แสดงผล : หน้าเว็บที่พัฒนาเองเพื่อแสดงข้อมูลอุณหภูมิและความชื้นแบบ Real-Time

ประโยชน์ที่คาดว่าจะได้รับ

1. ได้ระบบวัดอุณหภูมิภายในห้องเรียนที่สามารถแสดงผลได้แบบ Real-Time บน Website ช่วยให้ผู้ใช้สามารถรับทราบข้อมูลสภาพแวดล้อมได้ทันที
2. ระบบตรวจวัดอุณหภูมิและความชื้นภายในห้องเรียนสามารถใช้งานได้มีประสิทธิภาพ
3. ได้ต้นแบบระบบที่สามารถต่อยอดและขยายผลไปใช้ในห้องเรียนอื่น ๆ หรืออาคารเรียนทั้งหลายได้ในอนาคต ช่วยเพิ่มประสิทธิภาพในการบริหารจัดการสภาพแวดล้อมภายในสถานศึกษา

บทที่ 2

แนวคิดทฤษฎีและโครงการที่เกี่ยวข้อง

ในการพัฒนาโครงการเรื่องระบบตรวจวัดอุณหภูมิและความชื้นภายในห้องเรียน ผู้จัดทำได้ศึกษาทฤษฎีและเทคโนโลยีที่เกี่ยวข้องในด้านระบบ Internet of Things (IoT) การรับและประมวลผลข้อมูลจากเซ็นเซอร์ การเชื่อมต่อผ่านเครือข่ายภายใน (Local Area Network) การเรียกใช้ API ภายในเครือข่าย และการแสดงผลข้อมูลผ่าน Website โดยได้ศึกษาองค์ความรู้ทั้งในด้านซอฟต์แวร์ ฮาร์ดแวร์ และแนวทางจากโครงการที่เกี่ยวข้อง เพื่อให้สามารถพัฒนาระบบที่มีประสิทธิภาพ ใช้งานได้จริง และมีความเหมาะสมต่อการประยุกต์ใช้ในพื้นที่จริง

2.1 เซ็นเซอร์วัดอุณหภูมิและความชื้น S09

เซ็นเซอร์ S09 เป็นอุปกรณ์อิเล็กทรอนิกส์อัจฉริยะที่สามารถวัดค่าอุณหภูมิและความชื้นได้ในตัวเดียว โดยสามารถเชื่อมต่อกับแพลตฟอร์มสมาร์ต เช่น แอป Smart Life ได้ และส่งข้อมูลผ่านเครือข่ายภายใน (LAN) โดยไม่จำเป็นต้องใช้โมโครคอนโทรลเลอร์เพิ่มเติม จึงมีความสะดวกในการติดตั้งและใช้งาน โดยเฉพาะอย่างยิ่งสำหรับงานระบบบ้านอัจฉริยะหรือพื้นที่ที่ต้องการเฝ้าระวังสภาพแวดล้อมอย่างต่อเนื่อง ระบบนี้ใช้การดึงข้อมูลโดยตรงจากอุปกรณ์ผ่านโปรโตคอลที่รองรับบนเครือข่าย LAN ซึ่งช่วยเพิ่มความรวดเร็วและเสถียรภาพในการสื่อสารข้อมูล

2.2 การเชื่อมต่อและการสื่อสารผ่าน LAN

ในโครงการนี้ ระบบไม่ได้ใช้การดึงข้อมูลผ่าน Tuya Cloud หรือบริการคลาวด์ภายนอก แต่เลือกใช้การสื่อสารผ่านเครือข่ายภายใน (Local Area Network) โดยตรงกับ Sensor S09 ผ่าน Protocol เฉพาะที่รองรับ ซึ่งช่วยให้ระบบมีความเร็วในการตอบสนองและลดความล่าช้าที่เกิดจากการส่งข้อมูลผ่านอินเทอร์เน็ตสาธารณะ นอกจากนี้ยังช่วยเพิ่มความปลอดภัยของข้อมูล เนื่องจากข้อมูลไม่ถูกส่งออกไปยัง Server ภายนอก ผู้พัฒนายังใช้ Library Python เช่น TinyTuya ในการดึงข้อมูลจากอุปกรณ์ในเครือข่าย LAN และนำข้อมูลที่ได้รับไปประมวลผลและแสดงผลผ่าน Website ที่พัฒนาขึ้นเอง

2.3 ระบบ Internet of Things (IoT)

ระบบ Internet of Things หรือ IoT คือเทคโนโลยีที่เชื่อมโยงอุปกรณ์อิเล็กทรอนิกส์ต่าง ๆ ผ่านเครือข่ายอินเทอร์เน็ตหรือเครือข่ายอื่น ๆ เพื่อให้สามารถสื่อสาร รับส่งข้อมูล และควบคุมกันได้โดยอัตโนมัติ โดยองค์ประกอบหลักของระบบ IoT ได้แก่ เซ็นเซอร์ สำหรับตรวจจับและรวบรวมข้อมูลจากสภาพแวดล้อม เครือข่ายสื่อสาร สำหรับส่งข้อมูลจากเซ็นเซอร์ไปยังระบบประมวลผล ระบบประมวลผลหรือแสดงผล เช่น Website ที่แสดงข้อมูลให้ผู้ใช้งาน หรือสั่งการกลับไปยังอุปกรณ์

2.4 Web Application และ Flask Framework

Flask เป็นเฟรมเวิร์คของภาษา Python ที่นิยมใช้สำหรับพัฒนาเว็บแอปพลิเคชันขนาดเล็กถึงขนาดกลาง โดยมีลักษณะเด่นคือความเบาและยืดหยุ่นสูง เหมาะสำหรับการนำข้อมูลจาก API หรือแหล่งข้อมูลต่าง ๆ มาแสดงผลแบบ Real-Time ในโครงการนี้ Flask ถูกใช้สำหรับดึงข้อมูลจากเซ็นเซอร์ผ่านโปรโตคอลบนเครือข่าย LAN ด้วยไลบรารี Python และสร้าง Website เพื่อแสดงผลข้อมูลอุณหภูมิและความชื้นให้กับผู้ใช้งานในรูปแบบที่เข้าใจง่ายและเข้าถึงได้จากทุกอุปกรณ์ที่มีเบราว์เซอร์

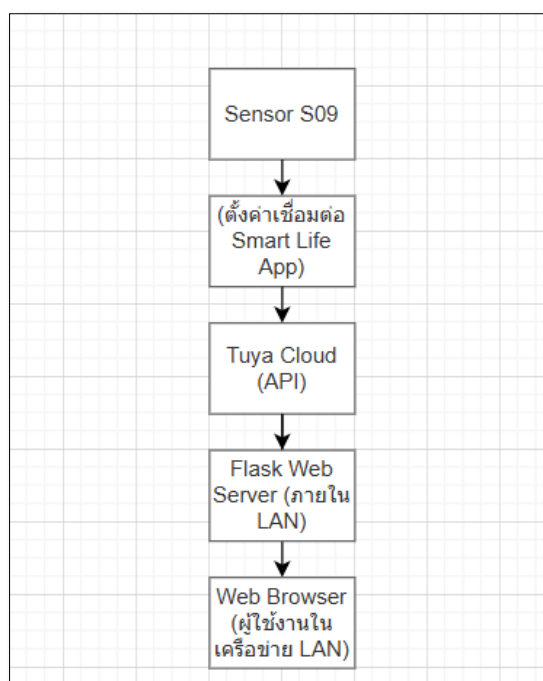
2.5 โครงการที่เกี่ยวข้อง

จากการศึกษา พบว่าเคยมีโครงการลักษณะคล้ายกัน เช่น โครงการระบบวัดอุณหภูมิและแสดงผลผ่าน Website โดยใช้เซ็นเซอร์ DHT11 ร่วมกับบอร์ด Arduino แล้วส่งข้อมูลผ่าน Wi-Fi ไปยัง Server ที่แสดงผลบนหน้า Website อย่างไรก็ตาม โครงการดังกล่าวจำเป็นต้องเขียนโปรแกรมฝั่งฮาร์ดแวร์เพื่อรับค่าจากเซ็นเซอร์และส่งข้อมูล ซึ่งมีความซับซ้อนในกระบวนการพัฒนา

ในโครงการนี้ ผู้จัดทำเลือกใช้เซ็นเซอร์อัจฉริยะ S09 ที่สามารถส่งข้อมูลผ่านแอป Smart Life และดึงข้อมูลผ่าน API โดยตรง จึงไม่จำเป็นต้องเขียนเฟิร์มแวร์เพิ่มเติม ส่งผลให้ระบบมีความเสถียร ลดเวลาในการพัฒนา และสะดวกต่อการนำไปประยุกต์ใช้งานหรือขยายผลในอนาคต

2.6 แผนภาพแนวคิดระบบ

ระบบที่พัฒนาขึ้นมีโครงสร้างการทำงานตามแผนภาพด้านล่าง โดยแสดงให้เห็นถึงลำดับขั้นการรับ - ส่งข้อมูล ตั้งแต่อุปกรณ์ตรวจวัดไปจนถึงการแสดงผลให้ผู้ใช้งาน



ภาพที่ 2.1 แนวคิดระบบ

จากภาพ 2.1 แผนภาพนี้แสดงให้เห็นถึงการส่งผ่านข้อมูลจากอุปกรณ์จริงสู่ผู้ใช้งาน ผ่านกระบวนการทางเทคนิคที่มีความปลอดภัยและสามารถใช้งานได้แบบ Real-Time

โปรแกรมที่ใช้ในการพัฒนาระบบ

1. แอป Smart Life



ภาพที่ 2.2 Smart Life

จากภาพที่ 2.2 ใช้ในการเชื่อมต่อและตั้งค่าเริ่มต้นอุปกรณ์ S09 Smart IR Sensor ให้เชื่อมต่อกับ Tuya Cloud

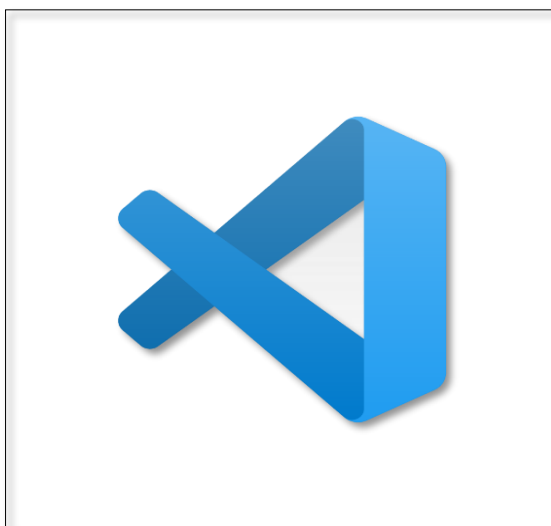
2. Tuya IoT Platform



ภาพที่ 2.3 Tuya cloud

จากภาพที่ 2.3 ใช้ในการสร้างและบริหารจัดการโปรเจกต์ IoT รวมถึงตั้งค่า Virtual ID, Local Key ของอุปกรณ์ และเข้าถึง Tuya API

3. Visual Studio Code



ภาพที่ 2.4 Visual Studio Code

จากภาพที่ 2.4 โปรแกรมสำหรับเขียนและแก้ไข Code HTML, CSS, JavaScript, Python หรือภาษาอื่นที่ใช้ในการดึงและแสดงผลข้อมูล

4. ภาษา Python



ภาพที่ 2.5 Python

จากภาพที่ 2.5 ใช้พัฒนาโปรแกรมฝั่ง Server ในการเชื่อมต่อกับ Tuya API และประมวลผลข้อมูล

5. Flask (Python Framework)



ภาพที่ 2.6 Flask (Python Framework)

จากภาพที่ 2.6 ใช้สร้างเว็บแอปพลิเคชันสำหรับแสดงผลอุณหภูมิแบบ Real-Time และย้อนหลัง

6. Web Browser (Google Chrome)



ภาพที่ 2.7 Google Chrome

จากภาพที่ 2.7 เป็นโปรแกรมเว็บเบราว์เซอร์ที่ใช้สำหรับทดสอบและแสดงผลหน้า Website ที่พัฒนาขึ้น โดยสามารถแสดงผลการทำงานของระบบ IoT ผ่าน Interface ที่ผู้ใช้สามารถเข้าถึงได้ง่าย

7. HTML / CSS / JavaScript



ภาพที่ 2.8 HTML / CSS / JavaScript

จากภาพที่ 2.8 ใช้สำหรับพัฒนา Interface ของหน้า Website เพื่อให้ผู้ใช้งานสามารถดูข้อมูลอันหนึ่งได้อย่างสะดวกและเข้าใจง่าย โดย HTML ใช้สำหรับโครงสร้างของเว็บเพจ CSS ใช้สำหรับจัดรูปแบบและตกแต่งหน้าตาเว็บและ JavaScript ใช้ในการควบคุมการทำงานแบบโต้ตอบและแสดงข้อมูลแบบ Real-Time

8. ChatGPT (โดย OpenAI)



ภาพที่ 2.9 ChatGPT

ใช้เป็นเครื่องมือช่วยในการอธิบายแนวคิดการทำงานของระบบ ให้คำแนะนำในการเขียนโค้ด รวมถึงช่วยจัดทำเอกสารประกอบโครงการ

9. Advanced IP Scanner



ภาพที่ 2.10 Advanced IP Scanner

จากภาพที่ 2.10 ใช้สำหรับสแกน IP Address ของอุปกรณ์ภายในเครือข่าย เช่น เซ็นเซอร์ S09 เพื่อค้นหาหมายเลข IP และ MAC Address ที่จำเป็นสำหรับการเชื่อมต่อและเรียกใช้งานข้อมูลผ่านระบบ Web Server ที่พัฒนาด้วย Python

บทที่ 3

วิธีการดำเนินโครงการ

การดำเนินโครงการระบบตรวจวัดอุณหภูมิและความชื้นด้วยเซ็นเซอร์ มีขั้นตอนและกระบวนการที่ชัดเจนเป็นลำดับ เพื่อให้สามารถออกแบบและพัฒนาระบบได้อย่างมีประสิทธิภาพ โดยแบ่งออกเป็น 9 ขั้นตอนต่าง ๆ ดังนี้ 1) ศึกษาปัญหาและความต้องการ 2) กำหนดขอบเขตของโครงการ 3) ศึกษาทฤษฎีและเครื่องมือที่เกี่ยวข้อง 4) ออกแบบระบบและโครงสร้าง Website 5) พัฒนาระบบและเชื่อมต่อกับอุปกรณ์ 6) การพัฒนา Website และ Web API 7) ติดตั้งและทดสอบระบบจริง 8) ปรับปรุงระบบตามข้อเสนอแนะ 9) จัดทำเอกสารและนำเสนอผลงาน

1.ศึกษาปัญหาและความต้องการ

ผู้จัดทำได้ศึกษาปัญหาในห้องเรียนที่ไม่มีระบบติดตามอุณหภูมิและความชื้นแบบ Real-Time และความต้องการในการควบคุมสภาพแวดล้อมอัตโนมัติ

2.กำหนดขอบเขตของโครงการ

2.1 ขอบเขตของโครงการ

ตัวแปรต้น คือ ระบบตรวจวัดอุณหภูมิและความชื้นภายใน

ตัวแปรตาม คือ ค่าข้อมูลอุณหภูมิ ($^{\circ}\text{C}$) และค่าความชื้นสัมพัทธ์ (%RH) ที่ได้จากการวัดโดยระบบ

2.2 ประชากรและกลุ่มตัวอย่าง

ประชากร ได้แก่ นักเรียนและนักศึกษาวิทยาลัยเทคนิคนครสวรรค์ แผนกวิชาช่างเทคนิคคอมพิวเตอร์ ปีการศึกษา 2568

กลุ่มตัวอย่าง คือ นักศึกษาระดับ ประกาศนียบัตรวิชาชีพ ปีที่ 2 ห้อง 3 แผนกวิชาช่างเทคนิคคอมพิวเตอร์ วิทยาลัยเทคนิคนครสวรรค์ จำนวน 18 คน ปีการศึกษา 2568

โดยใช้วิธีการเลือกแบบ การสุ่มแบบเจาะจง (Purposive Sampling) จากผู้ที่ใช้ห้องเรียน 1742 เป็นประจำ

2.3 ระยะเวลาในการศึกษาโครงการ

โครงการนี้ดำเนินการในภาคเรียนที่ 1 ปีการศึกษา 2568 ใช้เวลาประมาณ 6 เดือน โดยเริ่มจากการศึกษาปัญหาและกำหนดขอบเขต ศึกษาเทคโนโลยีที่เกี่ยวข้อง ออกแบบและพัฒนาระบบ เชื่อมต่อกับ Tuya Cloud ทดสอบและปรับปรุงระบบ รวมทั้งจัดทำเอกสาร โดยมีการพบอาจารย์ที่ปรึกษาอย่างต่อเนื่องตลอดโครงการ

3.ศึกษาทฤษฎีและเครื่องมือที่เกี่ยวข้อง

ผู้จัดทำศึกษาทฤษฎีด้าน Internet of Things (IoT), การเชื่อมต่ออุปกรณ์ผ่าน Tuya การใช้ Python, Flask, Web API รวมถึงเครื่องมือที่ใช้ในการพัฒนาดังตารางที่ 3.1

ตารางที่ 3.1 รายการอุปกรณ์และเครื่องมือที่ใช้ในการพัฒนาระบบ

รายการ	รายละเอียด
เซ็นเซอร์ S09	วัดอุณหภูมิและความชื้นภายในห้องเรียน
Smart Life App	ใช้เชื่อมต่ออุปกรณ์กับแพลตฟอร์ม Tuya
Tuya IoT Platform	ใช้ดูข้อมูล Local Key และ Device ID ของเซ็นเซอร์
Python + Flask	ใช้พัฒนา Web Server สำหรับดึงและแสดงข้อมูลจากเซ็นเซอร์
HTML/CSS/JavaScript	ใช้ออกแบบและพัฒนา Website ให้สามารถแสดงผลข้อมูลอย่างสวยงาม
Visual Studio Code	ใช้เป็นเครื่องมือเขียนและจัดการโค้ดในโครงการ
Web Browser	ใช้เปิดและแสดงผลหน้า Website ที่แสดงข้อมูลจากเซ็นเซอร์
Advanced IP Scanner	ใช้ค้นหา IP Address ของเซ็นเซอร์ภายในเครือข่าย

จากตารางที่ 3.1 แสดงให้เห็นถึงรายการอุปกรณ์และเครื่องมือที่ใช้ในการพัฒนาระบบ ตรวจวัดอุณหภูมิและความชื้น ซึ่งครอบคลุมทั้งส่วนของฮาร์ดแวร์ เช่น เซ็นเซอร์ S09 และซอฟต์แวร์ เช่น Python, Flask และ Visual Studio Code รวมถึงเครื่องมือสำหรับการสแกน IP อย่าง Advanced IP Scanner เพื่อค้นหา IP Address ของเซ็นเซอร์สำหรับเชื่อมต่อกับระบบเว็บเซิร์ฟเวอร์

4.ออกแบบระบบและโครงสร้าง Website

เริ่มจากการออกแบบสถาปัตยกรรมระบบ วางผังการทำงานของอุปกรณ์ การไหลของข้อมูลจากเซ็นเซอร์สู่ Website พร้อมทั้งออกแบบ Interface ของ Dashboard ให้แสดงผลข้อมูล Real-Time อย่างชัดเจน

5. พัฒนาระบบและเชื่อมต่อกับอุปกรณ์

เขียนโปรแกรมโดยใช้ภาษา Python และ Flask โดยใช้ดำเนินการเชื่อมต่ออุปกรณ์ตามลำดับ

- ติดตั้งแอป Smart Life → เพิ่มเซ็นเซอร์ S09
- สร้างโครงการใน Tuya IoT Platform และผูกกับ Smart Life
- ดึง Device ID และ Local Key จากเมนู API Explorer
- ใช้ Advanced IP Scanner หา IP Address ของอุปกรณ์
- เขียนโค้ด Python โดยใช้ Library TinyTuya เพื่อดึงข้อมูล JSON ผ่าน LAN

```

read_tuya.py > ...
1  import tinytuya
2  import json
3
4  # -- ข้อมูลสำหรับเชื่อมต่ออุปกรณ์ --
5  DEVICE_ID = "a363fd4a6a60153af3bhdg" # ID ของอุปกรณ์
6  DEVICE_IP = "192.168.0.104" # IP Address ของอุปกรณ์ในวง LAN
7  LOCAL_KEY = "@tU9ccyNOB$+2H02" # Local Key ของอุปกรณ์
8
9  # สร้าง Instant หรือ Object ของอุปกรณ์
10 try:
11     d = tinytuya.OutletDevice(DEVICE_ID, DEVICE_IP, LOCAL_KEY)
12
13     # ตั้งค่าเวอร์ชันของโปรโตคอล (ส่วนใหญ่เป็น 3.3)
14     d.set_version(3.4)
15
16     # -- ดึงข้อมูลสถานะจากอุปกรณ์ --
17     print("กำลังดึงข้อมูลจากอุปกรณ์...")
18     status_data = d.status()
19
20     # -- แสดงผลข้อมูลที่ได้รับ --
21     print("\n✅ ดึงข้อมูลสำเร็จ! สถานะของอุปกรณ์คือ:")
22     # ใช้ json.dumps เพื่อจัดรูปแบบการแสดงผลให้อ่านง่ายขึ้น
23     print(json.dumps(status_data, indent=4))
24
25 except Exception as e:
26     print(f"\n❌ เกิดข้อผิดพลาด: {e}")
27     print("กรุณาตรวจสอบข้อมูลการเชื่อมต่อ (IP, Device ID, Local Key) และตรวจสอบว่าอุปกรณ์เปิดใช้งานอยู่")

```

ภาพที่ 3.1 ตัวอย่างโค้ดสำหรับดึงข้อมูล

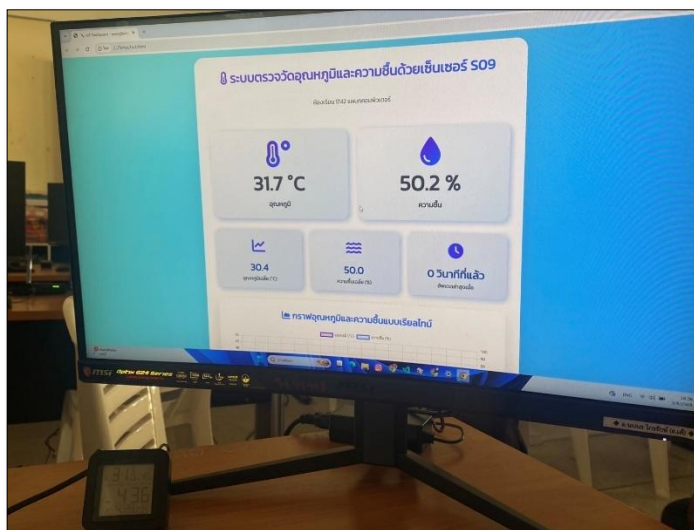
จากภาพที่ 3.1 ข้อมูลที่ได้รับจะอยู่ในรูปแบบ JSON ซึ่งสามารถนำไปประมวลผลต่อเพื่อแสดงผลอุณหภูมิและความชื้น

6. การพัฒนา Website และ Web API

สร้าง Web API ด้วย Flask ให้สามารถเรียกดูข้อมูลจากเซ็นเซอร์ในรูปแบบ JSON และสร้างหน้า Dashboard ด้วย HTML , CSS และ JavaScript เพื่อแสดงผลอุณหภูมิและความชื้นแบบ RealTime ผ่านกราฟและข้อความ

7. ติดตั้งและทดสอบระบบจริง

นำระบบต้นแบบไปติดตั้งที่ห้องเรียน 1742 และทำการทดสอบการเชื่อมต่อ ตรวจสอบความแม่นยำของข้อมูลที่แสดงบนหน้าเว็บ เปรียบเทียบกับอุปกรณ์วัดมาตรฐาน และตรวจสอบความเร็วในการ Update ข้อมูล



ภาพที่ 3.2 ติดตั้งและทดสอบระบบ

จากภาพที่ 3.2 เป็นขั้นตอนการติดตั้งระบบในห้องเรียน 1742 และทดสอบการแสดงผลข้อมูลจากเซ็นเซอร์ S09 ผ่านหน้า Website พร้อมตรวจสอบความแม่นยำและความเร็วในการอัปเดตข้อมูล

8. ปรับปรุงระบบตามข้อเสนอแนะ

หลังจากทดลองใช้งาน ได้รับข้อเสนอแนะให้ปรับอินเทอร์เฟซและปรับปรุงการแสดงผล เช่น แสดงเวลาอัปเดตล่าสุด, จำกัดจำนวนข้อมูลย้อนหลัง, ปรับขนาดตัวอักษร ฯลฯ จึงแก้ไขโค้ดให้ตอบโจทย์ผู้ใช้งานมากขึ้น

9. จัดทำเอกสารและนำเสนอผลงาน

จัดทำเอกสารรายงานและคู่มือประกอบการใช้งาน รวมถึงสไลด์นำเสนอ เพื่ออธิบายแนวคิด วิธีการพัฒนา และผลลัพธ์ของโครงการก่อนนำเสนอแก่คณะกรรมการประเมิน

บทที่ 4

ผลการทดสอบและประเมินโครงการ

การดำเนินโครงการระบบตรวจวัดอุณหภูมิและความชื้นภายในห้องเรียน ทำให้ผู้จัดทำเข้าใจขั้นตอนการดึงข้อมูลจากอุปกรณ์ IoT อย่างเป็นระบบ โดยไม่จำเป็นต้องเรียกใช้ API ผ่าน Cloud โดยตรง ส่งผลให้ระบบมีความเร็วในการตอบสนองมากขึ้นและสามารถควบคุมอุปกรณ์ได้ภายในเครือข่ายท้องถิ่น (LAN) โดยผลลัพธ์ของโครงการสามารถแบ่งออกเป็น 6 ประเด็นหลัก ดังนี้

4.1 ผลลัพธ์เชิงเทคนิค

4.1.1 การเชื่อมต่ออุปกรณ์ S09 เข้ากับแพลตฟอร์ม Smart Life

จากการดำเนินงานในขั้นต้น ผู้จัดทำสามารถเชื่อมต่อเซ็นเซอร์ S09 ซึ่งเป็นอุปกรณ์ IoT แบบสำเร็จรูปที่มีโมดูล Wi-Fi ในตัว เข้ากับแอปพลิเคชัน Smart Life ได้สำเร็จ โดยใช้ Interface ของแอปพลิเคชันในการควบคุมและรับค่าข้อมูลจากเซ็นเซอร์แบบ Real-Time ข้อมูลที่แสดงผลบนแอปประกอบด้วย ค่าอุณหภูมิในหน่วยองศาเซลเซียส ($^{\circ}\text{C}$) ค่าความชื้นสัมพัทธ์ในหน่วยเปอร์เซ็นต์ (%) จากการทดสอบพบว่า อุปกรณ์สามารถรายงานข้อมูลใหม่ได้ทุก ๆ 2-5 วินาที โดยไม่จำเป็นต้องรีเฟรชหน้าจอแอปพลิเคชันด้วยตนเอง

4.1.2 การดึงข้อมูลจากเซ็นเซอร์ผ่านโปรแกรม Python

แทนการเรียก API Cloud โดยตรง ผู้จัดทำได้ใช้กระบวนการเข้าถึงอุปกรณ์ภายในเครือข่ายท้องถิ่น ด้วยขั้นตอนต่อไปนี้ 1) นำ Device ID จากแอป Smart Life ไปใช้ค้นหา Local Key ผ่านเมนู API Explorer บนแพลตฟอร์ม Tuya IoT 2) ใช้โปรแกรม Advanced IP Scanner เพื่อตรวจสอบ IP Address ของอุปกรณ์ภายในเครือข่าย 3) ใช้ไลบรารี TinyTuya ในภาษา Python เพื่อเชื่อมต่ออุปกรณ์ผ่าน IP, Device ID และ Local Key 4) ดึงค่าข้อมูลอุณหภูมิและความชื้นโดยตรงแบบ Real-Time จากเซ็นเซอร์ ซึ่งวิธีนี้ช่วยลดความซับซ้อนในการขอ Token และเพิ่มความเร็วในการตอบสนองของระบบได้อย่างชัดเจน

4.2 การแสดงผลข้อมูลและวิเคราะห์ผลเบื้องต้น

ผู้จัดทำพัฒนาระบบแสดงผลข้อมูลจากเซ็นเซอร์เป็นกราฟเส้น โดยใช้ไลบรารี Matplotlib และ Datetime สำหรับแสดงผลแบบ Time Series

ตารางที่ 4.1 แสดงค่าการเปลี่ยนแปลงของอุณหภูมิและความชื้นในช่วงเวลาต่าง ๆ ที่ได้จากการทดสอบระบบ

เวลา (นาทีก)	อุณหภูมิ (°C)	ความชื้น (%)
0	29.5	62
5	30.0	63
10	30.3	61
15	30.7	60
20	31.1	59

จากตารางที่ 4.1 ข้อมูลพบว่า เมื่อเวลาผ่านไปในช่วงบ่าย อุณหภูมิภายในห้องมีแนวโน้มสูงขึ้น ขณะที่ความชื้นลดลงอย่างมีนัยสำคัญ

4.3 การทดสอบประสิทธิภาพของระบบ

ในการประเมินคุณภาพของระบบ ได้มีการออกแบบการทดลองย่อยเพื่อทดสอบความเสถียร ความแม่นยำ และความเร็วของการดึงข้อมูล โดยผลที่ได้มีดังนี้

ตารางที่ 4.2 แสดงผลการทดสอบความเสถียรของระบบ

ด้านที่ประเมิน	รายละเอียดการทดสอบ	ผลลัพธ์
ความแม่นยำ	เทียบกับเครื่องวัดมาตรฐาน	ค่าคลาดเคลื่อนน้อยกว่า ± 0.4 °C
ความเร็วในการดึงข้อมูล	ตรวจสอบระยะเวลาตอบสนองจากอุปกรณ์โดยตรง	เฉลี่ย 0.3–0.5 วินาที/คำขอ
ความเสถียร	ทดสอบต่อเนื่อง 48 ชั่วโมง	ไม่มีการตัดการเชื่อมต่อ

จากตารางที่ 4.2 จะเห็นได้ว่า ระบบมีความสามารถในการทำงานที่น่าเชื่อถือ โดยวัดค่าได้แม่นยำภายในช่วงที่ยอมรับได้ มีความรวดเร็วในการตอบสนองต่อคำขอข้อมูล และสามารถเชื่อมต่อได้อย่างต่อเนื่องโดยไม่หลุดการเชื่อมต่อ ทำให้ระบบนี้มีความพร้อมในการนำไปใช้งานจริงในสภาพแวดล้อมของห้องเรียนได้อย่างมีประสิทธิภาพ

4.4 ผลการประเมินระบบ

ตารางที่ 4.3 ผลการประเมินด้านการออกแบบระบบและการติดตั้ง

หัวข้อประเมิน	คะแนนเฉลี่ย (เต็ม 5)	ระดับความคิดเห็น
ความเหมาะสมของหน้าจอแสดงผล	4.3	ดี
ความสะดวกในการติดตั้งระบบ	4.2	ดี
ความแข็งแรงของอุปกรณ์และเซนเซอร์	4.4	ดีมาก
ความปลอดภัยในการใช้งาน	4.3	ดี
ความสวยงามของการจัดวางอุปกรณ์	4.2	ดี
รวมเฉลี่ย	4.30	ดี

จากตารางที่ 4.3 พบว่า กลุ่มตัวอย่างประเมินระบบในระดับ ดี โดยเฉลี่ย 4.30 คะแนน โดยด้านที่ได้คะแนนสูงสุดคือความแข็งแรงของอุปกรณ์และเซนเซอร์ 4.4 คะแนน แสดงถึงความเชื่อมั่นในคุณภาพอุปกรณ์ ส่วนด้านอื่น ๆ เช่น หน้าจอแสดงผล ความปลอดภัย และความสะดวกในการติดตั้ง ก็อยู่ในระดับ ดี แสดงว่าระบบมีการออกแบบที่เหมาะสมและสามารถใช้งานได้จริงในห้องเรียน

ตารางที่ 4.4 ผลการประเมินด้านการใช้งานระบบโดยผู้ใช

หัวข้อประเมิน	คะแนนเฉลี่ย (เต็ม 5)	ระดับความคิดเห็น
ความง่ายในการใช้งาน	4.5	ดีมาก
ความชัดเจนของข้อมูลบนหน้าจอ	4.4	ดีมาก
ความรวดเร็วในการตอบสนองของระบบ	4.3	ดี
การเชื่อมต่อ API และความเสถียร	4.2	ดี
ความปลอดภัยขณะใช้งาน	4.4	ดีมาก
รวมเฉลี่ย	4.36	ดีมาก

จากตารางที่ 4.4 พบว่า กลุ่มตัวอย่างประเมินการใช้งานระบบอยู่ในระดับ ดีมาก ด้วยคะแนนเฉลี่ยรวม 4.36 โดยด้านที่ได้คะแนนสูงสุด ได้แก่ ความง่ายในการใช้งาน ความชัดเจนของข้อมูลและ ความปลอดภัย แสดงให้เห็นว่าระบบใช้งานง่ายและน่าเชื่อถือ ส่วนด้านอื่น ๆ เช่น ความเร็วของระบบ และ ความเสถียรของการเชื่อมต่อ ก็อยู่ในระดับ ดี สะท้อนถึงความพร้อมในการใช้งานจริงในห้องเรียน

บทที่ 5

สรุปผล ปัญหา และข้อเสนอแนะ

5.1 สรุปผลของโครงการ

โครงการเรื่องระบบตรวจวัดอุณหภูมิและความชื้นด้วยเซ็นเซอร์ S09 ผ่านการเชื่อมต่อภายในเครือข่าย เป็นโครงการที่มุ่งเน้นการพัฒนาระบบ IoT สำหรับการติดตามสภาพแวดล้อมภายในอาคาร โดยเฉพาะในพื้นที่ที่ต้องการควบคุมอุณหภูมิและความชื้นอย่างต่อเนื่อง เช่น ห้องเรียน ห้องเก็บอุปกรณ์ หรือห้อง Server ซึ่งโครงการสามารถสรุปผลที่สำคัญได้ดังนี้

1. สามารถเชื่อมต่อเซ็นเซอร์ S09 เข้ากับแอปพลิเคชัน Smart Life ได้อย่างสมบูรณ์ ทำให้ผู้ใช้สามารถดูค่าต่าง ๆ แบบ Real-Time จากสมาร์ตโฟน
2. พัฒนาโปรแกรมดึงข้อมูลจากเซ็นเซอร์ S09 ผ่านการเชื่อมต่อในเครือข่ายภายใน (LAN) โดยใช้ไลบรารี TinyTuya ในภาษา Python ซึ่งสามารถอ่านค่าอุณหภูมิและความชื้นได้อย่างแม่นยำและต่อเนื่อง
3. ระบบมีความแม่นยำสูง โดยค่าที่ได้มีค่าคลาดเคลื่อนต่ำ (± 0.4 °C) และมีระยะเวลาตอบสนองที่รวดเร็ว (เฉลี่ย 0.3 – 0.5 วินาที/ค่าขอ)
4. มีความเสถียรในการทำงานอย่างต่อเนื่อง โดยสามารถออนไลน์ได้ตลอด 48 ชั่วโมงโดยไม่เกิดการขาดการเชื่อมต่อ
5. สามารถแสดงผลข้อมูลที่วัดได้ในรูปแบบกราฟ และวิเคราะห์แนวโน้มของสภาพอากาศในพื้นที่เป้าหมายได้เบื้องต้น
6. โครงการนี้มีศักยภาพในการนำไปต่อยอดสู่การควบคุมสิ่งแวดล้อมอัตโนมัติ เช่น การเปิด – ปิดพัดลมหรือเครื่องปรับอากาศอัตโนมัติในอนาคต

5.2 ปัญหาและอุปสรรคที่พบ

ตลอดการพัฒนาโครงการ พบปัญหาหลายประการ ซึ่งได้มีการวิเคราะห์สาเหตุและเสนอแนวทางการแก้ไขไว้ ดังตารางต่อไปนี้

ตารางที่ 5.1 ปัญหาที่พบและแนวทางการแก้ไข

ปัญหาที่พบ	สาเหตุที่เป็นไปได้	แนวทางการแก้ไข
ไม่พบอุปกรณ์ในแอป Smart Life	เครือข่าย Wi-Fi ไม่รองรับ 2.4GHz (อุปกรณ์ไม่รองรับ 5GHz)	เปลี่ยนมาใช้ Hotspot หรือเราเตอร์ที่ปล่อยสัญญาณ 2.4GHz เท่านั้น
ไม่สามารถดึง Local Key ได้	ยังไม่มี Device ID หรือไม่ใช้ API Explorer ถูกวิธี	ใช้ Device ID ไปค้นหา Local Key ผ่าน API Explorer บน Tuya Platform โดยใช้คำสั่ง Query Device Details
ไม่พบ IP ของเซ็นเซอร์	IP ถูกเปลี่ยนทุกครั้งเมื่อเปิดเครื่องใหม่ (Dynamic IP)	ใช้โปรแกรม Advanced IP Scanner เพื่อค้นหา IP และตั้งค่า Static IP ในเราเตอร์เพื่อป้องกันการเปลี่ยนแปลง
เซ็นเซอร์ไม่ตอบสนอง คำสั่ง	เซ็นเซอร์ไม่ได้เชื่อมต่อ Wi-Fi หรือ ล้มเหลวในการจ่ายไฟ	ตรวจสอบการเชื่อมต่อ Wi-Fi และแหล่งจ่ายไฟให้เสถียร
ดึงข้อมูลใน Python ไม่ได้	LOCAL_KEY หรือ IP ผิด, อุปกรณ์ไม่อยู่ในเครือข่ายเดียวกัน	ตรวจสอบว่า Python รันในเครือข่ายเดียวกับเซ็นเซอร์ และใช้ Local Key กับ IP ที่ถูกต้อง
กราฟไม่แสดงข้อมูล	ไม่มีข้อมูลส่งเข้า API หรือข้อมูลเก่าเกินไป	ตรวจสอบให้มีการดึงข้อมูลใหม่ทุก 10 วินาที และแสดงผลเฉพาะข้อมูลล่าสุด 10 ค่า

จากตารางที่ 5.1 แสดงปัญหาที่เกิดขึ้นระหว่างการติดตั้งและทดสอบระบบ พร้อมสาเหตุที่เป็นไปได้และแนวทางการแก้ไข เช่น ปัญหาเกี่ยวกับ Wi-Fi, IP Address และการตั้งค่าการเชื่อมต่ออุปกรณ์ โดยแนวทางที่นำมาใช้สามารถแก้ไขปัญหาและทำให้ระบบทำงานได้อย่างมีประสิทธิภาพ

5.3 ข้อเสนอแนะในการพัฒนาโครงการต่อไป

จากประสบการณ์และบทเรียนที่ได้รับ ผู้จัดทำมีข้อเสนอแนะเพื่อพัฒนาระบบในอนาคต ดังต่อไปนี้

5.3.1 เพิ่มระบบแจ้งเตือนอัตโนมัติ

พัฒนาให้ระบบสามารถแจ้งเตือนผู้ใช้งานเมื่ออุณหภูมิหรือความชื้นเกินค่าที่กำหนด เช่น แจ้งเตือนผ่าน Telegram หรือ Email

5.3.2 เชื่อมต่อกับระบบควบคุมอัตโนมัติ

นำข้อมูลจากเซ็นเซอร์ไปควบคุมอุปกรณ์อื่น เช่น เปิด – ปิดพัดลมหรือเครื่องปรับอากาศอัตโนมัติตามเงื่อนไขที่กำหนด

5.3.3 บันทึกข้อมูลลงฐานข้อมูล

พัฒนาระบบให้สามารถจัดเก็บข้อมูลลงฐานข้อมูล เช่น Time Series Database หรือ Firebase เพื่อวิเคราะห์ย้อนหลัง

5.3.4 พัฒนา Web Dashboard หรือ Mobile App

สร้าง Dashboard แสดงผลแบบโต้ตอบ หรือแอปพลิเคชันสำหรับแสดงกราฟและข้อมูลแบบ Real-Time

5.3.5 รองรับหลายเซ็นเซอร์พร้อมกัน

ปรับปรุงระบบให้สามารถเชื่อมต่อกับเซ็นเซอร์หลายตัวในเวลาเดียวกัน เช่น หลายห้องเรียนหรือหลายโซนในอาคาร

5.3.6 วิเคราะห์ข้อมูลเชิงสถิติ

พัฒนาโมดูลวิเคราะห์ข้อมูล เช่น หาค่าเฉลี่ย รายวัน รายสัปดาห์ หรือแนวโน้ม เพื่อการตัดสินใจที่แม่นยำ

5.4 บทสรุปโดยรวม

โครงการนี้ประสบความสำเร็จในการออกแบบและพัฒนาระบบตรวจวัดอุณหภูมิและความชื้นแบบ IoT โดยใช้เซ็นเซอร์ S09 ร่วมกับแอป Smart Life และเชื่อมต่อภายในเครือข่าย (LAN) โดยไม่จำเป็นต้องใช้ Tuya Cloud API โดยตรง ระบบสามารถทำงานได้จริง มีประสิทธิภาพสูง ทั้งในด้านความแม่นยำ ความรวดเร็ว และความเสถียร แม้จะมีข้อจำกัดบางประการ แต่ระบบสามารถนำไปปรับปรุงและต่อยอดได้อีกมากในอนาคต ถือเป็นโครงการที่เหมาะสมกับยุค Smart Home และ Smart Classroom อย่างแท้จริง

บรรณานุกรม

- สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ. (2565). อินเทอร์เน็ตของสรรพสิ่ง (IoT): แนวคิดและการประยุกต์ใช้งาน. สืบค้นเมื่อ 10 กรกฎาคม 2568, จาก <https://www.nstda.or.th>
- Advanced IP Scanner. (2024). **Network Scanner Tool**. Retrieved August 20, 2025, from <https://www.advanced-ip-scanner.com>
- Flask. (2024). **The Python microframework for building web applications**. Retrieved August 20, 2025, from <https://flask.palletsprojects.com>
- GitHub. (2024). **tinytuya: Python API for local Tuya devices**. Retrieved August 20, 2025, from <https://github.com/jasonacox/tinytuya>
- Tuya Developer Platform. (2024). **API Explorer – Query Device Details in Bulk**. Retrieved August 20, 2025, from <https://iot.tuya.com>

ภาคผนวก ก
ตัวอย่างโค้ด

ตัวอย่างโค้ดบางส่วนของระบบตรวจวัดอุณหภูมิและความชื้นภายในห้องเรียน

```

app.py > index
1 from flask import Flask, jsonify, render_template
2 import tinytuya
3 import datetime
4 import logging
5
6 # ข้อมูลอุปกรณ์ Tuya
7 DEVICE_ID = "a363fd4a6a60153af3bhdg"
8 DEVICE_IP = "192.168.0.104"
9 LOCAL_KEY = "@tU9ccyMOB$+2H02"
10 PROTOCOL_VERSION = 3.4
11
12 app = Flask(__name__)
13 logging.basicConfig(level=logging.INFO)
14
15 # เชื่อมต่อกับอุปกรณ์ Tuya
16 device = tinytuya.OutletDevice(DEVICE_ID, DEVICE_IP, LOCAL_KEY)
17 device.set_version(PROTOCOL_VERSION)
18
19 @app.route("/")
20 def index():
21     # ส่งไฟล์ HTML dashboard.html จากไฟล์ templates
22     return render_template("dashboard.html")
23
24 @app.route("/api/temperature")
25 def get_temperature():
26     try:
27         logging.info("✱ Reading data from device...")
28         status = device.status()
29         logging.info(f"Status from device: {status}")
30
31         if "dps" not in status:
32             raise ValueError("ไม่มีข้อมูล dps")
33
34         dps = status["dps"]
35
36         # ตรวจสอบและแปลงค่าที่ได้ให้เป็น float อย่างปลอดภัย
37         temp_raw = dps.get("101")
38
39         # ตรวจสอบและแปลงค่าที่ได้ให้เป็น float อย่างปลอดภัย
40         hum = dps.get("102")
41
42         if temp_raw is None or hum is None:
43             raise ValueError("ไม่พบข้อมูล dps 101 หรือ 102")
44
45         # แปลงเป็น float และแปลงอุณหภูมิเป็น °C
46         temperature = float(temp_raw) / 10
47         humidity = float(hum)
48
49         return jsonify({
50             "temperature": temperature,
51             "humidity": humidity,
52             "timestamp": datetime.datetime.now().strftime("%Y-%m-%d %H:%M:%S")
53         })
54     except Exception as e:
55         logging.error(f"✖ Error reading device: {e}")
56         return jsonify({
57             "error": "ไม่สามารถอ่านข้อมูลจากอุปกรณ์ได้",
58             "detail": str(e)
59         }), 500
60
61 @app.route("/api/status")
62 def api_status():
63     try:
64         status = device.status()
65         return jsonify(status)
66     except Exception as e:
67         return jsonify({"error": str(e)}), 500
68
69 if __name__ == "__main__":
70     app.run(host="0.0.0.0", port=5000, debug=True)

```

ภาคผนวก ข

แบบสอบถาม

ตอนที่ 1: ข้อมูลทั่วไปของผู้ตอบแบบสอบถาม

โปรดทำเครื่องหมาย ✓ หน้าคำตอบที่ตรงกับท่าน

1. เพศ

☐ ชาย

☐ หญิง

☐ ไม่ระบุ

2. อายุ

☐ ต่ำกว่า 20 ปี

☐ 21-30 ปี

☐ 31-40 ปี

☐ 41-50 ปี

☐ มากกว่า 50 ปี

3. อาชีพ

☐ นักเรียน / นักศึกษา

☐ ข้าราชการ / พนักงานรัฐวิสาหกิจ

☐ พนักงานบริษัท

☐ เจ้าของกิจการ / ธุรกิจส่วนตัว

☐ อื่น ๆ (โปรดระบุ): _____

ตอนที่ 2: ความพึงพอใจในการใช้งานระบบ

โปรดให้คะแนนความพึงพอใจตามระดับ 5 คะแนน ดังนี้

5 = มากที่สุด

4 = มาก

3 = ปานกลาง

2 = น้อย

1 = น้อยที่สุด

รายการประเมิน	1	2	3	4	5
1. ความสะดวกในการใช้งานระบบผ่านแอปพลิเคชัน Smart Life	☐	☐	☐	☐	☐
2. ความถูกต้องของข้อมูลอุณหภูมิและความชื้นที่แสดงผล	☐	☐	☐	☐	☐
3. ความรวดเร็วในการแสดงผลข้อมูลแบบ Real-Time	☐	☐	☐	☐	☐
4. ความสวยงามและเป็นมิตรต่อผู้ใช้งานของอินเทอร์เฟซ	☐	☐	☐	☐	☐
5. ความเสถียรของระบบระหว่างใช้งาน	☐	☐	☐	☐	☐
6. ความเข้าใจง่ายในการติดตั้งและเชื่อมต่อเซ็นเซอร์	☐	☐	☐	☐	☐
7. ความพึงพอใจโดยรวมของระบบที่พัฒนา	☐	☐	☐	☐	☐

ตอนที่ 3: ข้อเสนอแนะเพิ่มเติม (ถ้ามี)

.....

.....

.....

ภาคผนวก ค

การใช้ระบบตรวจวัดอุณหภูมิและความชื้นภายในห้องเรียน

คู่มือการใช้ระบบตรวจวัดอุณหภูมิและความชื้นภายในห้องเรียน

1. บทนำ

เพื่อให้ผู้ใช้งานสามารถติดตั้งระบบและใช้งานระบบตรวจวัดอุณหภูมิและความชื้นภายในห้องเรียนได้อย่างถูกต้องและมีประสิทธิภาพ

2. การเข้าถึงระบบ

ผู้ใช้งานสามารถเข้าสู่ระบบได้ง่ายผ่านอุปกรณ์ที่เชื่อมต่อเครือข่ายเดียวกัน โดยมีขั้นตอนดังนี้

1. เปิดเว็บเบราว์เซอร์ (เช่น Google Chrome, Mozilla Firefox, Microsoft Edge)
2. ป้อน URL ของระบบลงในช่อง Address Bar `http://192.168.1.100`
3. หน้า Dashboard จะแสดงข้อมูลอุณหภูมิแบบ Real-Time ทันที

3. รายละเอียดหน้าจอแสดงผล

3.1 ส่วนหัว (Header)

- แสดงชื่อระบบ “Temperature Dashboard” พร้อมไอคอนเทอร์โมมิเตอร์
- แสดงคำอธิบาย “ระบบติดตามอุณหภูมิห้องเรียน ปวส.2 คอมพิวเตอร์”

3.2 การ์ดแสดงอุณหภูมิหลัก

- แสดงชื่อห้องที่ติดตั้งเซ็นเซอร์
- แสดงสถานะการเชื่อมต่อ API:
 - จุดสีเขียว: เชื่อมต่อเซ็นเซอร์ได้ปกติ
 - จุดสีแดง: ไม่สามารถเชื่อมต่อกับเซ็นเซอร์
- มีวงกลมเกจแสดงค่าอุณหภูมิแบบกราฟิก (°C)

3.3 การ์ดสถิติ (Statistics)

- อุณหภูมิสูงสุดที่วัดได้ (°C)
- อุณหภูมิต่ำสุดที่วัดได้ (°C)
- อุณหภูมิเฉลี่ย (°C)
- เวลาที่มีการอัปเดตข้อมูลล่าสุด (เวลาไทย)

3.4 กราฟอุณหภูมิแบบ Real-Time

- แสดงกราฟเส้นแสดงแนวโน้มอุณหภูมีย้อนหลังประมาณ 3 นาที
- อัปเดตข้อมูลใหม่ทุก 10 วินาที
- ใช้ไลบรารี Chart.js ในการแสดงผลกราฟ

3.5 สถานะและการอัปเดตล่าสุด

- แสดงข้อความ เช่น “อัปเดตล่าสุด: 06/08/2025 15:30:12”
- แสดงข้อความสถานะ เช่น:
 - “เชื่อมต่อสำเร็จ”
 - “ไม่สามารถเชื่อมต่อได้”

4. วิธีการใช้งานระบบ

- 4.1 เข้าสู่หน้าเว็บของระบบตาม URL ที่กำหนด
- 4.2 ระบบจะเริ่มเชื่อมต่อกับเซ็นเซอร์ S09 โดยอัตโนมัติ
- 4.3 ข้อมูลอุณหภูมิจะอัปเดตอัตโนมัติทุก 10 วินาที
- 4.4 ผู้ใช้งานสามารถติดตามค่าปัจจุบัน สถิติ และแนวโน้มจาก Dashboard ได้
- 4.5 หากระบบแสดงสถานะ “ไม่สามารถเชื่อมต่อได้” ให้ตรวจสอบ
- 4.6 สถานะการเชื่อมต่อ Wi-Fi ของเซ็นเซอร์
- 4.7 IP Address ของเซ็นเซอร์อาจเปลี่ยน ควรตั้งค่าเป็น Static IP เพื่อป้องกันปัญหานี้

5. ข้อควรระวังและคำแนะนำ

- ควรใช้งานระบบในเครือข่ายที่มีความเสถียร เช่น Wi-Fi หรือ LAN ที่ไม่มีปัญหาการหลุดสัญญาณบ่อย
- หากพบว่าข้อมูลไม่อัปเดต
 - ตรวจสอบว่าเซ็นเซอร์ยังมีไฟเลี้ยงอยู่หรือไม่
 - ลองรีเฟรชหน้าเว็บ หรือรีสตาร์ทโปรแกรมฝั่งเซิร์ฟเวอร์
- หากแสดงค่า “0.0 °C” หรือค่าผิดปกติ:
 - ตรวจสอบสายไฟและสถานะการเชื่อมต่อเครือข่าย
- ระบบในเวอร์ชันปัจจุบันแสดงเฉพาะ อุณหภูมิ ยังไม่รวมค่าความชื้น
- ระบบยังไม่รองรับการบันทึกข้อมูลย้อนหลัง
 - หากต้องการดูประวัติ ควรพัฒนาเพิ่มเติมโดยเชื่อมต่อกับฐานข้อมูล เช่น SQLite หรือ Firebase

6. หมายเหตุเพิ่มเติม

- หากมีการรีสตาร์ทเซ็นเซอร์ S09 แล้วไม่สามารถดึงข้อมูลได้ อาจเกิดจาก IP Address เปลี่ยนแปลงโดยอัตโนมัติ
 - แนะนำให้เข้าไปตั้งค่า Static IP ผ่านหน้าจัดการของเราเตอร์ เพื่อหลีกเลี่ยงปัญหานี้
- การติดตั้งระบบครั้งแรกควรอยู่ภายใต้การดูแลของผู้ที่มีความรู้ทางเทคนิค หรืออาจารย์ที่ปรึกษาโครงการ
- หากต้องการเพิ่มฟังก์ชันเพิ่มเติม เช่น ระบบแจ้งเตือนหรือการเก็บข้อมูลย้อนหลัง สามารถต่อยอดได้ง่ายผ่านการเขียนโปรแกรมในฝั่งเซิร์ฟเวอร์

ประวัติผู้จัดทำโครงการ



ชื่อ - สกุล
ระดับการศึกษา

ที่อยู่
เบอร์โทร

นางสาววิภาวี พันธุ์เสถียร
ประกาศนียบัตรวิชาชีพชั้นสูง ระดับชั้นปีที่ 2
สาขาเทคโนโลยีคอมพิวเตอร์ ปีการศึกษา 2568
8/1 หมู่ 5 ตำบลห้วยคต อำเภอห้วยคต จังหวัดอุทัยธานี
064-959-3187



ชื่อ - สกุล
ระดับการศึกษา

ที่อยู่
เบอร์โทร

นางสาวปณิดา กองแสน
ประกาศนียบัตรวิชาชีพชั้นสูง ระดับชั้นปีที่ 2
สาขาเทคโนโลยีคอมพิวเตอร์ ปีการศึกษา 2568
95 หมู่ 7 ตำบลทุ่งทอง อำเภอรายทองวัฒนา จังหวัดกำแพงเพชร
064-052-7966