



จตุรรายงานสภาพอากาศ

WEATHER REPORTING SPOT

คงศักดิ์ ทองกำนบัว

KANIT THONGKANBUA

คณิศร์ ทองกำนบัว

KONGSAK THONGKANBUA

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งตามหลักปริญญาวិชากรรมศาสตร์บัณฑิต

สาขาวิชาวิศวกรรมสารสนเทศ ภาควิชาวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

วิทยาเขตชุมพรเขตอุดมศักดิ์ จังหวัดชุมพร

ปีการศึกษา 2563

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รับที่...../.....

งานทะเบียนและประมวลผล

ฉบับที่.....

ปริญญาโทปีการศึกษา 2563

สาขาวิชาวิศวกรรมสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

วิทยาเขตชุมพรเขตรอุดมศักดิ์ จังหวัดชุมพร

เรื่อง จุติรายงานสภาพอากาศ

WEATHER REPORTING SPOT

ผู้จัดทำ

1. นายคณิศร์ ทองกำนั้ว รหัสนักศึกษา 60515008
2. นายคณิศร์ ทองกำนั้ว รหัสนักศึกษา 60515009

พิศากร สิทธิวัจน์

.....อาจารย์ที่ปรึกษา

(อาจารย์พิศากร สิทธิวัจน์)

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2020

DEPARTMENT OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

PRINCE OF CHUMPHON CAMPUS

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาควิชาวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
วิทยาเขตชุมพรเขตรอุดมศักดิ์ จังหวัดชุมพร  
ใบรับรองปริญญาานิพนธ์

ชื่อปริญญาานิพนธ์ จุฑารายงานสภาพอากาศ  
Project Title WEATHER REPORTING SPOT  
ชื่อนักศึกษา นายคณิศร์ ทองก้านบัว รหัสนักศึกษา 60515008  
นายคณิศร์ ทองก้านบัว รหัสนักศึกษา 60515009  
ปริญญา วิศวกรรมศาสตรบัณฑิต  
สาขาวิชา วิศวกรรมสารสนเทศ  
อาจารย์ที่ปรึกษา อาจารย์พิศากร สิทธิวัจน์

คณะกรรมการสอบปริญญาานิพนธ์			ลายมือชื่อ
ผศ.ดร.รัฐพงษ์	สุวลักษณ์	ประธานกรรมการ	
รศ.ดร.บุญยวีร์	จามจรীগุสกาญจน์	กรรมการ	
อ.ดร.รัตติกร	สมบัติแก้ว	กรรมการ	
อ.พิศากร	สิทธิวัจน์	กรรมการ	

วัน/เดือน/ปี ที่สอบ 7-8 มิถุนายน พ.ศ. 2564 เวลา 09.00 น. - 17.30 น.  
สถานที่สอบ ออนไลน์ด้วยโปรแกรม Microsoft Team

ภาควิชาวิศวกรรมศาสตร์ รับรองแล้ว



(ผู้ช่วยศาสตราจารย์ ดร.ปราโมทย์ กุศล)  
หัวหน้าภาควิชาวิศวกรรมศาสตร์  
วันที่ 15 มิถุนายน พ.ศ. 2564

ชื่อปริญญาบัตร	จตุรรายงานสภาพอากาศ
นักศึกษา	นายคณศักดิ์ ทองก้านบัว รหัสนักศึกษา 60515008
	นายคณิศร์ ทองก้านบัว รหัสนักศึกษา 60515009
อาจารย์ที่ปรึกษา	อาจารย์พิศากร ลีทธิวิจน์
หลักสูตร	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2563

### บทคัดย่อ

ปริญญาบัตรฉบับนี้นำเสนอจตุรรายงานสภาพอากาศที่สามารถรายงาน ความเข้มยอีย อุณหภูมิ ความเข้มแสง ปริมาณน้ำ ความเร็วลม และถ่ายภาพ โดยทำแสดงค่าเป็นเรียลไทม์ประกอบไปด้วย หน้าเว็บและโมบายแอปพลิเคชันสำหรับแสดงผลรายงานสภาพอากาศเป็นตัวเลขและมีฟังก์ชันรายงานสภาพอากาศย้อนหลังเจ็ดวันเป็นกราฟ พร้อมดูภาพถ่ายทุกหนึ่งชั่วโมงเพื่อให้ผู้ที่จุดประสงค์ที่จะไปสถานที่ที่มีระบบนี้ ได้ทราบสภาพอากาศและได้เห็นภาพถ่ายในปัจจุบันเพื่อช่วยในการตัดสินใจของผู้ใช้งาน จากการทดลองเบื้องต้นพบว่า จตุรรายงานสภาพอากาศสามารถส่งค่าของแต่ละเซ็นเซอร์ลงดาต้าเบสได้ เว็บและโมบายแอปพลิเคชันสามารถนำค่าที่เก็บไว้ในดาต้าเบสมาแสดงบนเว็บและระบบโมบายแอปพลิเคชันได้

**คำสำคัญ** ความเข้มยอีย อุณหภูมิ ความเข้มแสง ปริมาณน้ำ ความเร็วลม ถ่ายภาพ

**Project Title** Weather Reporting Spot

**Students** Mr. Kongsak Thongkanbua ID 60515008  
Mr. Kanis Thongkanbua ID 60515009

**Advisor** Dr. Phisakorn Sittiwat

**Degree** Bachelor of Engineering

**Program in** Information Engineering

**Academic Year** 2020

## ABSTRACT

This thesis presents a report point of weather that can be reported. temperature, light intensity, water content, wind speed and aerial photography. by showing the value in real time consisting of Web pages and mobile applications for displaying numerical weather reports. and has a function to report the past seven days of weather as a graph. along with viewing photos every hour for those who intend to go places with this system. get to know the weather and see current photos to help users make decisions. from the preliminary experiments, it was found that The weather report point can send the values of each sensor into the database. Web and mobile applications can bring the values stored in the database to display on web and mobile applications.

**Keywords:** intensity, temperature, light intensity, water content, wind speed, photography

## กิตติกรรมประกาศ

โครงการจตุรายนสภาพอากาศและปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงลงได้อย่างสมบูรณ์ นั้นเป็นเพราะผู้จัดทำได้รับความช่วยเหลือ ได้รับการสนับสนุน เอื้อเฟื้อเครื่องมือ และได้รับคำปรึกษา ทั้งทางตรงและทางอ้อมจากบุคคลหลายท่าน ซึ่งผู้จัดทำรู้สึกซาบซึ้งอยากขอขอบพระคุณ ขอขอบคุณ และขอขอบใจ จึงขอแสดงออกถึงสิ่งที่รู้สึกไว้ ณ ที่แห่งนี้

ขอขอบพระคุณครอบครัวและบุพการีผู้ซึ่งเป็นกำลังใจ ส่งความห่วงใย สนับสนุนค่าใช้จ่าย ด้วยดีเสมอมา บุคคลเหล่านี้เป็นกลุ่มบุคคลที่ผู้จัดทำอยากกล่าวถึงเป็นลำดับแรก มีคำที่อยากบอก มากมายแต่ขอกลับออกมาเพียงหนึ่งคำที่มีความหมายอัดแน่นนั่นคือ ขอขอบพระคุณครับ

ขอขอบคุณพี่ๆ เพื่อนๆ ที่คอยช่วยเหลือกันในการทำโครงการชิ้นนี้จนสำเร็จได้อย่างสมบูรณ์ ขอขอบคุณที่สู้มาด้วยกัน อดทนมาด้วยกัน

ขอขอบคุณคูโปรเจกต์ ที่ร่วมมือกันด้วยดีมาโดยตลอด ผ่านความลำบาก ผ่านความไม่รู้และ ได้ลองผิดลองถูกไปด้วยกัน ต้องอดทนกับงานไปด้วยกัน ปรึกษากันจนจนที่สุดก็เรียนรู้ทั้งทฤษฎีและ ปฏิบัติไปด้วยกัน ผลลัพธ์ที่ได้มาคือโครงการนี้ก็สำเร็จลุล่วง ขอขอบคุณคูโปรเจกต์

และสุดท้ายขอขอบพระคุณ อาจารย์พิศากร สิทธิวิจน์ ผู้เป็นอาจารย์ที่ปรึกษาโครงการซึ่งได้ ให้คำแนะนำต่างๆ รวมทั้งเอื้อเฟื้อเครื่องมือเครื่องใช้ในการทำโครงการ และติดตามถามไถ่เกี่ยวกับ งานอย่างสม่ำเสมอ ผู้จัดทำรู้สึกซาบซึ้งในความเมตตาของท่านจึงขอขอบพระคุณเป็นอย่างสูง

คุณค่าและประโยชน์อันพึงมีจากปริญญานิพนธ์ฉบับนี้ ผู้จัดทำขอมอบแก่ผู้มีพระคุณทุกท่าน

คงศักดิ์ ทองก้านบัว

คณิศร์ ทองก้านบัว

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	IX
สารบัญรูป	X
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญ	1
1.2 วัตถุประสงค์ของโครงการ	2
1.3 ขอบเขตของโครงการ	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	2
1.5 แผนการดำเนินงาน	2
1.6 โครงสร้างของปริญญาานิพนธ์	5
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	6
2.1 ไมโครคอนโทรลเลอร์อีเอสพี 8266 (Microcontroller ESP826)	6
2.1.1 วิธีการติดตั้ง ESP8266 ลงในโปรแกรมอาร์ดูอีนโอไอดีอี (Arduino IDE)	7
2.1.2 การเขียนโค้ดในโปรแกรม Arduino IDE	11
2.1.3 การเชื่อมต่อและการสแกนหาชื่อ Wi-Fi	13
2.2 เซนเซอร์วัดความเข้มแสง BH1750	15
2.2.1 ขาของเซนเซอร์วัดความเข้มแสง BH1750	15
2.2.2 คุณสมบัติของของเซนเซอร์วัดความเข้มแสง BH1750	16
2.2.3 เซนเซอร์วัดความเข้มแสง BH1750	16
2.3 เซนเซอร์วัดความเข้มรังสีอัลตราไวโอเล็ต GUVA S12SD	22
2.3.1 คุณสมบัติของเซนเซอร์ยูวี GUVA S12SD	22

## สารบัญ (ต่อ)

	หน้า
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	6
2.3 เซนเซอร์วัดความเข้มรังสีอัลตราไวโอเล็ต GUVA S12SD	24
2.3.2 วิธีการเชื่อมต่อเซนเซอร์ยี่ห้อ GUVA S12SD ให้เข้ากับ ESP8266	24
2.3.3 ขาของเซนเซอร์ยี่ห้อ GUVA S12SD	24
2.3.4 ตัวอย่าง Code	24
2.4 เซ็นเซอร์วัดลม (Wind speed sensor)	25
2.4.1 ข้อมูลทางเทคนิคเซ็นเซอร์วัดลม	25
2.4.2 คุณสมบัติเซ็นเซอร์วัดลม	25
2.4.3 ภาพวาดมิติโครงสร้างเซ็นเซอร์วัดลม	25
2.4.4 การติดตั้งเซ็นเซอร์วัดลม	26
2.4.5 ขาของเซนเซอร์วัดความเร็วลม	27
2.4.6 การต่อใช้งานของเซนเซอร์วัดความเร็วลม	27
2.4.7 ตัวอย่างโค้ด	28
2.5 เซนเซอร์วัดปริมาณน้ำฝน (Raindrop Sensor)	29
2.5.1 คุณสมบัติของของเซนเซอร์วัดปริมาณน้ำฝน	29
2.5.2 หลักการทำงานของเซนเซอร์วัดปริมาณน้ำฝน	29
2.5.3 การต่อใช้งานของเซนเซอร์วัดปริมาณน้ำฝน	30
2.5.4 ตัวอย่างโค้ด	31
2.6 เซนเซอร์วัดอุณหภูมิและความชื้น (DHT11 Temperature and Humidity Sensor Module)	31
2.6.1 ข้อมูลทางเทคนิคของเซนเซอร์ DHT11	32
2.6.2 การต่อใช้งานเซนเซอร์วัดอุณหภูมิ DHT11เข้ากับ ESP8266	32
2.6.3 ตัวอย่างโค้ด	33
2.7 โมดูลกล้อง ESP32 (ESP32-CAM Module)	33
2.7.1 คุณสมบัติของกล้อง ESP32	34
2.7.2 ข้อมูลทางเทคนิคของกล้อง ESP32	34
2.7.3 โมดูลกล้อง ESP32 เหมาะสำหรับการใช้งานไอโอทีด้านใดบ้าง	34

## สารบัญ (ต่อ)

	หน้า
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	6
2.7 โมดูลกล้อง ESP32 (ESP32-CAM Module)	33
2.7.4 การเชื่อมต่อระหว่างกล้อง ESP32 กับ FT232RLUSB	35
2.7.5 การติดตั้งไดรเวอร์ของกล้อง ESP32 และ FT232RLUSB	36
2.8 ภาษา HTML	40
2.8.1 ประวัติของภาษา HTML	40
2.8.2 คุณสมบัติเด่นของภาษา HTML	40
2.8.3 ส่วนประกอบที่สำคัญของภาษา HTML	40
2.8.4 ส่วนประกอบของเอกสาร HTML (HTML Document)	42
2.8.5 การสร้างเอกสาร HTML	42
2.9 ภาษา CSS	45
2.9.1 CSS คืออะไร	45
2.9.2 รูปแบบการใช้คำสั่ง CSS	46
2.9.3 ประโยชน์ของ CSS	48
2.10 ภาษา JavaScript	49
2.10.1 JavaScript ทำอะไรได้บ้าง	50
2.10.2 ความสามารถของ JavaScript	50
2.10.3 ข้อดีและข้อเสียของ JavaScript	50
2.11 โปรแกรม MySQL	51
2.11.1 ความสามารถและการทำงานของโปรแกรม MySQL	51
2.11.2 ข้อควรระวังที่เกี่ยวข้องกับความปลอดภัยของโปรแกรม MySQL	51
บทที่ 3 หลักการและการออกแบบ	53
3.1 ภาพรวมของระบบ (Block Diagram)	53
3.2 การออกแบบโครงสร้างจตุรงานสถาพอากาศ	54
3.3 การออกแบบเชื่อมต่อวงจรเซนเซอร์	54
3.4 การประกอบเสาจตุรงานสถาพอากาศ	55

## สารบัญ (ต่อ)

	หน้า
บทที่ 3 หลักการและการออกแบบ	53
3.5 การติดตั้งจุดรายงานสภาพอากาศในสถานที่จริง	57
3.5.1 การติดตั้งจุดรายงานสภาพอากาศ ตัวที่ 1 (หน้าหอพักของผู้จัดทำ)	57
3.5.2 การติดตั้งจุดรายงานสภาพอากาศ ตัวที่ 2 (ด้านข้างหอพักของอาจารย์)	59
3.5.3 การติดตั้งจุดรายงานสภาพอากาศ ตัวที่ 1 (ณ หากพระจอม)	61
3.6 การออกแบบและสร้างฐานข้อมูล	62
3.6.1 ฐานข้อมูลสำหรับจุดรายงานสภาพอากาศ ตัวที่ 1 และ 2	62
3.6.2 ฐานข้อมูลเพื่อใช้เก็บภาพถ่ายจากกล้อง ESP32	64
3.7 แผนผังการทำงานของโปรแกรมและซอร์สโค้ด (Flowchart and Source Code)	67
3.7.1 แผนผังการทำงานของโปรแกรม	67
3.7.2 ซอร์สโค้ดสำหรับเซนเซอร์ของจุดรายงานสภาพอากาศ	68
3.7.3 ซอร์สโค้ดสำหรับกล้อง ESP32 ของจุดรายงานสภาพอากาศ	71
3.8 การสร้างเว็บไซต์และโมบายแอปพลิเคชัน	74
3.8.1 แผนผังเว็บไซต์และโมบายแอปพลิเคชัน	74
3.8.2 เว็บไซต์สำหรับรายงานสภาพอากาศ	75
3.8.3 โมบายแอปพลิเคชันสำหรับรายงานสภาพอากาศ	83
บทที่ 4 ผลการทดลอง	94
4.1 การทดสอบเซนเซอร์วัดค่าความชื้นของรังสีอัลตราไวโอเล็ต GUVA S12SD	94
4.1.1 การทดสอบการทำงานของเซนเซอร์วัดค่าความชื้นรังสีอัลตราไวโอเล็ต	94
4.1.2 การทดสอบนำค่าเซนเซอร์ที่วัดค่าความชื้นของรังสีอัลตราไวโอเล็ตเข้า ฐานข้อมูล	96
4.2 การทดสอบเซนเซอร์วัดความเข้มของแสง BH1750	99
4.2.1 การทดสอบการทำงานของเซนเซอร์	99
4.2.2 การทดสอบการนำค่าเซนเซอร์ที่วัดค่าความเข้มแสง BH1750เข้า ฐานข้อมูล	101
4.3 การทดสอบเซนเซอร์วัดความเร็วลม	103
4.3.1 การทดสอบการทำงานของเซนเซอร์วัดความเร็วลม	103
4.3.2 การทดสอบการนำค่าเซนเซอร์วัดความเร็วลมเข้าฐานข้อมูล	106

## สารบัญ (ต่อ)

	หน้า
บทที่ 4 ผลการทดลอง	94
4.4 การทดสอบเซนเซอร์วัดปริมาณน้ำฝน	108
4.4.1 การทดสอบการทำงานของเซนเซอร์วัดปริมาณน้ำฝน	108
4.4.2 การทดสอบการนำค่าเซนเซอร์วัดปริมาณน้ำฝนเข้าฐานข้อมูล	110
4.5 การทดสอบเซนเซอร์วัดอุณหภูมิ	113
4.5.1 การทดสอบการทำงานของเซนเซอร์วัดอุณหภูมิ	113
4.5.2 การทดสอบนำค่าเซนเซอร์วัดอุณหภูมิเข้าฐานข้อมูล	115
4.6 การทดสอบแสดงค่าเซนเซอร์ใช้แบบร่วมกัน	117
4.6.1 การทดสอบการทำงานของเซนเซอร์ใช้แบบร่วมกัน	117
4.6.2 การทดสอบนำค่าเซนเซอร์ของเซนเซอร์ใช้แบบร่วมกันเข้าฐานข้อมูล	121
4.7 การทดสอบกล้อง ESP-32 CAM	124
4.7.1 การทดสอบการทำงานของกล้อง ESP-32 CAM	124
4.8 การทดสอบเว็บแอปพลิเคชันจตุรายนสภาพอากาศ	125
4.8.1 การทดสอบการทำงานของเว็บแอปพลิเคชันจตุรายนสภาพอากาศ	125
4.9 การทดสอบโมบายแอปพลิเคชันจตุรายนสภาพอากาศ	128
4.9.1 การทดสอบการทำงานของเว็บแอปพลิเคชันจตุรายนสภาพอากาศ	128
บทที่ 5 สรุปผลการทดลอง	131
5.1 สรุปผลการทดลอง	131
5.1.1 สรุปผลการทดลองส่งค่าของเซ็นเซอร์	131
5.1.2 สรุปผลการทดลองส่งค่าของเซ็นเซอร์บันทึกลงดาต้าเบส	131
5.1.3 สรุปผลการแสดงค่าเซ็นเซอร์ออกหน้าเว็บ	131
5.1.4 สรุปผลการแสดงค่าเซ็นเซอร์ออกโมบายแอปพลิเคชัน	131
5.1.5 สรุปผลการถ่ายภาพของกล้องESP32 CAM	131
5.2 ปัญหาและอุปสรรคในการทำงาน	131
5.3 แนวทางแก้ไขปัญหาและอุปสรรค	131
5.4 ข้อเสนอแนะแล้วทางการพัฒนา	132

## สารบัญตาราง

ตารางที่		หน้า
1.1	แผนการดำเนินงานภาคเรียนที่ 1	3
1.2	แผนการดำเนินงานภาคเรียนที่ 2	4
2.1	ตารางข้อมูลเกี่ยวกับขาของเซนเซอร์วัดความเข้มแสง BH1750	15
2.2	ข้อมูลเกี่ยวกับขาของเซนเซอร์วัดความเข้มแสง BH1750	24
2.3	การเชื่อมต่อขาของเซนเซอร์วัดความเร็วลมให้เข้ากับขาของ ESP8266	27
2.4	ความสัมพันธ์ของแรงดันไฟฟ้าด้านเอาต์พุตและความเร็วลม	27
2.5	สถานะของการต่อใช้งานเซนเซอร์วัดปริมาณน้ำฝน	30
2.6	สถานะของการต่อใช้งานเซนเซอร์วัดอุณหภูมิ DHT11เข้ากับ ESP8266	32
2.7	การเชื่อมต่อขาของกล่อง ESP32 กับ FT232RLUSB ให้ทำงานร่วมกัน	36
2.8	แท็กคำสั่งในภาษา HTML	41
4.1	ตารางแสดงผลทดสอบเซนเซอร์เพื่อวัดค่าความเข้มของรังสีอัลตราไวโอเล็ต GUVAS12SD	95
4.2	ตารางแสดงผลการทดสอบนำค่าเซนเซอร์ที่ได้มาวัดค่าความเข้มของรังสีอัลตราไวโอเล็ต GUVA S12SD บนฐานข้อมูล	97
4.3	ตารางแสดงผลทดสอบเซนเซอร์วัดค่าความเข้มของแสง BH1750	100
4.4	ตารางแสดงผลการทดสอบการนำค่าเซนเซอร์ที่วัดค่าความเข้มของแสง BH1750 บนฐานข้อมูล	102
4.5	ตารางแสดงผลการทดสอบเซนเซอร์วัดความเร็วลม	104
4.6	ตารางแสดงผลการทดสอบการนำค่าเซนเซอร์วัดความเร็วลมบนฐานข้อมูล	107
4.7	ตารางแสดงผลทดสอบของเซนเซอร์วัดปริมาณน้ำฝน	109
4.8	ตารางแสดงผลของทดสอบนำค่าเซนเซอร์วัดปริมาณน้ำฝนบนฐานข้อมูล	111
4.9	ตารางแสดงผลการทดสอบของเซนเซอร์วัดอุณหภูมิ	114
4.10	ตารางแสดงผลทดสอบของการนำค่าเซนเซอร์วัดอุณหภูมิบนฐานข้อมูล	116
4.11	ตารางแสดงผลการทดสอบของเซนเซอร์ใช้แบบร่วมกัน	119
4.12	ตารางแสดงผลการทดสอบการนำค่าเซนเซอร์ทุกเซนเซอร์บนฐานข้อมูล	122
4.13	ตารางแสดงผลการทดสอบการทำงานของกล่อง ESP-32 CAM	125
4.14	ตารางแสดงผลการทดสอบของเว็บแอปพลิเคชันจตุรายนสภาพอากาศ	126
4.15	ตารางแสดงผลการทดสอบของโมบายแอปพลิเคชันจตุรายนสภาพอากาศ	128

## สารบัญรูป

รูปที่		หน้า
2.1	ไมโครคอนโทรลเลอร์ ESP8266	6
2.2	หน้าต่างของโปรแกรม Arduino IDE	7
2.3	การใส่ลิงก์ของ ESP8266 ลงใน Arduino IDE	7
2.4	การค้นหา ESP8266 ในโปรแกรม Arduino IDE	8
2.5	การติดตั้งชุดคำสั่งสำหรับ ESP8266 ลงใน Arduino IDE	8
2.6	ชุดคำสั่งของ ESP8266 ที่ติดตั้งใน Arduino IDE เรียบร้อยแล้ว	9
2.7	ขั้นตอนการทำให้ Arduino IDE รู้จักกับ ESP8266	9
2.8	การเชื่อมต่อพอร์ตระหว่าง Arduino IDE กับ ESP8266	10
2.9	การบอกรหัสที่เชื่อมต่อระหว่าง Arduino IDE กับ ESP8266	11
2.10	หน้าต่างเริ่มต้นของโปรแกรม Arduino IDE	11
2.11	ตัวอย่างการเขียนโค้ดใน Arduino IDE	12
2.12	ผลลัพธ์จากการรันโปรแกรม	13
2.13	โครงสร้างของบอร์ด ESP8266	13
2.14	ตัวอย่างโค้ดสำหรับเชื่อมต่อกับ Wi-Fi สำหรับ ESP8266	14
2.15	การอัปโหลดโค้ดไปใส่ไว้ในตัว ESP8266	14
2.16	ด้านหน้าของเซนเซอร์วัดความเข้มแสง BH1750	15
2.17	ด้านหลังของเซนเซอร์วัดความเข้มแสง BH1750	15
2.18	การเชื่อมต่อระหว่าง ESP8266 กับเซนเซอร์วัดความเข้มแสง BH1750	16
2.19	การดาวน์โหลดไลบรารีของเซนเซอร์วัดความเข้มแสง BH1750	17
2.20	การติดตั้งไลบรารีของเซนเซอร์วัดความเข้มแสง BH1750	17
2.21	ตัวอย่างโค้ดที่ทำให้ ESP8266 ทำงานร่วมกับ BH1750	18
2.22	การเลือกรุ่นของบอร์ด ESP8266 เพื่อการอัปโหลด	18
2.23	การเชื่อมต่อระหว่าง ESP8266 กับ BH1750 ผ่านพอร์ตที่ต้องการ	19
2.24	ผลลัพธ์จากการอัปโหลดโค้ดเพื่อใช้งาน BH1750 เข้าไปใน ESP8266	19
2.25	การกำหนด Serial Monitor ให้กับบอร์ด ESP8266	20
2.26	การกำหนด Baud rate ให้ ESP8266 สื่อสารกับ BH1750 ได้ถูกต้อง	20
2.27	ค่าที่ได้จากการวัดค่าความเข้มแสงที่ปรากฏบน Serial Monitor	21
2.28	ตัวอย่างจริงในการเชื่อมต่อระหว่าง ESP8266 กับ BH1750	21
2.29	การวัดค่าความเข้มแสงของ BH1750 เมื่อมีความเข้มแสงสูงขึ้น	22

## สารบัญรูป (ต่อ)

รูปที่		หน้า
2.30	ตัวอย่างเซนเซอร์ยี่ห้อ GUVA S12SD	22
2.31	รูปการเปรียบเทียบค่าแรงดันไฟฟ้ากระแสตรงกับเลขดัชนีของความเข้มรังสียูวี	23
2.32	เลขดัชนีความเข้มรังสียูวีกับข้อปฏิบัติตัวเพื่อปกป้องอันตรายจากรังสียูวี	24
2.33	ตัวอย่างการต่อวงจรเซนเซอร์ยี่ห้อ GUVA S12SD กับ ESP8266	24
2.34	ตัวอย่างโค้ดที่ทำให้ ESP8266 ทำงานร่วมกับเซนเซอร์ยี่ห้อ GUVA S12SD ได้	24
2.35	เซนเซอร์วัดความเร็วลม	25
2.36	โครงสร้างของเซนเซอร์วัดความเร็วลม	26
2.37	แบบที่ตำแหน่งฐานของเซนเซอร์สำหรับการติดตั้ง	26
2.38	สถานะขาของเซนเซอร์วัดความเร็วลม	27
2.39	การต่อใช้งานของเซนเซอร์วัดความเร็วลมกับ ESP8266	27
2.40	โค้ดตัวอย่างการใช้งานเซนเซอร์วัดความเร็วลมซึ่งต่อกับ ESP8266	28
2.41	ลักษณะภายในของเซนเซอร์วัดปริมาณน้ำฝน	29
2.42	เซนเซอร์วัดปริมาณน้ำฝน	29
2.43	การต่อใช้งานของเซนเซอร์วัดปริมาณน้ำฝน	30
2.44	ตัวอย่างโค้ดการใช้งานเซนเซอร์วัดปริมาณน้ำฝนที่เชื่อมต่อกับ ESP8266	30
2.45	เซนเซอร์วัดอุณหภูมิและความชื้น DHT11	31
2.46	การเชื่อมต่อเซนเซอร์ DHT11 ให้เข้ากับ ESP8266	32
2.47	ตัวอย่างโค้ดสำหรับสั่งการให้เซนเซอร์ DHT11 ทำงานร่วมกับ ESP8266	33
2.48	โมดูลกล่อง ESP32	33
2.50	การบอกจุดการใช้งานที่สำคัญของโมดูลกล่อง ESP32	35
2.51	การต่อวงจรใช้งานกล่อง ESP32 กับ FT232RLUSB	35
2.52	ขั้นตอนแรกของการค้นหาไดรเวอร์ของกล่อง ESP32	35
2.53	ขั้นตอนที่สองของการค้นหาไดรเวอร์ของกล่อง ESP32	35
2.54	คอมพิวเตอร์กับกล่อง ESP32 ยังไม่สามารถทำงานร่วมกันได้	37
2.55	การอัปเดตไดรเวอร์ของกล่อง ESP32	38
2.56	การค้นหาไฟล์ไดรเวอร์ของ FT232RLUSB	38
2.57	การติดตั้งไดรเวอร์กล่อง ESP32 ที่เสร็จเรียบร้อยแล้ว	39
2.58	ตัวอย่างโค้ดในการควบคุมให้กล่อง ESP32 ทำงานได้	39
2.59	ส่วนประกอบของเอกสาร HTML ที่สำคัญ	42

## สารบัญรูป (ต่อ)

รูปที่		หน้า
2.60	ตัวอย่างการสร้างเอกสาร HTML ด้วยโปรแกรม Notepad	42
2.61	ตัวอย่างการเซฟไฟล์เป็นนามสกุล .html	43
2.62	ผลลัพธ์จากการเปิดไฟล์ test.html	43
2.63	การเริ่มใช้งานโปรแกรม EditPlus	44
2.64	การเริ่มเขียนโค้ด HTML ด้วยโปรแกรม EditPlus	44
2.65	การเริ่มใช้งานโปรแกรม Dreamweaver	45
2.66	เอกสาร HTML โดย Dreamweaver	45
3.1	ภาพรวมของระบบจตุรายนสภาพอากาศ	53
3.2	การออกแบบเสาของจตุรายนสภาพอากาศ	54
3.3	การออกแบบวงจรเพื่อเชื่อมต่อเซนเซอร์ทุกตัว	54
3.4	รูปของวงจรพีซีบีของไมโครคอนโทรลเลอร์ ESP8266	55
3.5	การประกอบเสาจตุรายนสภาพอากาศ	55
3.6	การประกอบภายในกล่องควบคุมอุปกรณ์	56
3.7	รูปด้านหลังของเสาจตุรายนสภาพอากาศ	56
3.8	ด้านหน้าของจตุรายนสภาพอากาศ ตัวที่ 1	57
3.9	ด้านข้างของจตุรายนสภาพอากาศ ตัวที่ 1	58
3.10	ด้านหน้าของจตุรายนสภาพอากาศ ตัวที่ 1 แบบถ่ายระยะใกล้	58
3.11	ภายในกล่องควบคุมของจตุรายนสภาพอากาศ ตัวที่ 1	59
3.12	ด้านหน้าของจตุรายนสภาพอากาศ ตัวที่ 2	60
3.13	ด้านหลังของจตุรายนสภาพอากาศ ตัวที่ 2	60
3.14	ภายในกล่องควบคุมของจตุรายนสภาพอากาศ ตัวที่ 2	61
3.15	การติดตั้งจตุรายนสภาพอากาศที่บริเวณหาดพระจอม	61
3.16	การสร้างตารางเก็บข้อมูลสำหรับจตุรายนสภาพอากาศ ตัวที่ 1	62
3.17	ข้อมูลที่ได้จากจตุรายนสภาพอากาศ ตัวที่ 1	63
3.18	การสร้างตารางเก็บข้อมูลสำหรับจตุรายนสภาพอากาศ ตัวที่ 2	63
3.19	ข้อมูลที่ได้จากจตุรายนสภาพอากาศ ตัวที่ 2	64
3.20	โพลเดอร์สำหรับเก็บไฟล์รูปไว้ในฐานข้อมูล	64
3.21	การเก็บไฟล์รูปจากจตุรายนสภาพอากาศ ตัวที่ 1	65
3.22	การเก็บไฟล์รูปจากจตุรายนสภาพอากาศ ตัวที่ 2	65

## สารบัญรูป (ต่อ)

รูปที่		หน้า
3.23	แผนผังการทำงานของโปรแกรมของจตุรรายงานสภาพอากาศ	66
3.24	ซอร์สโค้ดของจตุรรายงานสภาพอากาศ ส่วนที่ 1	67
3.25	ซอร์สโค้ดของจตุรรายงานสภาพอากาศ ส่วนที่ 2	67
3.26	ซอร์สโค้ดของจตุรรายงานสภาพอากาศ ส่วนที่ 3	68
3.27	ซอร์สโค้ดของจตุรรายงานสภาพอากาศ ส่วนที่ 4	68
3.28	ซอร์สโค้ดของจตุรรายงานสภาพอากาศ ส่วนที่ 5	69
3.29	ซอร์สโค้ดของจตุรรายงานสภาพอากาศ ส่วนที่ 6	69
3.30	ซอร์สโค้ดสำหรับกล่อง ESP32 ของจตุรรายงานสภาพอากาศ ส่วนที่ 1	70
3.31	ซอร์สโค้ดสำหรับกล่อง ESP32 ของจตุรรายงานสภาพอากาศ ส่วนที่ 2	71
3.32	ซอร์สโค้ดสำหรับกล่อง ESP32 ของจตุรรายงานสภาพอากาศ ส่วนที่ 3	71
3.33	ซอร์สโค้ดสำหรับกล่อง ESP32 ของจตุรรายงานสภาพอากาศ ส่วนที่ 4	71
3.34	ซอร์สโค้ดสำหรับกล่อง ESP32 ของจตุรรายงานสภาพอากาศ ส่วนที่ 5	72
3.35	ซอร์สโค้ดสำหรับกล่อง ESP32 ของจตุรรายงานสภาพอากาศ ส่วนที่ 6	72
3.36	ซอร์สโค้ดสำหรับกล่อง ESP32 ของจตุรรายงานสภาพอากาศ ส่วนที่ 7	72
3.37	แผนผังเว็บไซต์และโมบายแอปพลิเคชัน	73
3.38	หน้าแรกของเว็บไซต์	74
3.39	การเลือกดูข้อมูลจากจตุรรายงานสภาพอากาศ ตัวที่ 1 หรือ 2	74
3.40	เมนูสำหรับเลือกดูข้อมูลของจตุรรายงานสภาพอากาศ ตัวที่ 1	75
3.41	เมนูสำหรับเลือกดูข้อมูลของจตุรรายงานสภาพอากาศ ตัวที่ 2	75
3.42	รูปถ่ายจากจตุรรายงานสภาพอากาศ ตัวที่ 1	76
3.43	รูปถ่ายจากจตุรรายงานสภาพอากาศ ตัวที่ 2	76
3.44	หน้าต่างข้อมูลของเซนเซอร์แบบกราฟจากจตุรรายงานสภาพอากาศ ตัวที่ 1	76
3.45	หน้าต่างข้อมูลของเซนเซอร์แบบกราฟจากจตุรรายงานสภาพอากาศ ตัวที่ 2	77
3.46	กราฟวัดค่าความเข้มข้นสียูวีย้อนหลังเจ็ดวัน	77
3.47	กราฟวัดค่าปริมาณน้ำฝนย้อนหลังเจ็ดวัน	77
3.48	กราฟวัดค่าความเข้มแสงย้อนหลังเจ็ดวัน	78
3.49	กราฟวัดค่าอุณหภูมิและความชื้นย้อนหลังเจ็ดวัน	78
3.50	กราฟวัดค่าความเร็วลมย้อนหลังเจ็ดวัน	78
3.51	หน้าต่างข้อมูลของเซนเซอร์จากจตุรรายงานสภาพอากาศ ตัวที่ 1	79

## สารบัญรูป (ต่อ)

รูปที่		หน้า
3.52	หน้าต่างข้อมูลของเซนเซอร์จากจตุรายนสภาพอากาศ ตัวที่ 2	79
3.53	ข้อมูลระดับของปริมาณน้ำฝน	80
3.54	ข้อมูลระดับความเข้มรังสียูวี	80
3.55	ข้อมูลระดับอุณหภูมิ	80
3.56	ข้อมูลระดับความเข้มแสง	81
3.57	ข้อมูลระดับความเร็วลม	81
3.58	แอปพลิเคชันที่ติดตั้งบนโทรศัพท์	81
3.59	ไอคอนของแอปพลิเคชัน WeatherApp	82
3.60	หน้าแรกของแอปพลิเคชัน WeatherApp	82
3.61	การเลือกดูข้อมูลของจตุรายนสภาพอากาศ	82
3.62	เมนู PHOTO	83
3.63	เมนู CHART	83
3.64	เมนู AFTER	84
3.65	ภาพถ่ายจากจตุรายนสภาพอากาศ ตัวที่ 1	84
3.66	ภาพถ่ายจากจตุรายนสภาพอากาศ ตัวที่ 2	85
3.67	หน้าแรกของกราฟข้อมูลการรายงานสภาพอากาศ	85
3.68	กราฟความเร็วลมย้อนหลังเจ็ดวัน	85
3.69	กราฟความเข้มรังสียูวีย้อนหลังเจ็ดวัน	86
3.70	กราฟความเข้มแสงย้อนหลังเจ็ดวัน	87
3.71	กราฟอุณหภูมีย้อนหลังเจ็ดวัน	87
3.72	กราฟปริมาณน้ำฝนย้อนหลังเจ็ดวัน	88
3.73	การแสดงผลข้อมูลจากเซนเซอร์วัดอุณหภูมิและความชื้น	88
3.74	การแสดงผลข้อมูลจากเซนเซอร์วัดปริมาณน้ำฝน	89
3.75	การแสดงผลข้อมูลจากเซนเซอร์วัดความเข้มรังสียูวี	89
3.76	การแสดงผลข้อมูลจากเซนเซอร์วัดความเร็วลม	90
3.77	การแสดงผลข้อมูลจากเซนเซอร์วัดความเข้มแสง	90
3.78	Pop-up คู่มือรายงานปริมาณน้ำฝนย้อนหลัง 7 วัน	91
3.79	Pop-up คู่มือรายงานความเข้มรังสียูวีย้อนหลัง 7 วัน	91
3.80	Pop-up คู่มือรายงานอุณหภูมีย้อนหลัง 7 วัน	92

## สารบัญรูป (ต่อ)

รูปที่		หน้า
3.81	Pop-up คู่มือรายงานความเข้มแสงย้อนหลัง 7 วัน	92
3.82	Pop-up คู่มือรายงานความเร็วลมย้อนหลัง 7 วัน	93
4.1	การต่อใช้งานอีเอสพีแปดสองหกหกกับเซนเซอร์เพื่อวัดค่าความเข้มของ รังสีอัลตราไวโอเล็ต GUV A S12SD	94
4.2	โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์เพื่อวัดค่าความเข้มของ รังสีอัลตราไวโอเล็ต GUV A S12SD	95
4.3	โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์เพื่อวัดค่าความเข้มของ รังสีอัลตราไวโอเล็ต GUV A S12SD บนฐานข้อมูล	97
4.4	แสดงการส่งข้อมูลเซนเซอร์วัดค่าความเข้มของรังสีอัลตราไวโอเล็ต GUV A S12SD ไปยังฐานข้อมูล	97
4.5	การต่อใช้งานอีเอสพีแปดสองหกหกกับเซนเซอร์วัดค่าความเข้มของแสง BH1750	99
4.6	โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์เพื่อวัดค่าความเข้มของแสง BH1750	99
4.7	โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์ที่วัดค่าความเข้มของแสง BH1750 บนฐานข้อมูล	101
4.8	แสดงการส่งข้อมูลวัดค่าความเข้มแสง BH1750 ไปยังฐานข้อมูล	102
4.9	การต่อใช้งานอีเอสพีแปดสองหกหกเข้ากับเซนเซอร์วัดความเร็วลม	104
4.10	โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์วัดความเร็วลม	104
4.11	โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์วัดความเร็วลมบนฐานข้อมูล	106
4.12	แสดงการส่งข้อมูลวัดความเร็วลมไปยังฐานข้อมูล	106
4.13	การต่อใช้งานอีเอสพีแปดสองหกหกกับเซนเซอร์วัดปริมาณน้ำฝน	108
4.14	โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์วัดปริมาณน้ำฝน	109
4.15	โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์วัดปริมาณน้ำฝนบนฐานข้อมูล	111
4.16	แสดงการส่งข้อมูลวัดปริมาณน้ำฝนไปยังฐานข้อมูล	111
4.17	การต่อใช้งานอีเอสพีแปดสองหกหกกับเซนเซอร์วัดอุณหภูมิ	113
4.18	โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์วัดอุณหภูมิ	113
4.19	โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์วัดอุณหภูมิบนฐานข้อมูล	115
4.20	การต่อใช้งานอาร์ดูอิโน้กับเซนเซอร์รวมกันทุกตัว	117
4.21	โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์รวมกันทุกตัว	118
4.22	โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์รวมกันทุกตัวบนฐานข้อมูล	121
4.23	แสดงการส่งข้อมูลเซนเซอร์รวมกันทุกตัวไปยังฐานข้อมูล	121

## สารบัญรูป (ต่อ)

รูปที่	หน้า	
4.24	การต่ออุปกรณ์งานESP32-CAM เข้ากับ FT232RLUSB เพื่อทำการทดสอบ	124
4.25	โค้ดที่ใช้สำหรับการทดสอบ ESP32-CAM บนฐานข้อมูล	124
4.26	รูปภาพที่กล้อง ESP-32 CAM ทำการถ่ายได้นำมาเก็บไว้บนฐานข้อมูล	125
4.27	หน้าแสดงข้อมูลหน้าเว็บแอปพลิเคชันจตุรายนสภาพอากาศ	127
4.28	หน้าแสดงรูปภาพที่กล้อง ESP-32 CAM ทำการถ่ายภาพหน้าเว็บแอปพลิเคชันตรงจุด รายงานสภาพอากาศ	127
4.29	หน้าแสดงข้อมูลหน้าโมบายแอปพลิเคชันจตุรายนสภาพอากาศ	129
4.30	หน้าแสดงข้อมูลหน้าโมบายแอปพลิเคชันจตุรายนสภาพอากาศ	130
ก.1	หน้าเว็บแอปพลิเคชันจตุรายนสภาพอากาศ	136
ก.2	หน้าเลือกสถานที่ของเว็บแอปพลิเคชันจตุรายนสภาพอากาศ	136
ก.3	หน้าแรกของสถานที่ 1	137
ก.4	หน้าแรกของสถานที่ 2	137
ก.5	ฟังก์ชันดูภาพถ่ายล่าสุดของสถานที่ 1	138
ก.6	ฟังก์ชันดูภาพถ่ายล่าสุดของสถานที่ 2	139
ก.7	ดูภาพถ่ายล่าสุดของสถานที่ 1	139
ก.8	ดูภาพถ่ายล่าสุดของสถานที่ 2	140
ก.9	ฟังก์ชันดูรายงานสภาพอากาศของสถานที่ 1	140
ก.10	ฟังก์ชันดูรายงานสภาพอากาศของสถานที่ 2	141
ก.11	หน้าแรกเว็บดูรายงานสภาพอากาศสถานที่ 1	141
ก.12	ฟังก์ชันของหน้ารายงานสภาพอากาศ	142
ก.13	หน้าเว็บค่าล่าสุดที่วัดได้สถานที่1	142
ก.14	ป๊อปอัพค่าการวัดมาตรฐานของปริมาณน้ำฝน	143
ก.15	ป๊อปอัพค่าการวัดมาตรฐานของความชื้นยูวี	143
ก.16	ป๊อปอัพค่าการวัดมาตรฐานของอุณหภูมิ	144
ก.17	ป๊อปอัพค่าการวัดมาตรฐานของความเข้มแสง	144
ก.18	ป๊อปอัพค่าการวัดมาตรฐานของความเร็วลม	145
ก.19	หน้าแรกเว็บดูรายงานสภาพอากาศสถานที่ 2	145
ก.20	ฟังก์ชันของหน้ารายงานสภาพอากาศ	146
ก.21	หน้าเว็บค่าล่าสุดที่วัดได้สถานที่ 2	146

## สารบัญรูป (ต่อ)

รูปที่	หน้า
ก.22 ฟังก์ชันดูรายงานสภาพอากาศย้อนหลัง 7 วันของสถานที่ 1	147
ก.23 ฟังก์ชันดูรายงานสภาพอากาศย้อนหลัง 7 วันของสถานที่ 2	147
ก.23 ฟังก์ชันของรายงานสภาพอากาศย้อนหลัง 7 วัน	148
ก.24 รายงานสภาพอากาศย้อนหลัง 7 วันของความเข้มยูวี	148
ก.25 รายงานสภาพอากาศย้อนหลัง 7 วันของปริมาณน้ำฝน	149
ก.26 รายงานสภาพอากาศย้อนหลัง 7 วันของความเข้มแสง	149
ก.27 รายงานสภาพอากาศย้อนหลัง 7 วันของอุณหภูมิ	150
ก.28 รายงานสภาพอากาศย้อนหลัง 7 วันของความเร็วลม	150



# บทที่ 1

## บทนำ

ในบทนี้จะกล่าวถึง ความเป็นมาและความสำคัญ วัตถุประสงค์ของโครง ขอบเขตของ  
โครงการ ประโยชน์ที่ได้รับ แผนการดำเนินงาน และโครงสร้างของปริญญาโท

### 1.1 ความเป็นมาและความสำคัญ

ประเทศไทยเป็นประเทศหนึ่งในโลกที่มีภูมิประเทศ ทัศนียภาพ และแหล่งท่องเที่ยวที่สวยงามมากมาย จึงทำให้คนในชาตินิยมไปท่องเที่ยวตามสถานที่ท่องเที่ยวทั่วประเทศ รวมไปถึงนักท่องเที่ยวต่างชาติก็นิยมเดินทางมาท่องเที่ยวในประเทศไทยด้วยเหมือนกัน ผู้จัดทำขอยกตัวอย่างแหล่งท่องเที่ยวที่สำคัญและเป็นที่ยอมรับโด่งดังไปทั่วโลกเช่น ภูเขาทางภาคเหนือ ทะเลภาคใต้ ห้างสรรพสินค้า และผืนป่าอันดามัน วัดในพระพุทธศาสนาที่สวยงามด้วยศิลปะ และพระบรมมหาราชวัง เป็นต้น ซึ่งการท่องเที่ยวทำให้เกิดรายได้อย่างมากมายมหาศาลต่อประเทศและผู้ประกอบการที่เกี่ยวข้อง หากแต่สภาพดินฟ้าอากาศก็เป็นส่วนประกอบหนึ่งในการตัดสินใจของนักท่องเที่ยวว่าจะไปเยือนสถานที่ท่องเที่ยวนั้นหรือไม่ ยกตัวอย่างเช่น หากเกิดฝนตกหนักก็ไม่อาจไปเที่ยวภูเขาได้ หรือแสงแดดที่แรงกล้าก็ส่งผลให้ไม่สามารถเล่นน้ำทะเลได้ เป็นต้น ข้อมูลของสภาพอากาศทั้งหลายไม่ว่าจะเป็น อุณหภูมิ ความชื้นของรังสีอัลตราไวโอเล็ต ความเร็วลม ความชื้นในอากาศ และอื่นๆ ประชาชนทั่วไปจะรับรู้ได้จากกรมอุตุนิยมวิทยา แต่การรายงานสภาพอากาศจะเป็นการพยากรณ์และเป็นการอธิบายโดยภาพรวม ไม่ได้เจาะจงบอกข้อมูลเฉพาะสถานที่ท่องเที่ยวจุดนั้น หากรู้ข้อมูลเหล่านี้ได้ด้วยสภาพอากาศปัจจุบันก็จะส่งผลให้การตัดสินใจไปท่องเที่ยวได้ง่ายดายยิ่งขึ้น

จากสาเหตุที่กล่าวไปข้างต้น ผู้จัดทำจึงได้มีแนวคิดที่จะประดิษฐ์ชุดรายงานสภาพอากาศแบบทันทีทันใดหรือเวลาจริง (Realtime) ที่สามารถสภาพอากาศได้ด้วยข้อมูลหลายอย่างรวมทั้งมีภาพถ่ายสถานที่จริงให้นักท่องเที่ยวเข้าไปรับรู้ข้อมูลผ่านเว็บไซต์และแอปพลิเคชัน เพื่อให้ นักท่องเที่ยวสามารถตัดสินใจไปท่องเที่ยวได้อย่างง่ายดายด้วยข้อมูลที่มี และโครงการนี้ยังเป็นต้นแบบให้ขยายผลเป็นโครงข่ายการรายงานสภาพอากาศแบบเวลาจริงต่อไปในอนาคต การจัดทำโครงการนี้จะมีชุดรายงานสภาพอากาศไว้ 2 ชุดคือ หาดทุ่งช้างและหาดพระจอม ต.ชุมโค อ.ปะทิว จ.ชุมพร

### 1.2 วัตถุประสงค์ของโครงการ

- 1.2.1 เพื่อให้ให้นักท่องเที่ยวได้รับทราบข้อมูลสภาพอากาศที่หาดทุ่งช้างและหาดพระจอม
- 1.2.2 เพื่อให้เป็นต้นแบบของการสร้างโครงข่ายรายงานสภาพอากาศในอนาคต

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.2 วัตถุประสงค์ของโครงการ(ต่อ)

1.2.3 เพื่อเรียนรู้และฝึกทักษะการควบคุมไมโครคอนโทรลเลอร์และเซนเซอร์ที่เกี่ยวข้องอีก ทั้งส่งค่าที่วัดได้จากเซนเซอร์ไปเก็บไว้ในฐานข้อมูล

1.2.4 เพื่อเรียนรู้และฝึกฝนการใช้งานโปรแกรมฐานข้อมูล

1.2.5 เพื่อเรียนรู้และฝึกฝนการสร้างเว็บไซต์ด้วยภาษา HTML/CSS/JavaScript/SQL

1.2.6 เพื่อเรียนรู้และฝึกฝนการสร้างโมบายแอปพลิเคชัน

1.2.7 เพื่อฝึกฝนการวางแผนงาน การปฏิบัติจริง และนำความรู้มาประยุกต์ใช้

## 1.3 ขอบเขตของโครงการ

1.3.1 วัดค่าอุณหภูมิ ความชื้น ความเข้มของแสง ความเข้มของรังสีอัลตราไวโอเล็ต ความเร็วลม และปริมาณน้ำฝนด้วยเซนเซอร์

1.3.2 ติดกล้องสำหรับถ่ายรูปภาพอากาศจริง ณ สถานที่แห่งนั้น

1.3.3 ค่าที่วัดได้จากเซนเซอร์และรูปถ่ายจะเก็บไว้ในฐานข้อมูล

1.3.4 สร้างเว็บไซต์และโมบายแอปพลิเคชันสำหรับดูข้อมูลสภาพอากาศ และดูข้อมูลย้อนหลังได้เต็มที่ 7 วัน โดยแสดงข้อมูลเป็นกราฟ

1.3.5 สร้างจตุรายนสภาพอากาศไว้ 2 แห่ง

1.3.6 บนเว็บไซต์และโมบายแอปพลิเคชันสามารถเลือกแสดงข้อมูลแต่ละแห่งได้

1.3.7 สร้างโมบายแอปพลิเคชันที่สามารถติดตั้งบนสมาร์ตโฟนได้

## 1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 นักท่องเที่ยวได้รับทราบข้อมูลสภาพอากาศแบบเวลจริงได้อย่างถูกต้อง

1.4.2 สามารถประดิษฐ์จตุรายนสภาพอากาศที่เป็นต้นแบบเอาไว้ใช้งานได้

1.4.3 ได้รับความรู้และทักษะในการสั่งงานไมโครคอนโทรลเลอร์ที่เชื่อมต่อกับเซนเซอร์ที่เกี่ยวข้องในงาน รวมทั้งส่งค่าที่วัดได้ไปเก็บไว้ในฐานข้อมูล

1.4.4 ได้รับความรู้และความชำนาญในการใช้งานโปรแกรมฐานข้อมูล

1.4.5 ได้รับความรู้และทักษะในการสร้างเว็บไซต์ด้วยภาษา HTML/CSS/JavaScript/SQL

1.4.6 ได้รับความรู้และทักษะในการสร้างโมบายแอปพลิเคชัน

1.4.7 ได้รับประสบการณ์ในการวางแผนงาน การปฏิบัติจริง และนำความรู้มาประยุกต์ใช้ให้เกิดประโยชน์ตามวัตถุประสงค์

## 1.5 แผนการดำเนินงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 1 แผนการดำเนินงานภาคเรียนที่ 1

ขั้นตอนการดำเนินงาน	ระยะเวลาดำเนินงาน (สิงหาคม - พฤศจิกายน 2563)																	
	สิงหาคม				กันยายน				ตุลาคม					พฤศจิกายน				
	1	2	3	4	1	2	3	4	1	2	3	4	5	1	2	3	4	
1.ศึกษาและรวบรวมข้อมูลเกี่ยวกับโครงการ	■	■	■															
2.เข้าพบอาจารย์ที่ปรึกษา				■	■													
3.นำเสนอหัวข้อโครงการต่อคณาจารย์						■	■	■										
4.จัดซื้ออุปกรณ์									■	■	■							
5.เชื่อมต่ออุปกรณ์เข้าหากันด้วยภาษา c												■	■					
6.ทดสอบและแก้ไข														■				
7.ออกแบบและสร้างฐานข้อมูล															■			
8.สร้างเว็บไซต์																■		
9.สอบปลายภาคโครงการ 1																		■



## 1.6 โครงสร้างของปฏิญานิพนธ์

ปฏิญานิพนธ์เรื่องจตุรายนสภาพอากาศฉบับนี้ มีโครงสร้างที่แบ่งเป็น 3 ส่วนหลัก ซึ่งจะอธิบายเนื้อหาในแต่ละส่วนไว้อย่างชัดเจน ดังรายละเอียดต่อไปนี้

บทที่ 1 บทนำ ในบทนี้จะกล่าวถึง ความเป็นและความสำคัญ วัตถุประสงค์ขอบเขตของโครงการ ประโยชน์ที่คาดว่าจะได้รับ แผนการดำเนินงาน และโครงสร้างของปฏิญานิพนธ์

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง ในบทนี้จะกล่าวถึง ไมโครคอนโทรลเลอร์อีเอสพี8266 วิธีการติดตั้งโหมดเอ็มซียู8266 ลงในโปรแกรมอาร์ดูอิโนไอดีอี การเขียนโปรแกรมในอาร์ดูอิโนไอดีอี การเชื่อมต่อไว-ไฟ และการสแกนหาชื่อไว-ไฟ เซนเซอร์วัดความเข้มแสง โมดูลเซนเซอร์ ยูวี เซ็นเซอร์วัดลม เซนเซอร์วัดปริมาณน้ำฝน โมดูล อีเอสพี 32 แคม โมดูลเซนเซอร์วัดอุณหภูมิ+ ความชื้น ภาษา เอสทีเอ็ม HTML โครงสร้างของภาษา HTML ภาษา CSS

บทที่ 3 การออกแบบระบบ ในบทนี้จะกล่าวถึง การทำงานและภาพรวมของระบบ การออกแบบโครงสร้างระบบรายงานสภาพอากาศ การออกแบบเชื่อมต่อวงจรเซนเซอร์ การออกแบบอุปกรณ์ต้นแบบ การติดตั้งเครื่องในสถานที่จริง ขั้นตอนการส่งข้อมูลจากเซนเซอร์ไปยังกาต้าเบส ขั้นตอนการส่งรูปถ่ายไปยังกาต้าเบส รูปแบบหน้าเว็บ หน้าแอปโมบาย การออกแบบการทดลอง

บทที่ 4 การทดลองและผลการทดลอง ในบทนี้จะกล่าวถึง ทดสอบการทำงานของแอปพลิเคชัน หน้าจอหลักเลือกตำแหน่งที่ตั้ง ทดสอบการรับค่าจากเซนเซอร์และแสดงค่าบนดาต้าเบส ทดสอบการส่งรูปถ่ายส่งขึ้นไปเก็บไว้ในดาต้าเบส

บทที่ 5 บทสรุปและข้อเสนอแนะ ในบทนี้จะกล่าวถึง สรุปผลการทดลอง ปัญหาและอุปสรรค ข้อเสนอแนะแล้วทางการพัฒนา

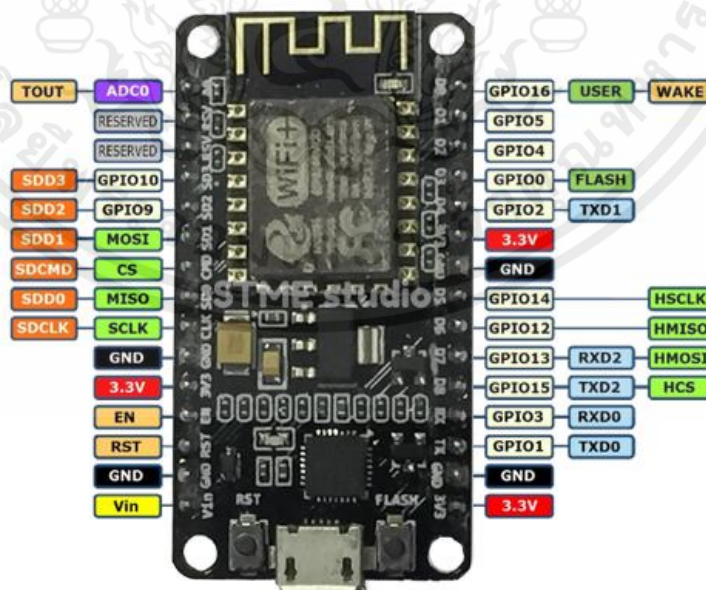
## บทที่ 2 ทฤษฎีที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึง ไมโครคอนโทรลเลอร์อีเอสพี8266 เซนเซอร์วัดความเข้มแสง BH1750 เซนเซอร์วัดความเข้มแสงยูวี GUVVA S12SD เซนเซอร์วัดความเร็วลม เซนเซอร์วัดปริมาณน้ำฝน เซนเซอร์วัดอุณหภูมิและความชื้น DHT11 โมดูลกล้อง ESP32 ภาษา HTML ภาษา CSS ภาษา JavaScript และโปรแกรม MySQL

### 2.1 ไมโครคอนโทรลเลอร์อีเอสพี 8266 (Microcontroller ESP8266)

ไมโครคอนโทรลเลอร์ ESP8266 [1] หรือมีอีกชื่อเรียกหนึ่งนั่นคือโนดเอ็มซียูอีเอสพี8266 (NodeMCU ESP8266) และสามารถเรียกโดยย่อว่าอีเอสพี8266 ก็ได้ ในหัวข้อนี้ผู้เขียนจะกล่าวถึง ไมโครคอนโทรลเลอร์ตัวนี้ โดยใช้คำว่า ESP8266 ซึ่งผู้จัดทำจะขออธิบายขยายความข้อมูลที่เกี่ยวข้องกับ ESP8266 ไว้ดังต่อไปนี้

ESP8266 คือบอร์ดไมโครคอนโทรลเลอร์ตัวหนึ่ง ซึ่งมีจุดประสงค์การใช้งานเหมือนกับบอร์ดไมโครคอนโทรลเลอร์อาร์ดูอิโน้ (Arduino) แต่ต่างกันตรงที่ ESP8266 จะสามารถเชื่อมต่อกับไว-ไฟ (Wi-Fi) ได้ด้วยตัวมันเอง เพราะภายในบอร์ด ESP8266 จะมีไมโครชิปที่ชื่อว่า ESP8266 ที่สามารถเชื่อมต่อกับ Wi-Fi ได้ ซึ่งการเชื่อมต่อนี้เป็นการสื่อสารกันด้วยโปรโตคอล TCP/IP ดังนั้น ESP8266 จึงเป็นแพลตฟอร์มไอโอที (IoT Platform) ที่สามารถเขียนโปรแกรมควบคุมการทำงานได้อีกด้วย

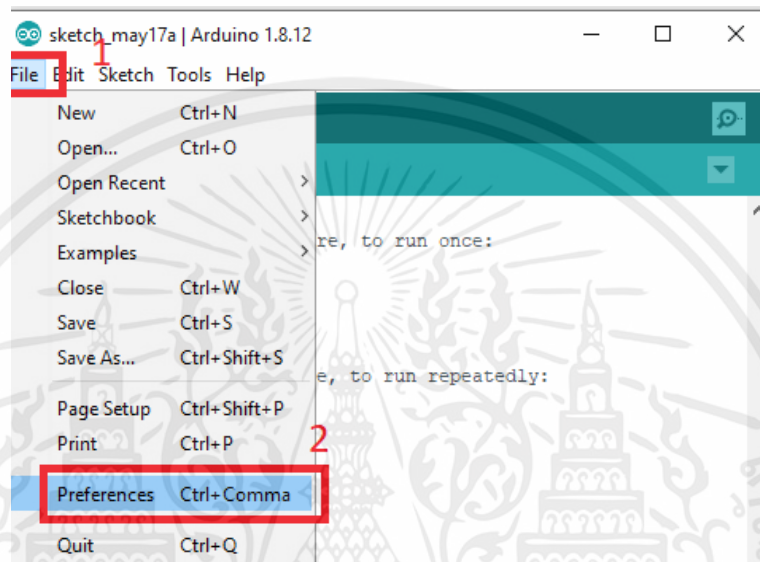


รูปที่ 2.1 ไมโครคอนโทรลเลอร์ ESP8266

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

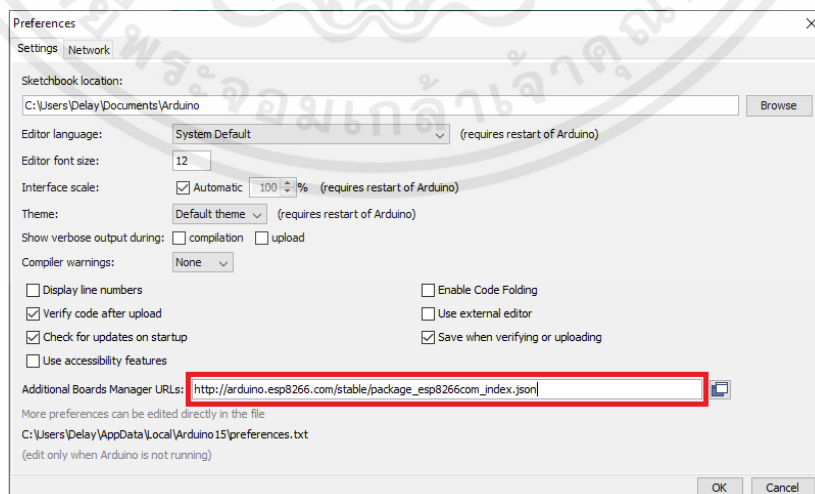
### 2.1.1 วิธีการติดตั้ง ESP8266 ลงในโปรแกรมอาร์ดูอโนไอดีอี (Arduino IDE)

1. เริ่มจากตรวจสอบก่อนว่าในเครื่องคอมพิวเตอร์ที่ใช้งานมีโปรแกรมอาร์ดูอโนไอดีอี หรือไม่ หากยังไม่มีสามารถดาวน์โหลดได้ที่ <https://www.arduino.cc/en/Main/Software>
2. เมื่อดาวน์โหลดและติดตั้งโปรแกรมเสร็จเรียบร้อยแล้ว ให้ไปที่เมนู File จากนั้นให้เลือก Preferences (หรือจะกด Ctrl+, ก็ได้) ดังรูปที่ 2.2



รูปที่ 2.2 หน้าต่างของโปรแกรม Arduino IDE

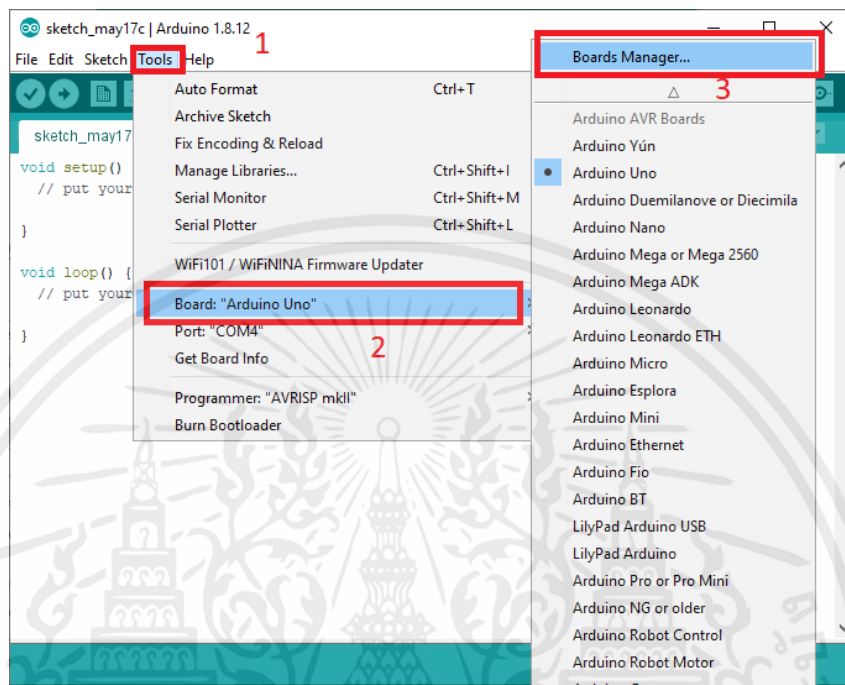
3. เพิ่ม URLs ลงไปในช่อง Additional Boards Manager URLs ซึ่งลิงก์ที่ต้องใส่ลงไปคือ [http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json) และกด OK



รูปที่ 2.3 การใส่ลิงก์ของ ESP8266 ลงใน Arduino IDE

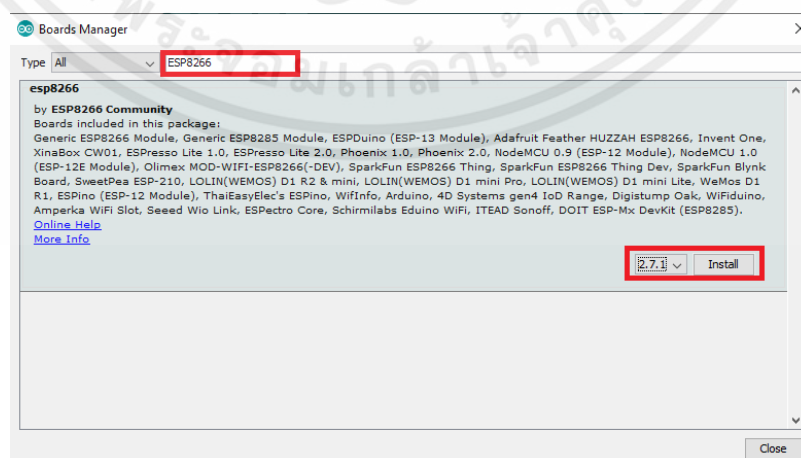
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เมื่อเพิ่มลิงก์ดังกล่าวลงใน ESP8266 เสร็จแล้ว ตัวโปรแกรมก็จะรู้ว่าต้องไปหาข้อมูลเกี่ยวกับบอร์ดตัวนี้ได้จากที่ไหน จากนั้นให้เลือกที่เมนู Tools > Board: “Arduino Uno” > Board Manager เพื่อทำการค้นหา ESP8266 ดังรูปที่ 2.4



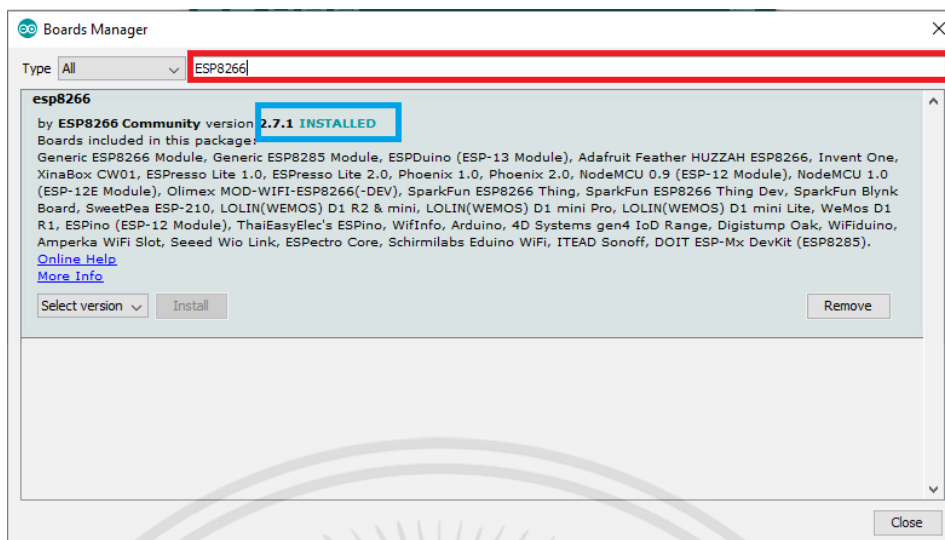
รูปที่ 2.4 การค้นหา ESP8266 ในโปรแกรม Arduino IDE

5. ให้ค้นหา ESP8266 ในช่องค้นหา (Search) ก็将会เห็น esp8266 by ESP8266 Community จากนั้นให้ทำการติดตั้ง (Install) โดยรุ่นของ ESP8266 ที่ได้จะเป็นรุ่นล่าสุด เมื่อติดตั้งเรียบร้อยแล้วจะมีข้อความขึ้นว่า INSTALLED ต่อจากนั้นให้ทำการปิดหน้าต่าง ดังรูปที่ 2.5-2.6



รูปที่ 2.5 การติดตั้งชุดคำสั่งสำหรับ ESP8266 ลงใน Arduino IDE

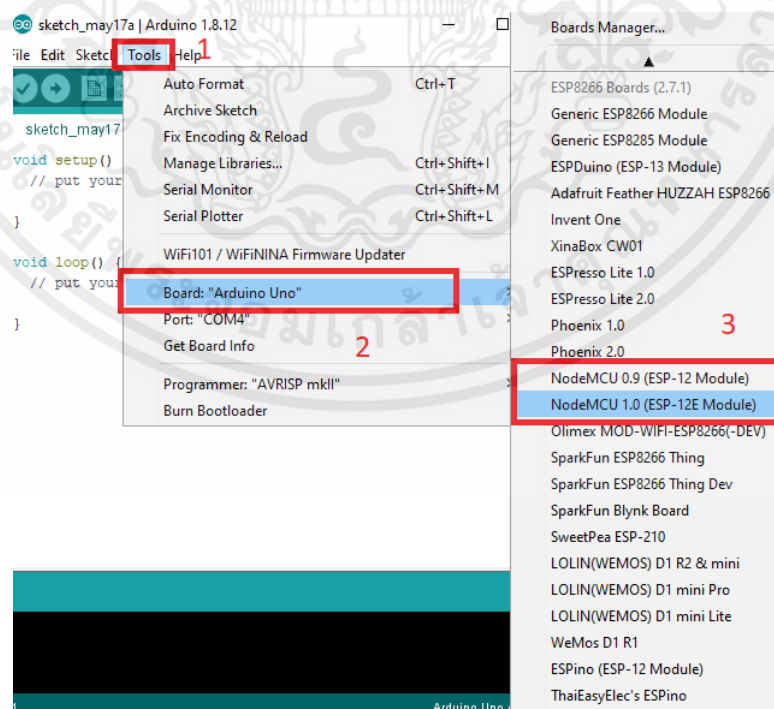
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 ชุดคำสั่งของ ESP8266 ที่ติดตั้งใน Arduino IDE เรียบร้อยแล้ว

ข้อควรระวัง ถ้าหากยังไม่ใส่ URLs ในขั้นตอนที่ 3 จะทำให้ไม่สามารถค้นหา ESP8266 ได้

6. การทำให้ ESP8266 กับ Arduino IDE รู้จักกันและทำงานร่วมกันได้ โดยไปที่เมนู Tools จากนั้นเลือก Board: "Arduino Uno" แล้วเลือก NodeMCU 0.9 (ESP-12 Module)/ NodeMCU 1.0 (ESP-12E Module)

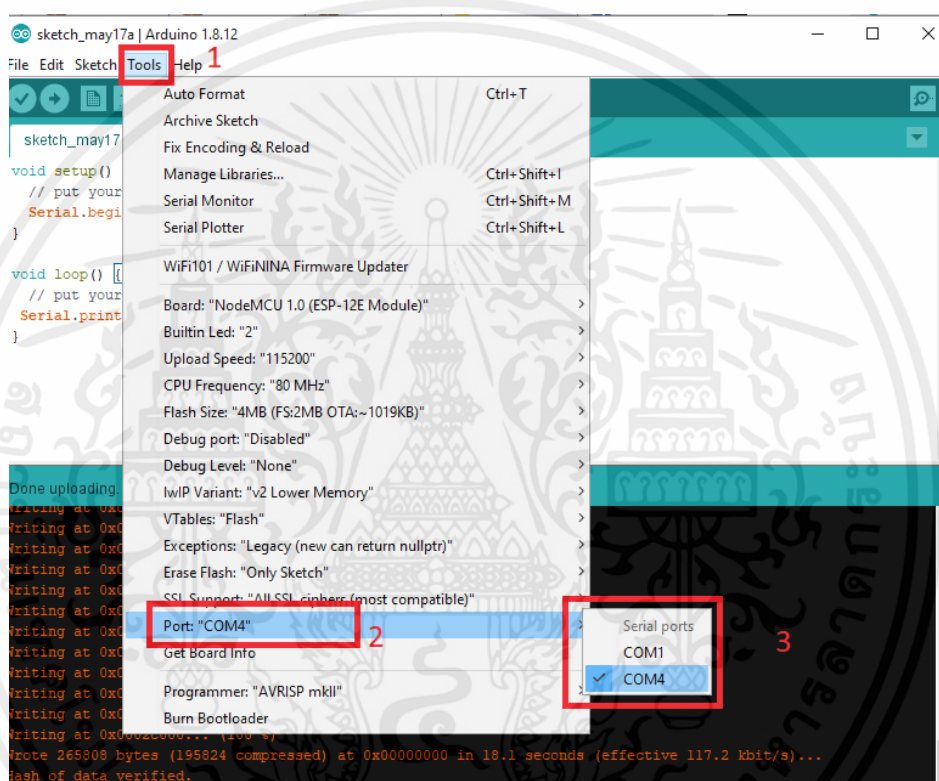


รูปที่ 2.7 ขั้นตอนการทำให้ Arduino IDE รู้จักกับ ESP8266

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. ทำการตั้งช่องทางการเชื่อมต่อหรือพอร์ต (Port) เพื่อให้สั่งการจาก Arduino IDE ไปยังบอร์ด ESP8266 ได้โดยเริ่มจากไปที่แถบเครื่องมือแล้วเลือกเมนู Tools จากนั้นเลือก Port: “COM4” แล้วจึงเลือก COM4 ดังแสดงในรูป 2.8 ถ้าหากไม่รู้ว่าเชื่อมต่อกันที่พอร์ตไหนสามารถไปดูได้ที่ Computer Management > System Tools > Device Manager > Ports (COM & LPT) ก็ จะรู้ได้ว่าเชื่อมต่อที่พอร์ตหมายเลขใด ดังแสดงไว้ในรูปที่ 2.9

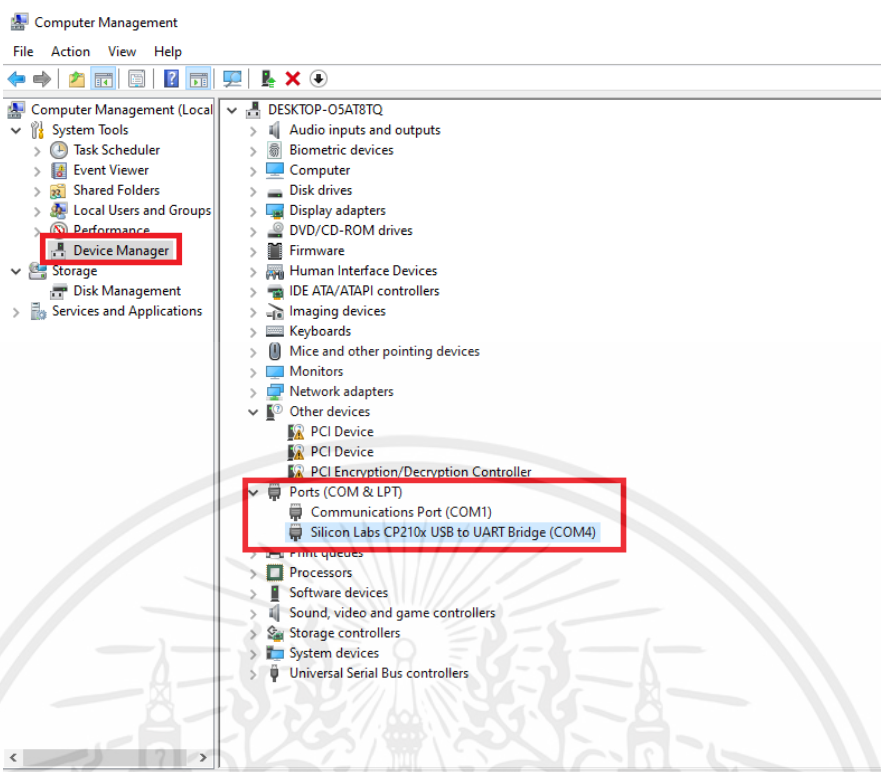
ข้อแนะนำ หากไม่ทราบที่กำลังต่อพอร์ตไหนให้ถอดสาย USB ของ ESP8266 ออกก่อน ต่อจากนั้นให้ต่อสาย USB อีกครั้งแล้วสังเกตดูว่าพอร์ตที่เชื่อมต่ออยู่ชื่อว่าอะไร



รูปที่ 2.8 การเชื่อมต่อพอร์ตระหว่าง Arduino IDE กับ ESP8266

ข้อควรรู้ ถ้าหาพอร์ตไม่เจอให้ลองตรวจทั้งสาย USB และบอร์ด ESP8266 เพราะอาจเกิดความเสียหายจากอย่างใดอย่างหนึ่งหรือทั้งสองอย่าง จึงทำให้เชื่อมต่อกันไม่ได้

ข้อแนะนำ หากไม่ทราบว่าจะไปที่ Computer Management ได้อย่างไรก็สามารถทำได้ดังนี้ เริ่มจากไปที่ Control Panel > Device Manager > Ports (COM & LPT) ทำอย่างนี้ก็สามารถรู้พอร์ตที่เชื่อมต่อได้เช่นเดียวกัน



รูปที่ 2.9 การบอกพอร์ตที่เชื่อมต่อระหว่าง Arduino IDE กับ ESP8266

### 2.1.2 การเขียนโค้ดในโปรแกรม Arduino IDE

โค้ด (Code) หรือจะเรียกว่าซอร์สโค้ด (Source Code) ที่ปรากฏอยู่บนพื้นที่เขียนโปรแกรม นั้นท่านผู้อ่านจะพบว่ามีฟังก์ชันอยู่ 2 ฟังก์ชันหลักคือ void setup() และ void loop() ซึ่งฟังก์ชัน void setup() จะมีไว้เพื่อกำหนดขาอินพุตและเอาต์พุตให้กับบอร์ดอีเอสพี8266 อีกฟังก์ชันหนึ่งคือ void loop() มีไว้เพื่อเขียนโค้ดสั่งการทำงานอย่างที่ต้องการ จากนั้นให้กด Upload Code



รูปที่ 2.10 หน้าต่างเริ่มต้นของโปรแกรม Arduino IDE

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในฟังก์ชัน void setup() จะพบคำสั่ง Serial.begin(9600); ปรากฏอยู่ คำสั่งนี้บ่งบอกว่า เซนเซอร์หรืออุปกรณ์อื่นๆ ที่เชื่อมต่อกับ ESP8266 จะมีความเร็วสัญญาณนาฬิกา (Clock) หรือความเร็วในการทำงานร่วมกันหรือความเร็วในการรับส่งข้อมูลที่ 9600 ครั้งต่อวินาที หากอุปกรณ์ที่เชื่อมต่อกับบอร์ดต้องการความเร็วในการรับส่งข้อมูลเท่าใด ก็ให้กำหนดได้ที่ตรงนี้ ดังรูปที่ 2.10 ส่วนรูปที่ 2.11 เป็นการทดสอบเขียนโค้ด และรูปที่ 2.12 คือผลลัพธ์ที่ได้จากการรันโปรแกรม

```

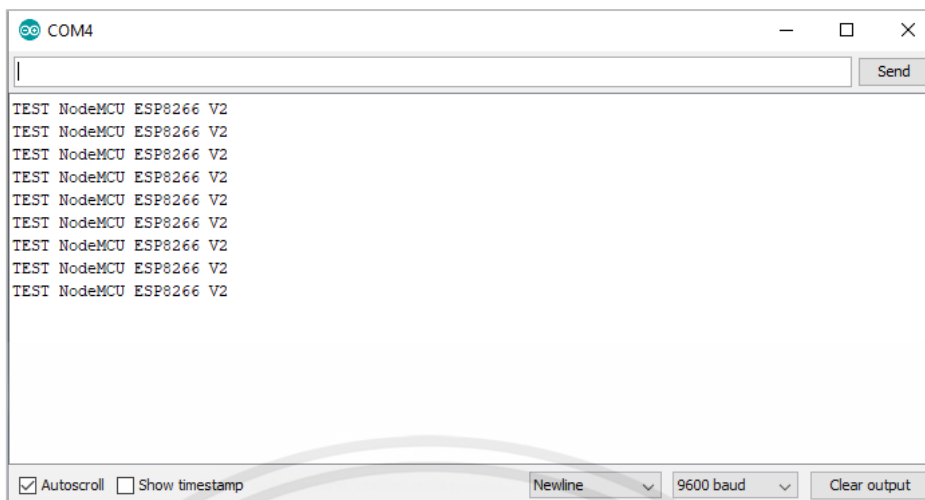
sketch_may17a | Arduino 1.8.12
File Edit Sketch Tools Help
sketch_may17a $
void setup() {
  Serial.begin(9600);
}

void loop() {
  Serial.println("TEST NodeMCU ESP8266 V2");
  delay(2000);
}

Done uploading.
Writing at 0x0001c000... (66 %)
Writing at 0x00020000... (75 %)
Writing at 0x00024000... (83 %)
Writing at 0x00028000... (91 %)
Writing at 0x0002c000... (100 %)
Wrote 265840 bytes (195868 compressed) at 0x00000000 in 18.1 seconds (effective 117.2 kbit/s)...
Hash of data verified.
Leaving...
Hard resetting via RTS pin...
8 NodeMCU 1.0 (ESP-12E Module) on COM4

```

รูปที่ 2.11 ตัวอย่างการเขียนโค้ดใน Arduino IDE



```

COM4
TEST NodeMCU ESP8266 V2
TEST NodeMCU ESP8266 V2
TEST NodeMCU ESP8266 V2
TEST NodeMCU ESP8266 V2
TEST NodeMCU ESP8266 V2
TEST NodeMCU ESP8266 V2
TEST NodeMCU ESP8266 V2
TEST NodeMCU ESP8266 V2
TEST NodeMCU ESP8266 V2
TEST NodeMCU ESP8266 V2
Autoscroll Show timestamp Newline 9600 baud Clear output

```

รูปที่ 2.12 ผลลัพธ์จากการรันโปรแกรม

### 2.1.3 การเชื่อมต่อและการสแกนหาชื่อ Wi-Fi

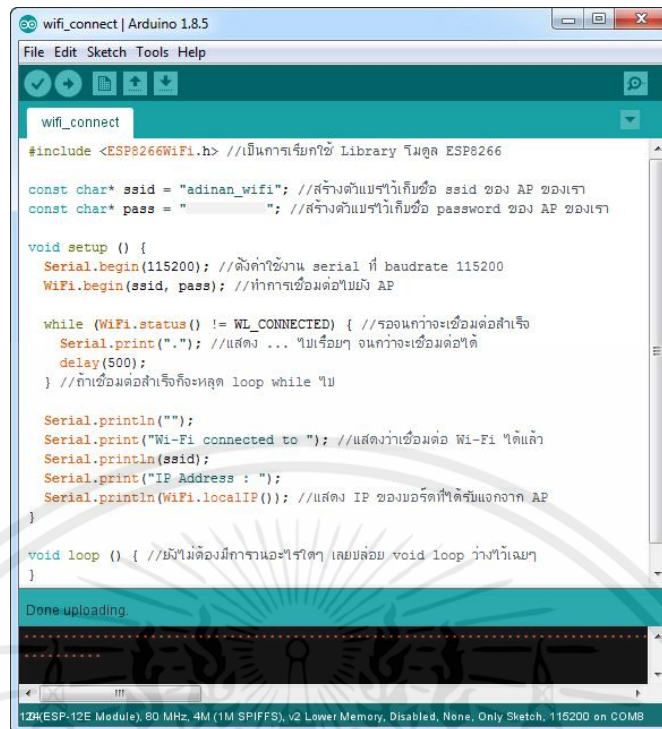
การเชื่อมต่อและสแกนหา Wi-Fi [2] ก่อนอื่นให้ทำการแนะนำตัว ESP8266 กับ Arduino IDE ให้รู้จักกันให้เรียบร้อยเสียก่อน (อ่านจากหัวข้อที่ 2.1.1 ได้) จากนั้นให้เขียนโค้ดทำการเชื่อมต่อและสแกนหา Wi-Fi ดังตัวอย่างที่จะอธิบายต่อไปนี้



รูปที่ 2.13 โครงสร้างของบอร์ด ESP8266

จากรูปที่ 2.14 คือตัวอย่างการเขียนโค้ดเพื่อให้ ESP8266 เชื่อมต่อกับ Wi-Fi ได้ ซึ่งผู้จัดทำจะขออธิบายให้พอเข้าใจคร่าวๆ ดังต่อไปนี้ เริ่มจากการเรียกใช้ไลบรารี ESP8266WiFi.h เพื่อให้เรียกใช้งานคำสั่งที่เกี่ยวข้องกับ Wi-Fi ได้ ต่อจากนั้นประกาศตัวแปร 2 ตัวที่มีชนิดของข้อมูลเป็น char นั่นคือตัวแปร ssid และตัวแปร pass เพื่อเก็บชื่อ WiFi และรหัสผ่าน (Password) สำหรับใช้งาน Wi-Fi ตัวนั้น ตามลำดับ ในฟังก์ชัน void setup() ให้ใส่คำสั่ง Serial.begin(115200) เพื่อกำหนดความเร็วในการสื่อสาร (Baud rate) ระหว่าง ESP8266 กับแอสซายน์ให้ค่าเท่ากับ 115200 บิต/วินาที ซึ่งเป็นความเร็วที่เข้ากันได้พอดี จากนั้นให้ใส่คำสั่ง WiFi.begin(ssid, pass) เพื่อทำการเชื่อมต่อ Wi-Fi ตัวที่ต้องการ ส่วนสำคัญจะจบที่ตรงนี้ ส่วนที่เหลือเป็นเพียงการแสดงค่าให้เห็นโดยง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

wifi_connect | Arduino 1.8.5
File Edit Sketch Tools Help

wifi_connect

#include <ESP8266WiFi.h> //เป็นการเรียกใช้ Library โมดูล ESP8266

const char* ssid = "adinan_wifi"; //สร้างตัวแปรไว้เก็บชื่อ ssid ของ AP ของเรา
const char* pass = " "; //สร้างตัวแปรไว้เก็บชื่อ password ของ AP ของเรา

void setup () {
  Serial.begin(115200); //ตั้งค่าใช้งาน serial ที่ baudrate 115200
  WiFi.begin(ssid, pass); //ทำการเชื่อมต่อไปยัง AP

  while (WiFi.status() != WL_CONNECTED) { //รอก่อนที่จะเชื่อมต่อสำเร็จ
    Serial.print("."); //แสดง ... ไปเรื่อยๆ จนกว่าจะเชื่อมต่อได้
    delay(500);
  } //ถ้าเชื่อมต่อสำเร็จก็จะหลุด loop while "ไป

  Serial.println("");
  Serial.print("Wi-Fi connected to "); //แสดงว่าเชื่อมต่อ Wi-Fi ได้แล้ว
  Serial.println(ssid);
  Serial.print("IP Address : ");
  Serial.println(WiFi.localIP()); //แสดง IP ของบอร์ดที่ได้รับแจกจาก AP
}

void loop () { //ยังไม่ต้องการทำอะไรเลย เลยปล่อย void loop วางไว้เฉยๆ
}

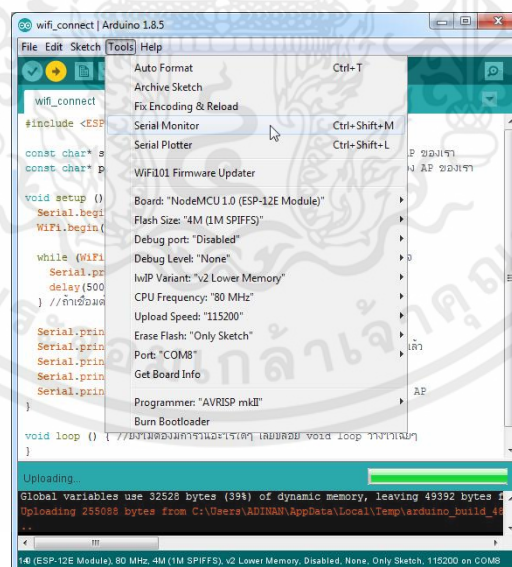
Done uploading.

12M(ESP-12E Module), 80 MHz, 4M (1M SPIFFS), v2 Lower Memory, Disabled, None, Only Sketch, 115200 on COM8

```

รูปที่ 2.14 ตัวอย่างโค้ดสำหรับเชื่อมต่อกับ Wi-Fi สำหรับ ESP8266

จากรูปที่ 2.15 จะเป็นการนำโค้ดเชื่อมต่อกับ Wi-Fi ที่เขียนเสร็จแล้วไปทำการอัปโหลดใส่ในตัว ESP8266 เพื่อให้ตัวบอร์ดทำงานเองโดยอัตโนมัติ โดยไปที่เมนู Tools > Burn Bootloader

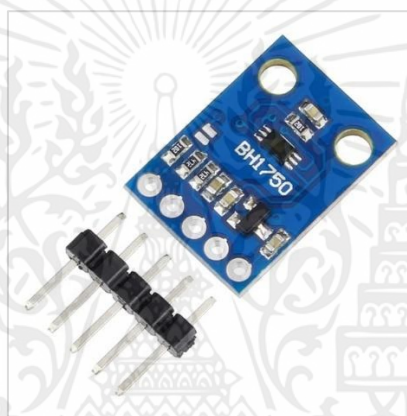


รูปที่ 2.15 การอัปโหลดโค้ดไปใส่ไว้ในตัว ESP8266

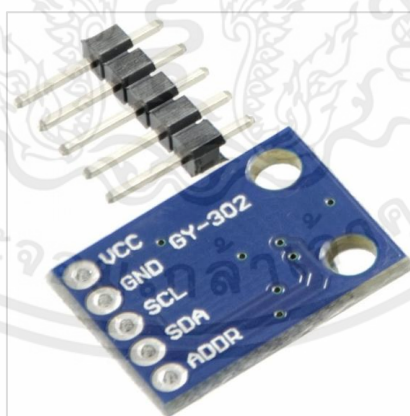
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.2 เซนเซอร์วัดความเข้มแสง BH1750

เซนเซอร์วัดความเข้มแสง รุ่น BH1750 [3] (Ambient Light Sensor Module BH1750) คืออุปกรณ์อิเล็กทรอนิกส์ที่เปลี่ยนแปลงค่าความต้านทาน หรือการนำกระแสไฟฟ้าที่ไหลผ่าน ตัวเซนเซอร์เมื่อมีแสงมาตกกระทบ โดยใช้ไอซีเบอร์ BH1750 มาเป็นตัวตรวจจับแสงและใช้การติดต่อกับบอร์ด ESP8266 ด้วยวิธี I<sup>2</sup>C ผลลัพธ์ที่ได้จากการวัดความเข้มแสงมีหน่วยเป็นลักซ์ (Lux) ทำให้นำข้อมูลที่ได้อไปใช้ประโยชน์ต่อได้ทันที โดยไม่ต้องพึ่งกระบวนการทางคณิตศาสตร์เพื่อแปลงหน่วย ภายในตัวตรวจจับแสงมีวงจรแปลงสัญญาณแอนะล็อกเป็นดิจิทัลซึ่งมีความละเอียด 16 บิต ทำให้ได้ข้อมูลความเข้มแสงแบบดิจิทัลที่มีความละเอียดแม่นยำมากพอที่จะนำไปสร้างเครื่องวัดความเข้มแสง หรือที่เรียกว่าลักซ์มิเตอร์ (Lux Meter) เซนเซอร์ตัวนี้สามารถวัดเข้มแสงได้ถึง 65535 ลักซ์



รูปที่ 2.16 ด้านหน้าของเซนเซอร์วัดความเข้มแสง BH1750



รูปที่ 2.17 ด้านหลังของเซนเซอร์วัดความเข้มแสง BH1750

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.2.1 ขาของเซนเซอร์วัดความเข้มแสง BH1750

ตารางที่ 2.1 ตารางข้อมูลเกี่ยวกับขาของเซนเซอร์วัดความเข้มแสง BH1750

หมายเลขขา	ชื่อเรียก	คำอธิบาย
1	VCC	ขารับแรงดันไฟฟ้าจากแหล่งจ่ายแรงดันไฟฟ้ากระแสตรง เพื่อจ่ายแรงดันไฟฟ้าให้เซนเซอร์ทำงาน ซึ่งรับได้ในช่วง 3-5 V
2	GND	ขากราวด์สำหรับต่อเข้ากับกราวด์ของวงจร
3	SCL	ขาสำหรับส่งสัญญาณนาฬิกาที่ใช้ในการส่งข้อมูลแบบ I <sup>2</sup> C
4	SDA	ขาที่ใช้ในการถ่ายโอนข้อมูลผ่านการสื่อสารแบบ I <sup>2</sup> C
5	ADDR	ขาสำหรับบอกแอดเดรสของเซนเซอร์ เมื่อบอร์ด ESP8266 มีการเชื่อมต่อกับเซนเซอร์วัดความเข้มแสงตั้งแต่ 2 โมดูล

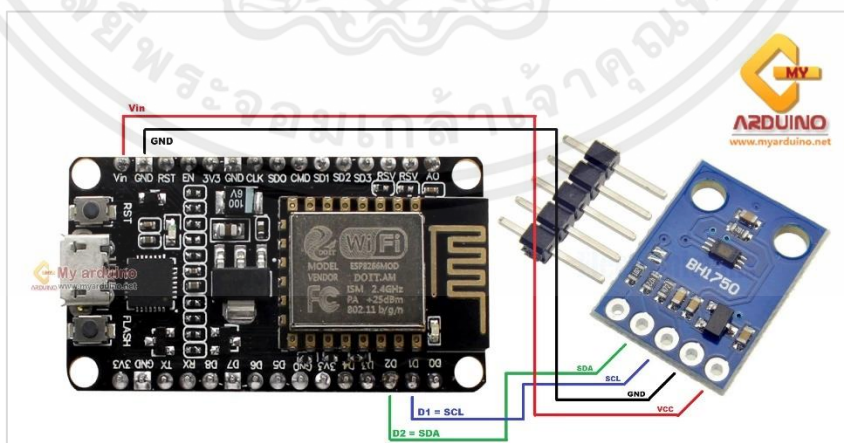
## 2.2.2 คุณสมบัติของของเซนเซอร์วัดความเข้มแสง BH1750

1. รับแรงดันไฟฟ้ากระแสตรงในช่วง 2.4-3.6 V
2. ใช้กระแสไฟ 0.2 mA
3. ค่าคลาดเคลื่อน  $\pm 20\%$
4. มีตัวแปลงข้อมูลแบบแอนะล็อกให้เป็นดิจิทัล (A/D Converter)

## 2.2.3 วิธีต่อเซนเซอร์วัดความเข้มแสง BH1750 ให้เข้ากับ ESP8266

รายละเอียดการต่อวงจรระหว่าง ESP8266 กับเซนเซอร์ BH1750 [4] มีดังต่อไปนี้

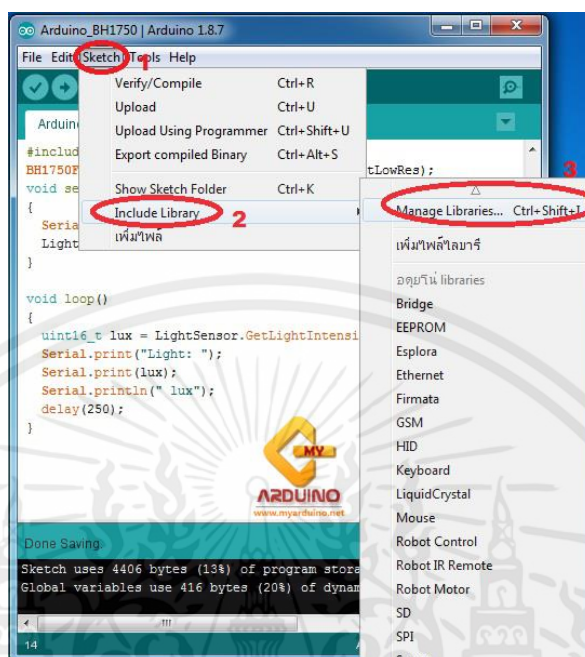
1. ต่อขา VCC เข้ากับแหล่งจ่ายไฟกระแสตรงด้วยแรงดันไฟฟ้า 5 V
2. ต่อขา GND เข้ากับกราวด์ของวงจร
3. ต่อขา SCL เข้ากับขาหมายเลข D1 ของ ESP8266
4. ต่อขา SDA เข้ากับขาหมายเลข D2 ของ ESP8266



รูปที่ 2.18 การเชื่อมต่อระหว่าง ESP8266 กับเซนเซอร์วัดความเข้มแสง BH1750

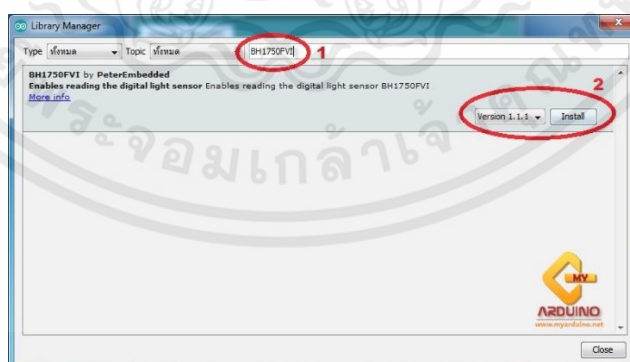
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. เปิดโปรแกรม Arduino IDE จากนั้นให้ดาวน์โหลดไลบรารีเซ็นเซอร์วัดความเข้มแสง BH1750 โดยเข้าไปที่ Sketch > Include Library > Manage Library ตามรูปที่ 2.19



รูปที่ 2.19 การดาวน์โหลดไลบรารีของเซ็นเซอร์วัดความเข้มแสง BH1750

6. ให้พิมพ์คำว่า BH1750FVI ในช่องค้นหาของ Library Manager แล้วกด Install เป็นการติดตั้งไลบรารีของเซ็นเซอร์วัดความเข้มแสง BH1750 เสร็จสิ้น ดังรูปที่ 2.20



รูปที่ 2.20 การติดตั้งไลบรารีของเซ็นเซอร์วัดความเข้มแสง BH1750

7. ต่ออุปกรณ์ตามรูปที่ 2.18 จากนั้นให้เขียนโค้ดตามตัวอย่างที่แสดงไว้ในรูปที่ 2.21 เมื่อเสร็จแล้วให้อัปโหลดโค้ดเข้าไปใน ESP8266

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1 #include <BH1750FVI.h>
2
3 BH1750FVI LightSensor(BH1750FVI::k_DevModeContLowRes);
4 void setup()
5 {
6   Serial.begin(9600);
7   LightSensor.begin();
8 }
9
10 void loop()
11 {
12   uint16_t lux = LightSensor.GetLightIntensity();
13   Serial.print("Light: ");
14   Serial.print(lux);
15   Serial.println(" lux");
16   delay(250);
17 }

```

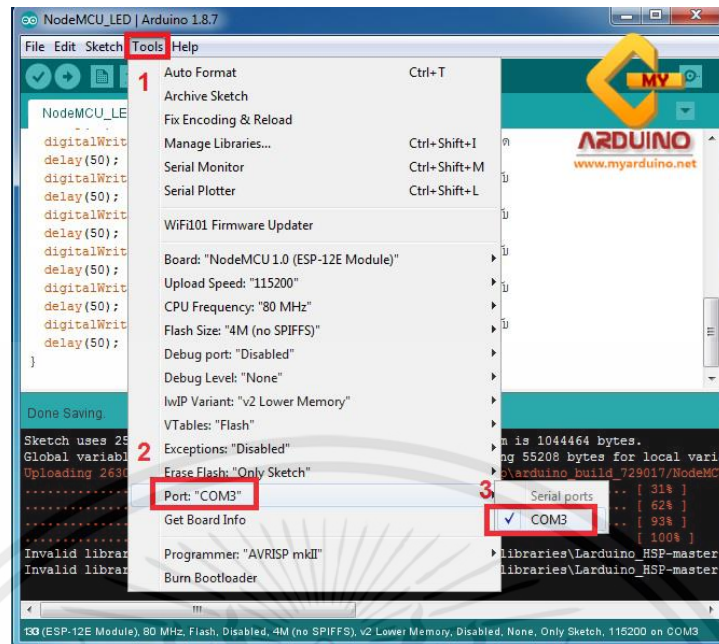
รูปที่ 2.21 ตัวอย่างโค้ดที่ทำให้ ESP8266 ทำงานร่วมกับ BH1750

8. เลือกบอร์ด ESP8266 ที่ต้องการจะอัปโหลดตามที่แสดงในรูปที่ 2.22



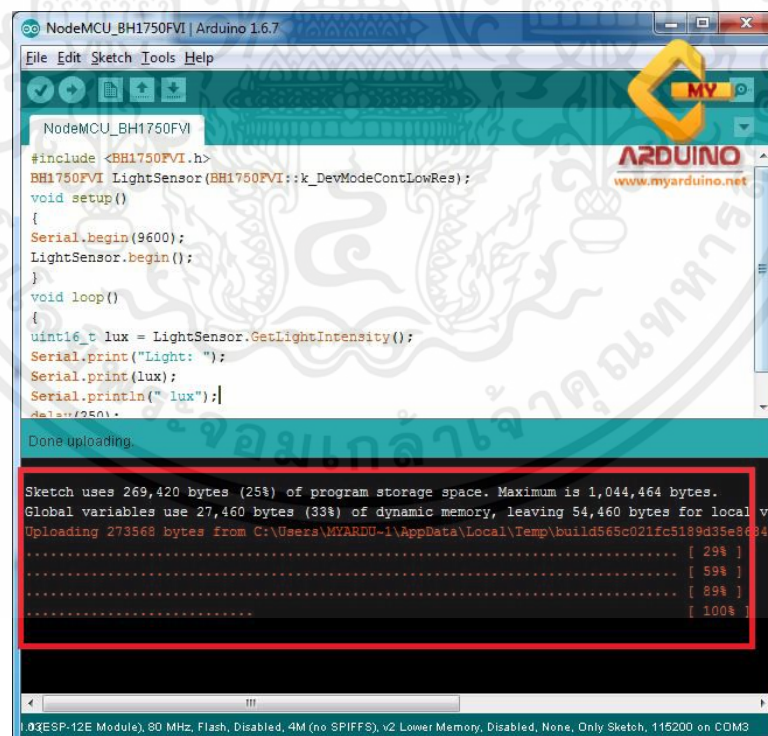
รูปที่ 2.22 การเลือกรุ่นของบอร์ด ESP8266 เพื่อการอัปโหลด

9. เลือกพอร์ตที่บอร์ด ESP8266 สำหรับการเชื่อมต่อแล้วกดอัปโหลดโค้ด



รูปที่ 2.23 การเชื่อมต่อระหว่าง ESP8266 กับ BH1750 ผ่านพอร์ตที่ต้องการ

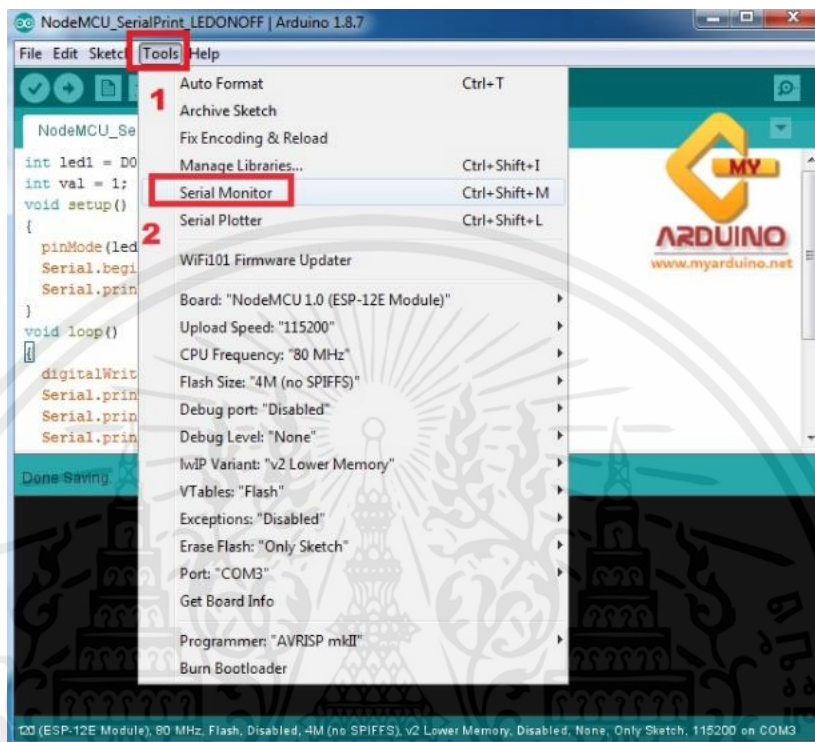
10. เมื่ออัปโหลดสำเร็จแล้ว จะได้ผลลัพธ์ตามรูปที่ 2.24



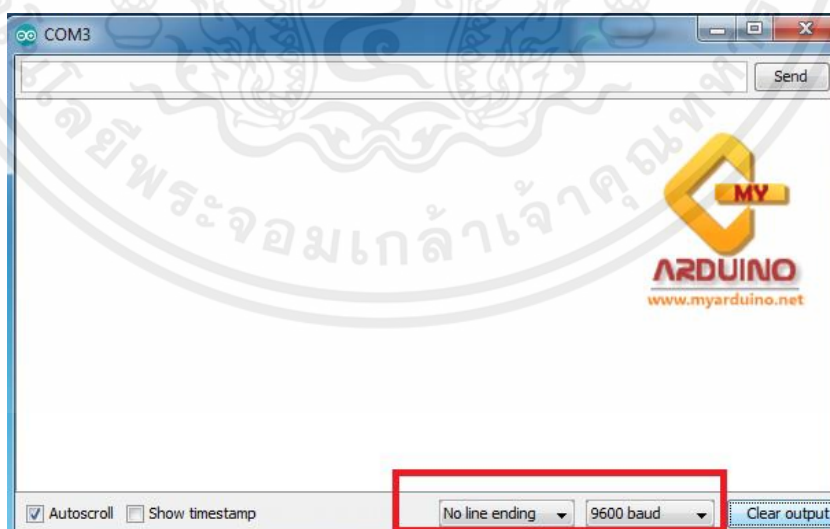
รูปที่ 2.24 ผลลัพธ์จากการอัปโหลดโค้ดเพื่อใช้งาน BH1750 เข้าไปใน ESP8266

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. ให้เปิด Serial Monitor ขึ้นมาแล้วกำหนดอัตราการส่งข้อมูลให้ตรงกับโค้ดที่จะอัปโหลดเข้าไปใน ESP8266 ในการนี้จะต้องกำหนดให้ Baud rate มีค่าเท่ากับ 9600 บิต วิธีการเปิด Serial Monitor และกำหนด Baud rate เป็นไปตามรูปที่ 2.25 และ 2.26 ตามลำดับ



รูปที่ 2.25 การกำหนด Serial Monitor ให้กับบอร์ด ESP8266



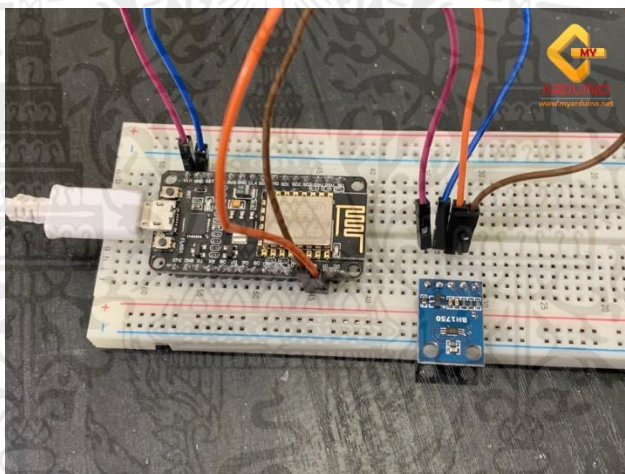
รูปที่ 2.26 การกำหนด Baud rate ให้ ESP8266 สื่อสารกับ BH1750 ได้ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

12. จากรูปที่ 2.27 หากเปิด Serial Monitor ผู้อ่านจะเห็นการแสดงค่าความเข้มแสงที่วัดได้จากเซนเซอร์วัดความเข้มแสง BH1750



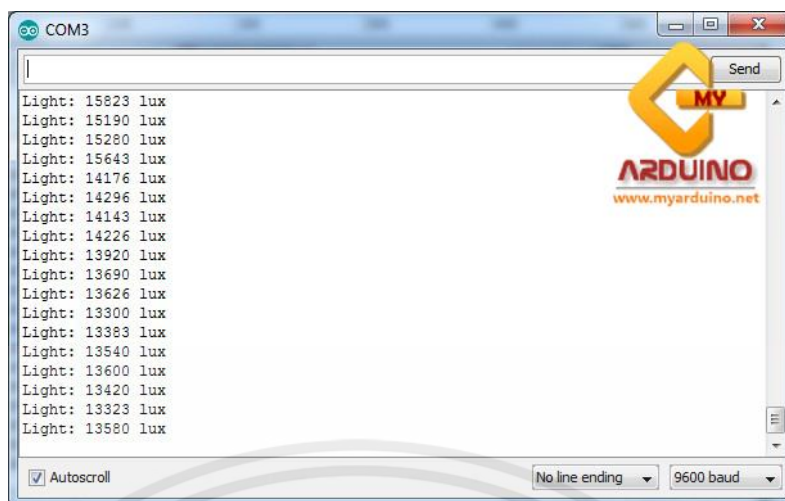
รูปที่ 2.27 ค่าที่ได้จากการวัดค่าความเข้มแสงที่ปรากฏบน Serial Monitor



รูปที่ 2.28 ตัวอย่างจริงในการเชื่อมต่อระหว่าง ESP8266 กับ BH1750

13. หากนำไฟฉายมาส่องที่ BH1750 ค่าความเข้มแสงก็จะมากขึ้นตามรูปที่ 2.29

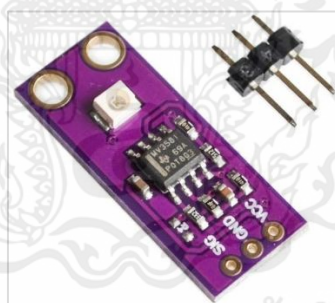
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.29 การวัดค่าความเข้มแสงของ BH1750 เมื่อมีความเข้มแสงสูงขึ้น

## 2.3 เซนเซอร์วัดความเข้มรังสีอัลตราไวโอเล็ต GUVA S12SD

เซนเซอร์วัดความเข้มรังสีอัลตราไวโอเล็ต รุ่น CJMCU GUVA S12SD [5] หรือมีชื่อเรียกภาษาอังกฤษว่า Ultraviolet Ray Light Intensity Sensor CJMCU GUVA S12SD ซึ่งผู้จัดทำจะขอเรียกอย่างย่อว่า เซนเซอร์วัดความเข้มรังสียูวีหรือเซนเซอร์ยูวี ค่าที่เซนเซอร์ยูวีตัวนี้วัดได้จะเป็นแรงดันไฟฟ้ากระแสตรง ( $V_{DC}$ ) ซึ่งจัดว่าเป็นผลลัพธ์แบบแอนะล็อก และเหมาะกับการนำไปเก็บข้อมูลรังสียูวีได้ตลอดเวลาในแต่ละวัน รวมทั้งใช้งานร่วมกับ ESP8266 หรือ Arduino ได้อีกด้วย



รูปที่ 2.30 ตัวอย่างเซนเซอร์ยูวี GUVA S12SD

### 2.3.1 คุณสมบัติของเซนเซอร์ยูวี GUVA S12SD

1. มีขนาดกว้าง 11 มิลลิเมตร และยาว 27 มิลลิเมตร (mm)
2. ต้องการแรงดันไฟฟ้ากระแสตรง 2.5-5 V
3. ต้องการกระแสไฟฟ้าในการทำงานอยู่ที่ระดับไมโครแอมแปร์ ( $\mu A$ )
4. ตรวจวัดความถี่ของแสงได้ในช่วง 240-370 นาโนเมตร (nm)
5. มุมในการรับแสงกว้าง 130 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ใช้ไดโอดไวแสงชนิด Schottky ในการแปลงความเข้มรังสีให้เป็นแรงดันไฟฟ้า  
กระแสตรง

7. เอาต์พุตที่เป็นแรงดันไฟฟ้ากระแสตรงอยู่ในช่วง 0-1 V โดยแบ่งเป็นเลขดัชนี  
ความเข้มรังสีอยู่ที่ 0-10 และ 11+

UV Index	0	1	2	3	4	5
Vout(mV)	<50	227	318	408	503	606
UV Index	6	7	8	9	10	11+
Vout(mV)	696	795	881	976	1079	1170+

รูปที่ 2.31 รูปการเปรียบเทียบค่าแรงดันไฟฟ้ากระแสตรงกับเลขดัชนีของความเข้มรังสี

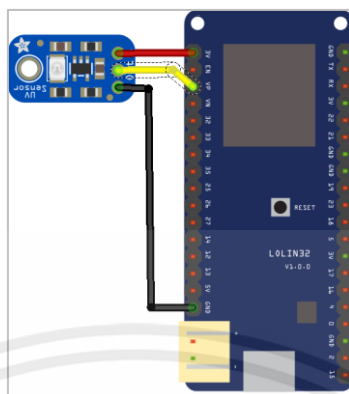
จากรูปที่ 2.31 เป็นการเทียบให้เห็นว่าเอาต์พุตที่ได้จากเซนเซอร์ตัวนี้ซึ่งเป็นแรงดันไฟฟ้า  
กระแสตรงกับเลขดัชนีที่บ่งบอกความเข้มของรังสี ซึ่งพบว่าหากแรงดันไฟฟ้าที่ได้มีค่ามากขึ้น นัน  
ก็หมายความว่าความเข้มรังสีมีค่ามากขึ้นด้วย เมื่อเลขดัชนีรังสีมีค่ามากขึ้นก็จะมีอันตราย  
ต่อร่างกายของมนุษย์เพิ่มขึ้นเช่นกัน

ความเสี่ยง	เลขดัชนี รังสี	ข้อควรปฏิบัติ
ต่ำ ↓ สูง	0-2	ให้ปกป้องศีรษะและดวงตา
	3-5	ให้ปกป้องศีรษะและดวงตาและใช้ครีมกันแดดค่า SPF ไม่สูง
	6-7	ให้ปกป้องศีรษะและดวงตา ใช้ครีมกันแดดที่มีค่า SPF สูง และไม่อยู่กลางแจ้งเป็นเวลานาน
	8-10	ให้ปกป้องศีรษะและดวงตา ใช้ครีมกันแดดที่มีค่า SPF สูง และหลีกเลี่ยงการอยู่กลางแจ้ง
	11+	ไม่ควรออกนอกอาคารหรือไม่ควรอยู่กลางแจ้ง

รูปที่ 2.32 เลขดัชนีความเข้มรังสีกับข้อปฏิบัติตัวเพื่อปกป้องอันตรายจากรังสี

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.2 วิธีการเชื่อมต่อเซนเซอร์ยูวี GUYA S12SD ให้เข้ากับ ESP8266



รูปที่ 2.33 ตัวอย่างการต่อวงจรเซนเซอร์ยูวีกับ ESP8266

### 2.3.3 ขาของเซนเซอร์ยูวี GUYA S12SD

ตารางที่ 2.2 ข้อมูลเกี่ยวกับขาของเซนเซอร์วัดความเข้มแสง BH1750

ขาของ ESP8266	ขาของเซนเซอร์ยูวี GUYA S12SD
Vcc	Vcc
GND	GND
A0	SIG

จากตารางที่ 2.2 เป็นการอธิบายให้เห็นอย่างชัดเจนว่าขาของ ESP8266 กับขาของเซนเซอร์ยูวี GUYA S12SD สมควรต่อขาใดของอุปกรณ์ตัวหนึ่งเข้ากับขาใดของอุปกรณ์อีกตัวหนึ่ง

### 2.3.4 ตัวอย่างโค้ด

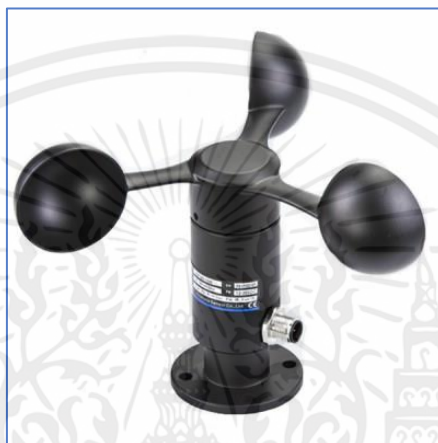
```
UV-Sensor-2.cpp
1  #include<GUYA-S12SD.h>
2
3  GUYAS12SD uv(A0);
4
5  void setup()
6  {
7    Serial.begin(9600);
8  }
9  void loop()
10 {
11   float sensorVoltage;
12   float sensorValue;
13
14   sensorValue = analogRead(36);
15   sensorVoltage = (sensorValue/4095)*3.3;
16   Serial.print("Sensor reading = ");
17   Serial.print(sensorValue);
18   Serial.println(" ");
19   Serial.print("Sensor voltage = ");
20   Serial.print(sensorVoltage);
21   Serial.println("V");
22   delay(1000);
23 }
```

รูปที่ 2.34 ตัวอย่างโค้ดที่ทำให้ ESP8266 ทำงานร่วมกับเซนเซอร์ยูวี GUYA S12SD ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.4 เซนเซอร์วัดความเร็วลม (Wind Speed Sensor)

เซนเซอร์วัดความเร็วลม [6] เป็นเซนเซอร์ที่ใช้สำหรับวัดความเร็วลมซึ่งให้เอาต์พุตที่มีขนาดแรงดันไฟฟ้า 0-5 V<sub>DC</sub> ตัวเซนเซอร์มีลักษณะเป็นถ้วย 3 ถ้วยทรงครึ่งวงกลม ซึ่งมาพร้อมกับสายเคเบิลยาว 2 เมตร และตัวเซนเซอร์ผลิตมาจากอลูมิเนียมอัลลอย (Aluminum Alloy) สำหรับวัดลมผลิตจากอลูมิเนียมอัลลอยคุณภาพสัญญาณเอาต์พุต 0-5V ใช้ไฟอินพุต 12-24V เครื่องวัดความเร็วลมพร้อมสายเคเบิลยาว 2 เมตร



รูปที่ 2.35 เซนเซอร์วัดความเร็วลม

### 2.4.1 ข้อมูลทางเทคนิคเซนเซอร์วัดความเร็วลม

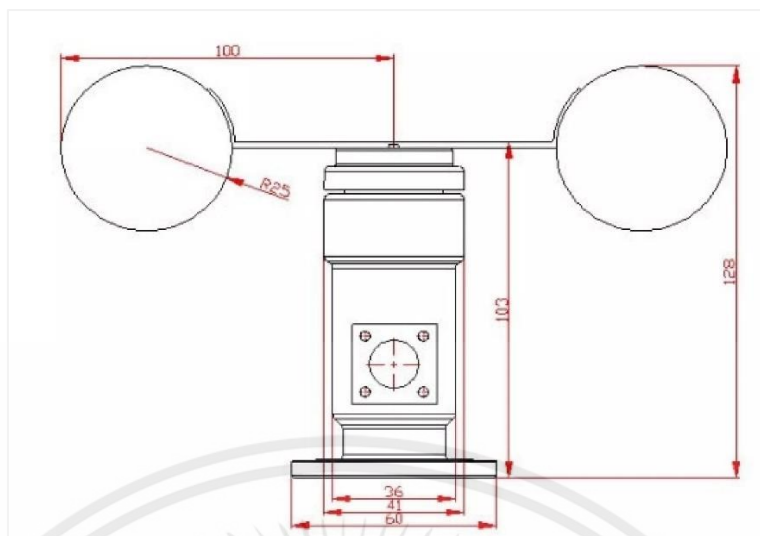
1. บอเอาต์พุตเป็นแรงดันไฟฟ้าขนาด 0-5 V<sub>DC</sub>
2. มีกระแสไฟฟ้้ออกทางด้านเอาต์พุตขนาด 4-20 mA
2. รับอินพุตเป็นแรงดันไฟฟ้าขนาด 9-24 V<sub>DC</sub> เพื่อให้เซนเซอร์ทำงาน
3. วัดความเร็วลมได้ตั้งแต่ 0-30 m/s (เมตร/วินาที)
4. มีความละเอียดในการวัดความเร็วลมที่ 0.1 m/s
5. ทำงานได้ในช่วงอุณหภูมิ -40 ถึง 80°C (องศาเซลเซียส)

### 2.4.2 คุณสมบัติเซ็นเซอร์วัดลม

1. ขนาดเล็กพกพาง่าย ติดตั้งง่าย รูปลักษณะที่น่าสนใจ
2. ทนทานต่อการกัดกร่อนและทนต่อสภาพอากาศได้ดี
3. มีความแม่นยำสูงช่วงการวัดกว้างเสถียรภาพที่ดี
4. ใช้พลังงานน้อย ป้องกันการรบกวนที่ดี และใช้งานได้ยาวนาน

### 2.4.3 ภาพวาดมิติโครงสร้างของเซนเซอร์วัดความเร็วลม

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

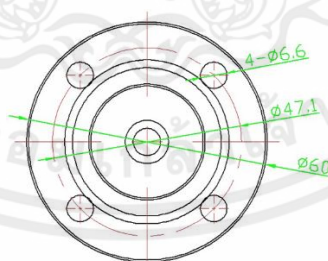


รูปที่ 2.36 โครงสร้างของเซนเซอร์วัดความเร็วลม

จากรูปที่ 2.36 คือโครงสร้างของเซนเซอร์วัดความเร็วลมที่มองจากด้านข้าง โดยมีหน่วยวัดระยะทั้งหมดเป็นมิลลิเมตร (mm) จะเห็นได้ว่าตัวเซนเซอร์จะมีรัศมี 25 จากจุดกึ่งกลางของด้านบนไปจนสุดขอบแกนของเซนเซอร์จะมีระยะ 100 ความสูงจากฐานถึงขอบเซนเซอร์ด้านบนมีระยะ 128 และฐานของเซนเซอร์มีระยะ 60

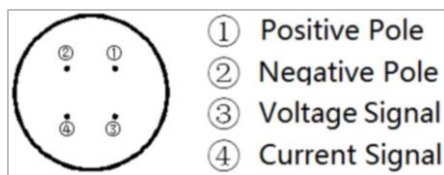
#### 2.4.4 การติดตั้งเซนเซอร์วัดความเร็วลม

ควรติดตั้งเซนเซอร์ในแนวตั้งเพื่อให้เซนเซอร์วัดความเร็วลมได้อย่างแม่นยำ การติดตั้งเซนเซอร์ควรยึดถือตามแบบในรูปที่ 2.37 เส้นผ่านศูนย์กลางของหน้าแปลนติดตั้งอยู่ที่ 60 มิลลิเมตรและช่องใส่เนื้อยึดสี่ช่อง ซึ่งมีขนาดช่องอยู่ที่ช่องละ 6.6 มิลลิเมตร



รูปที่ 2.37 แบบที่ตำแหน่งฐานของเซนเซอร์สำหรับการติดตั้ง

#### 2.4.5 การต่อขาของเซนเซอร์วัดความเร็วลม

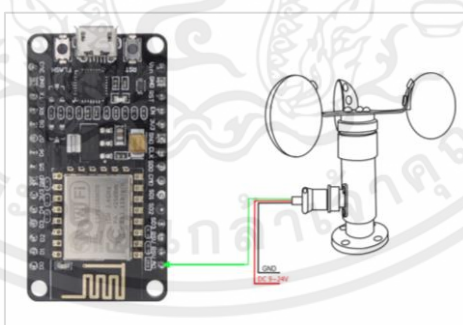


รูปที่ 2.38 สถานะขาของเซนเซอร์วัดความเร็วลม

จากรูปที่ 2.38 ขาที่ 1 กับ 2 ของเซนเซอร์วัดความเร็วลมจะต่อเข้ากับขา  $V_{CC}$  และ GND ของบอร์ด ESP8266 ตามลำดับ สำหรับขาที่ 3 จะต่อเข้ากับขา A0 และขาที่ 4 จะต่อเข้ากับขา SC ของ ESP8266 หรือดูเทียบได้จากตารางที่ 2.3 หรือรูปที่ 2.39

ตารางที่ 2.3 การเชื่อมต่อขาของเซนเซอร์วัดความเร็วลมให้เข้ากับขาของ ESP8266

สีของสายไฟ/ หมายเลขขา	สถานะขาของเซนเซอร์ วัดความเร็วลม	การเชื่อมต่อขา กับ ESP8266
สีแดง (Red)/1	$V_{CC}$	$V_{CC}$
สีดำ (Black)/2	GND	GND
สีเหลือง (Yellow)/3	Voltage Signal	A0
สีน้ำเงิน (Blue)/4	Current Signal	SC



รูปที่ 2.39 การต่อใช้ขาของเซนเซอร์วัดความเร็วลมกับ ESP8266

#### 2.4.6 ความสัมพันธ์ระหว่างเอาต์พุตกับความเร็วลม

ผลลัพธ์การวัดความเร็วลมของเซนเซอร์จะแสดงออกมาเป็นแรงดันไฟฟ้ากระแสตรง ซึ่งจะเทียบเป็นความเร็วลมได้ดังที่แสดงข้อมูลไว้ในตารางที่ 2.4 หากวัดแรงดันไฟฟ้ากระแสตรงได้มากขึ้น ก็แสดงว่าความเร็วลมก็จะเร็วขึ้นเช่นกัน หรือเรียกว่าแปรผันตามกันก็ได้

ตารางที่ 2.4 ความสัมพันธ์ของแรงดันไฟฟ้าด้านเอาต์พุตและความเร็วลม

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความเร็วลม (m/s)	แรงดันไฟฟ้า (V)	ความเร็วลม (m/s)	แรงดันไฟฟ้า (V)	ความเร็วลม (m/s)	แรงดันไฟฟ้า (V)
1	0.17	11	1.83	21	3.5
2	0.33	12	2	22	3.67
3	0.5	13	2.17	23	3.83
4	0.67	14	2.33	24	4
5	0.83	15	2.5	25	4.17
6	1	16	2.67	26	4.33
7	1.17	17	2.83	27	4.5
8	1.33	18	3	28	4.67
9	1.5	19	3.17	29	4.83
10	1.67	20	3.33	30	5

#### 2.4.7 ตัวอย่างโค้ด

```

WindSpeedSensor.cpp
1  #include<ESPWifi8266.h>
2
3  void setup()
4  {
5      Serial.begin(9600);
6  }
7  void loop()
8  {
9      int sensorValue = analogRead(A0);
10     float outvoltage = sensorValue*(5.0/1023.0);
11     Serial.print("outvoltage = ");
12     Serial.print(outvoltage);
13     Serial.println("V");
14     int Level = 6*outvoltage
15     Serial.print("wind speed is ");
16     Serial.print(Level);
17     Serial.println(" level now");
18     Serial.println();
19     delay(500);
20 }

```

รูปที่ 2.40 โค้ดตัวอย่างการใช้งานเซนเซอร์วัดความเร็วลมซึ่งต่อกับ ESP8266

## 2.5 เซนเซอร์วัดปริมาณน้ำฝน (Raindrop Sensor)

เซนเซอร์วัดปริมาณน้ำฝน รุ่น CM2000 ClimeMET [8] เป็นอุปกรณ์ที่สามารถบ่งชี้ได้ว่าในบริเวณที่ฝนตกนั้น มีน้ำฝนตกลงมาสะสมสูงสักเท่าไร (คิดที่น้ำฝนไม่กระจายและระบายออกไป) หน่วยที่วัดได้จากเซนเซอร์คือหน่วย มิลลิเมตร เช่น ปริมาณน้ำฝน 16.3 มิลลิเมตร ก็หมายความว่า มีน้ำขังสะสมอยู่ภาชนะสูง 1.63 มิลลิเมตร ซึ่งภาชนะในที่นี้หมายถึงพื้นที่รองรับน้ำฝน

### 2.5.1 คุณสมบัติของของเซนเซอร์วัดปริมาณน้ำฝน

1. มีความแม่นยำมากในการตรวจจับปริมาณน้ำฝน
2. มีความแม่นยำในการส่งค่าที่วัดได้ไปยังไมโครคอนโทรลเลอร์
3. มีความแข็งแรงสูงทนต่อสภาพอากาศและทนน้
4. เป็นชิ้นงานที่ใช้สำหรับสถานีตรวจอากาศระดับมืออาชีพ

### 2.5.2 หลักการทำงานของเซนเซอร์วัดปริมาณน้ำฝน

มาตรวัดหรือเรียกว่า เกจ (Gauge) วัดปริมาณน้ำฝนตัวนี้มีลักษณะกระดานหกแบบ เหน้าทิ้งเองหรือจะเรียกว่าแบบทิป (Tip) ก็ได้ ดังรูปที่ 2.41 กระดานหกแต่ละด้านมีความจุน้ำฝน 0.28 มิลลิเมตรโดยประมาณ (ในความจริงคือ 0.2792 มิลลิเมตร) เมื่อเกิดการกระดกจะทำให้เกิดการปิดหน้าสัมผัสชั่วขณะหนึ่ง ในจังหวะนั้นเซนเซอร์จะบันทึกค่าด้วยตัวนับแบบดิจิทัล แล้วจะส่งค่าที่วัดได้ไปยังไมโครคอนโทรลเลอร์แบบขัดจังหวะ (อินเทอร์รัปต์: Interrupt)



รูปที่ 2.41 ลักษณะภายในของเซนเซอร์วัดปริมาณน้ำฝน

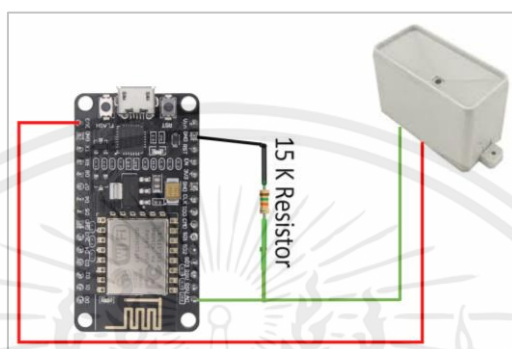


รูปที่ 2.42 เซนเซอร์วัดปริมาณน้ำฝน

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.5.3 การต่อใช้งานของเซนเซอร์วัดปริมาณน้ำฝน

การเชื่อมต่อวงจรจากเซนเซอร์วัดปริมาณน้ำฝนให้เข้ากับ ESP8266 [9] เริ่มจากสายไฟสีแดงของเซนเซอร์วัดปริมาณน้ำฝนจะต่อเข้ากับขา V<sub>CC</sub> ของ ESP8266 สายไฟสีเขียวของเซนเซอร์จะเชื่อมต่อกับขา A0 ของบอร์ด ESP8266 และต่อร่วมกับตัวต้านทานขนาด 15 k $\Omega$  แบบพูลดาวน์ (Pull-Down) ดังที่ปรากฏในรูปที่ 2.43 และข้อมูลในตารางที่ 2.5



รูปที่ 2.43 การต่อใช้งานของเซนเซอร์วัดปริมาณน้ำฝน

ตารางที่ 2.5 สถานะของการต่อใช้งานเซนเซอร์วัดปริมาณน้ำฝน

สีของสายไฟ	การเชื่อมต่อกับ ESP8266
แดง	V <sub>CC</sub>
เขียว	A0

```

RainDropSensor.cpp
1 #include "RTClib.h"
2 #include <Wire.h>
3 #define RainPin A0
4
5 bool bucketPositionA = false;
6 const double bucketAmount = 0.01610595;
7 double dailyRain = 0.0;
8 double hourlyRain = 0.0;
9 double dailyRain_till_LastHour = 0.0;
10 bool first;
11 RTC_Millis rtc;
12
13 void reedSwitch_ISR()
14 {
15     bucketPositionA = true;
16     dailyRain += bucketAmount;
17 }
18 void setup()
19 {
20     Serial.begin(9600);
21     rtc.begin(DateTime(__DATE__, __TIME__));
22     pinMode(RainPin, INPUT_PULLUP);
23     attachInterrupt(digitalPinToInterrupt(RainPin), reedSwitch_ISR, FALLING);
24     delay(4000);
25     Serial.println("Ready!!!");
26 }

```

รูปที่ 2.44 ตัวอย่างโค้ดการใช้งานเซนเซอร์วัดปริมาณน้ำฝนที่เชื่อมต่อกับ ESP8266

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.5.4 ตัวอย่างโค้ด (ต่อ)

```

27 void loop(void)
28 {
29     DateTime now = rtc.now();
30     if ((bucketPositionA==true)&&(digitalRead(RainPin)==LOW))
31         bucketPositionA = false;
32     if(now.second() != 0)
33         first = true;
34     if(now.second() == 0 && first == true)
35     {
36         hourlyRain = dailyRain - dailyRain_till_LastHour;
37         dailyRain_till_LastHour = dailyRain;
38         Serial.println();
39         Serial.print(now.hour());
40         Serial.print(":");
41         Serial.print(now.minute());
42         Serial.print(": Total Rain for the day = ");
43         Serial.print(dailyRain,8);
44         Serial.println(" inches");
45         Serial.println();
46         Serial.print("    : Rain in last hour = ");
47         Serial.print(hourlyRain,8);
48         Serial.println(" inches");
49         Serial.println();
50         first = false;
51     }
52 }

```

รูปที่ 2.45 ตัวอย่างโค้ดการใช้งานเซนเซอร์วัดปริมาณน้ำฝนที่เชื่อมต่อกับ ESP8266 (ต่อ)

## 2.6 เซนเซอร์วัดอุณหภูมิและความชื้น (DHT11 Temperature and Humidity Sensor Module)

เซนเซอร์วัดความชื้นและอุณหภูมิ DHT11 [10] ผู้จัดทำขอเรียกอย่างย่อว่า เซนเซอร์ DHT11 ซึ่งมีทั้งแบบ 3 ขา และแบบ 4 ขา ในโครงการนี้ผู้จัดทำเลือกใช้แบบ 3 ขา เซนเซอร์ตัวนี้เป็นแบบดิจิทัล มีขาเอาต์พุตดิจิทัลเพียงขาเดียวเท่านั้น ต่อใช้งานได้ง่ายและมีราคาถูก



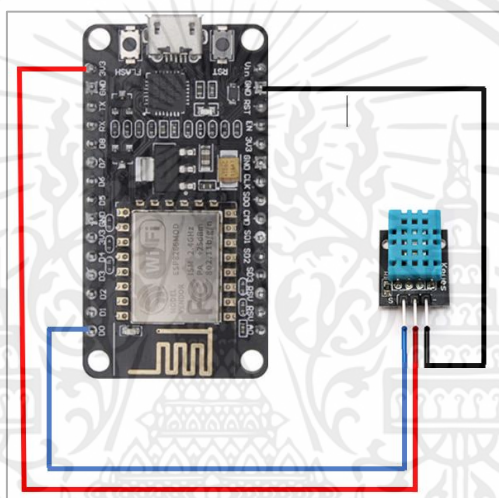
รูปที่ 2.45 เซนเซอร์วัดอุณหภูมิและความชื้น DHT11

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.6.1 ข้อมูลทางเทคนิคของเซนเซอร์ DHT11

1. ต้องการแรงดันไฟฟ้ากระแสตรง 3-5 V เพื่อให้เซนเซอร์ทำงาน
2. วัดความชื้นสัมพัทธ์ได้ที่ 20-80% และมีความผิดพลาดในการวัดไม่เกิน  $\pm 5\%$
3. วัดอุณหภูมิได้ในช่วง 0-50 °C โดยมีความผิดพลาดในการวัดไม่เกิน  $\pm 2$  °C
4. มีความถี่ในการวัด 1 Hz หรือก็คือ 1 วินาที/ครั้ง
5. มีขนาด กว้างxยาวxหนา เท่ากับ 12x15.5x5.5 มิลลิเมตร
6. มีจำนวน 3 ขา

### 2.6.2 การต่อใช้งานเซนเซอร์วัดอุณหภูมิ DHT11เข้ากับ ESP8266



รูปที่ 2.46 การเชื่อมต่อเซนเซอร์ DHT11 ให้เข้ากับ ESP8266

จากรูปที่ 2.46 เป็นการต่อวงจรระหว่างเซนเซอร์ DHT11 กับ ESP8266 เพื่อให้ใช้งานในวัดค่าอุณหภูมิและความชื้นในอากาศ และค่าที่วัดได้จะส่งไปให้บอร์ด ESP8266 เพื่อใช้ประมวลผล ซึ่งผู้เขียนได้บอกการเชื่อมต่อแต่ละขาเอาไว้ในตารางที่ 2.6

ตารางที่ 2.6 สถานะของการต่อใช้งานเซนเซอร์วัดอุณหภูมิ DHT11เข้ากับ ESP8266

ขาของเซนเซอร์ DHT11	ขาของ ESP8266
ขาสีแดง	V <sub>CC</sub>
ขาสีน้ำเงิน	Data
ขาสีดำ	GND

### 2.6.3 ตัวอย่างโค้ด

```
DHT11-Sensor.cpp
1 #include<DHT.h>
2 #include<ESPWifi8266.h>
3
4 DHT dht;
5
6 void setup()
7 {
8     Serial.begin(9600);
9     Serial.println();
10    Serial.println("Status\tHumidity (%)\tTemperature (C)\t(F)");
11    dht.setup(D1);
12 }
13 void loop()
14 {
15     delay(dht.getMinimumSamplingPeriod());
16     float humidity = dht.getHumidity();
17     float temperature = dht.getTemperature();
18     Serial.print(dht.getStatusString());
19     Serial.print("\t");
20     Serial.print(humidity, 1);
21     Serial.print("\t\t");
22     Serial.print(temperature, 1);
23     Serial.print("\t\t");
24     Serial.println(dht.toFahrenheit(temperature), 1);
25 }
```

รูปที่ 2.47 ตัวอย่างโค้ดสำหรับส่งการให้เซนเซอร์ DHT11 ทำงานร่วมกับ ESP8266

## 2.7 โมดูลกล้อง ESP32 (ESP32-CAM Module)

โมดูลกล้อง ESP32 [11] เป็นโมดูลกล้อง OV2640 ขนาดเล็กที่ติดตั้งอยู่บนบอร์ด ESP32 ซึ่งใช้พลังงานน้อย และมีช่องเสียบการ์ดสำหรับเก็บข้อมูลอยู่ในตัว กล้อง ESP32 นี้เป็นที่นิยมอย่างมากใช้งานกันอย่างแพร่หลายในแอปพลิเคชันของงานไอโอที (IoT) เช่น การตรวจสอบวิดีโอแบบไร้สาย การอัปโหลดรูปภาพ การระบุ QR Code เป็นต้น นอกจากนี้ยังสามารถเชื่อมต่อไวไฟได้อีกด้วย



รูปที่ 2.48 โมดูลกล้อง ESP32

### 2.7.1 คุณสมบัติของกล้อง ESP32

1. โมดูลออนบอร์ด ESP32-S สามารถเชื่อมต่อไว-ไฟและบลูทูธ (Bluetooth) ได้
2. กล้อง OV2640 มีแฟลชในตัว
3. มีช่องใส่การ์ด TF แบบออนบอร์ด สามารถเก็บข้อมูลได้สูงสุด 4 กิกะไบต์ (GB)
4. สามารถอัปโหลดรูปและถ่ายทอวิดีโอแบบทันทีทันใดผ่านไว-ไฟได้

#### ข้อควรระวัง

1. โปรดตรวจสอบทุกครั้งว่าโมดูลตัวนี้ได้รับแหล่งจ่ายไฟฟ้ากระแสตรงขนาด 5V ที่กระแสไฟฟ้า 2A อยู่หรือไม่ เพื่อให้โมดูลทำงานได้อย่างเต็มที่ มิเช่นนั้นอาจมีคลื่นน้ำปรากฏบนภาพ
2. ESP32 GPIO32 pin ใช้สำหรับควบคุมการใช้งานของกล้อง ดังนั้นเมื่อกำลังทำงานอยู่จะทำให้การเรียกใช้งาน GPIO32 จะไม่เสถียร

### 2.7.2 ข้อมูลทางเทคนิคของกล้อง ESP32

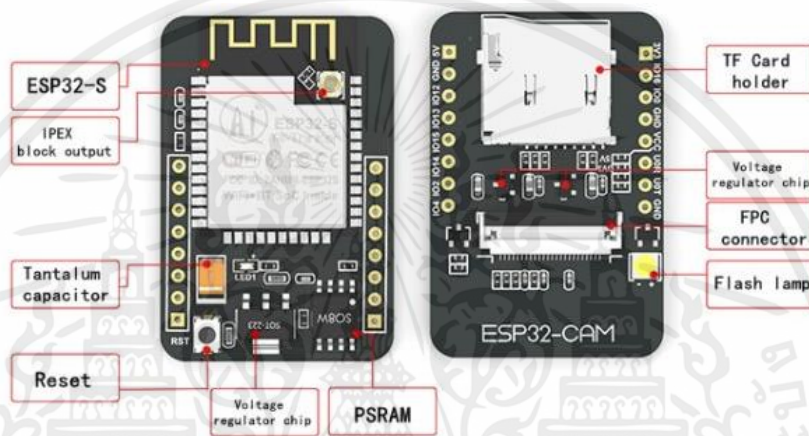
1. โมดูลไว-ไฟชื่อ: ESP-32S
2. หน่วยประมวลผล: ESP32-D0WD
3. แฟลชในตัว: 32Mbit
4. แรมภายในและภายนอก: 512 KB และ 4 MB (PSRAM) ตามลำดับ
5. เสาอากาศ PCB ออนบอร์ด
6. รองรับบลูทูธแบบ 4.2BR/EDR และ BLE
7. ความปลอดภัย: WPA/WPA2/WPA2-Enterprise/WPS
8. รูปแบบของภาพที่ส่งออก: JPEG (รองรับ OV2640 เท่านั้น), BMP, GRAYSCALE
9. การ์ด TF ที่รองรับ: สูงสุด 4 GB
10. อินเทอร์เฟซกับอุปกรณ์ต่อพ่วง: UART/SPI/I2C/PWM
11. มีพอร์ตอินพุตและเอาต์พุต (I/O): 9 ขา (รวมทั้งหมด)
12. อัตราการรับส่งข้อมูล: ค่าเริ่มต้น 115200 บิต/วินาที (bps)
13. แหล่งจ่ายแรงดันไฟฟ้า: 5 V<sub>DC</sub>

### 2.7.3 โมดูลกล้อง ESP32 เหมาะสำหรับการใช้งานไอโอทีด้านใดบ้าง

1. อัปโหลดรูปภาพอุปกรณ์สมาร์ทโฮม (Smart Home)
2. การตรวจสอบแบบไร้สาย
3. เกษตรอัจฉริยะ
4. รหัสไร้สาย QR (QR Code)
5. การตรวจจับใบหน้า/จดจำใบหน้า

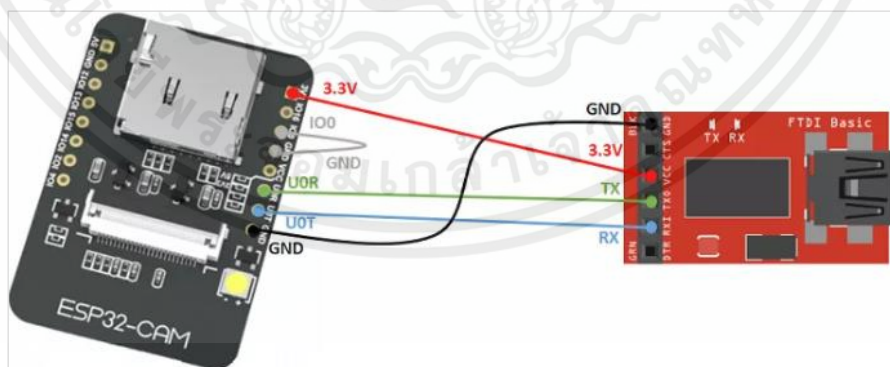


รูปที่ 2.49 สถานะขาของโมดูลกล้อง ESP32



รูปที่ 2.50 การบอกจุดการใช้งานที่สำคัญของโมดูลกล้อง ESP32

2.7.4 การเชื่อมต่อระหว่างกล้อง ESP32 กับ FT232RLUSB



รูปที่ 2.51 การต่อวงจรใช้งานกล้อง ESP32 กับ FT232RLUSB

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.51 เป็นการต่อวงจรระหว่างกล่อง ESP32 ให้ทำงานร่วมกับ FT232RLUSB เพื่อให้สามารถควบคุมการถ่ายรูปนิ่งได้ และสามารถควบคุมให้นำไฟล์รูปไปเก็บไว้ในฐานข้อมูลได้ สำหรับการต่อขาเพื่อให้งานร่วมกัน ได้แสดงไว้ในตารางที่ 2.7

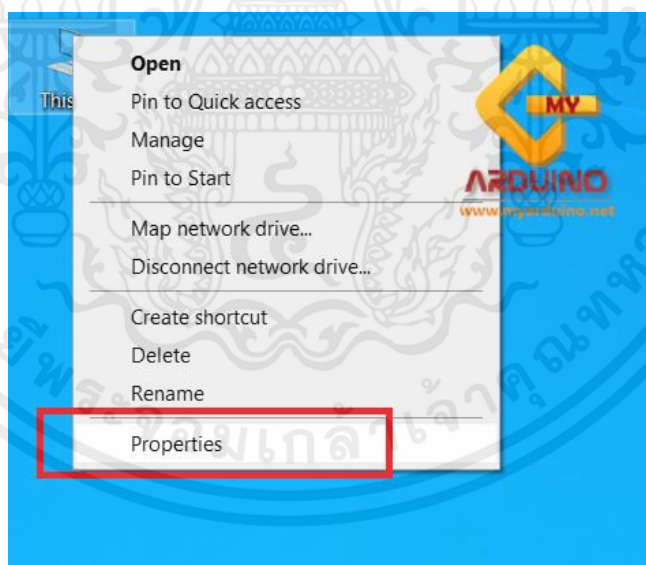
ตารางที่ 2.7 การเชื่อมต่อขาของกล่อง ESP32 กับ FT232RLUSB ให้ทำงานร่วมกัน

กล่อง ESP32	บอร์ด FT232RLUSB
3.3V	V <sub>cc</sub>
GND	GND
UoT	Rx
UoR	Tx

### 2.7.5 การติดตั้งไดรเวอร์ของกล่อง ESP32 และ FT232RLUSB

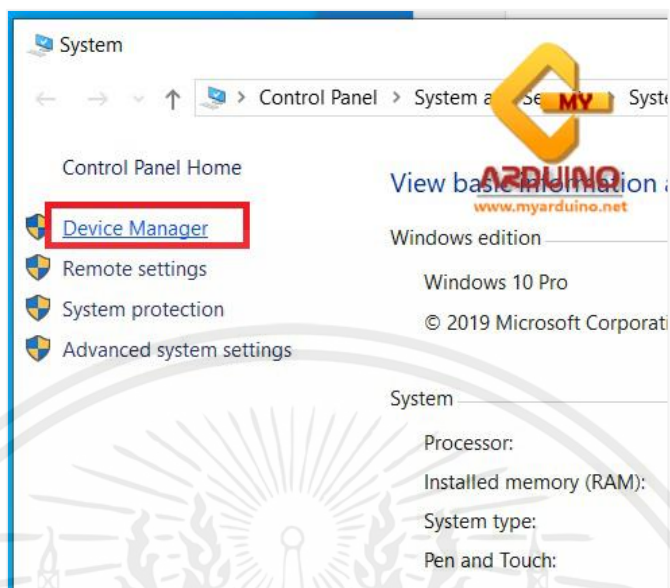
การติดตั้งไดรเวอร์ (Driver) เพื่อให้กล่อง ESP32 กับบอร์ด FT232RLUSB ได้ทำงานร่วมกันได้ [12] มีขั้นตอนดังต่อไปนี้

1. ให้ต่ออุปกรณ์ตามรูปที่ 2.51 ให้เสร็จก่อน ต่อจากนั้นให้ต่อสายมินิ USB (Mini-USB) เข้าคอมพิวเตอร์ให้เรียบร้อย จากนั้นให้คลิกขวาบนหน้าจอแล้วจึงเลือก Properties



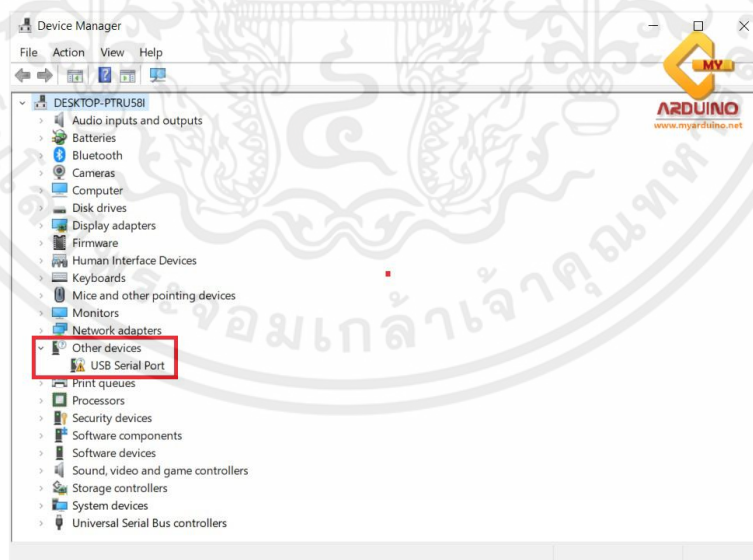
รูปที่ 2.52 ขั้นตอนแรกของการค้นหาไดรเวอร์ของกล่อง ESP32

2. เมื่อเข้ามาใน Properties แล้วให้คลิกที่ Device Manager



รูปที่ 2.53 ขั้นตอนที่สองของการค้นหาไดรเวอร์ของกล้อง ESP32

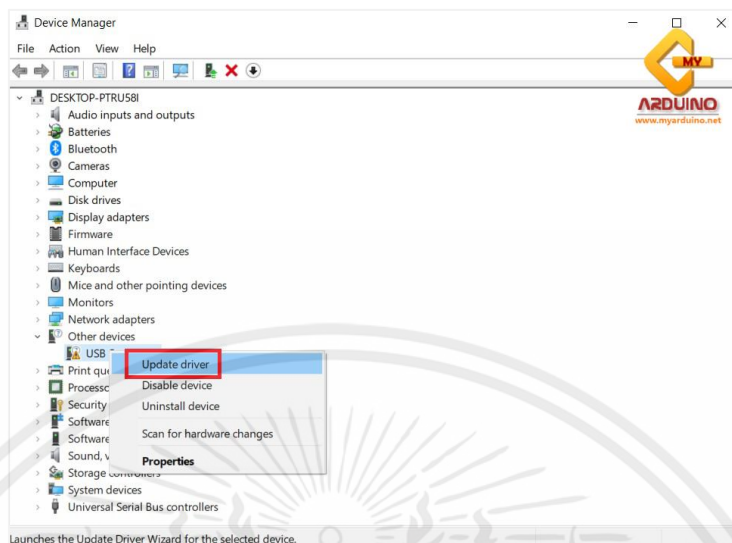
3. ขั้นตอนต่อมาคือการติดตั้งไดรเวอร์ของ FT232RLUSB เพื่อให้คอมพิวเตอร์รู้จักกับกล้อง ESP32 ดังรูปที่ 2.54 บ่งบอกให้เห็นว่าคอมพิวเตอร์ยังไม่รู้จักอุปกรณ์



รูปที่ 2.54 คอมพิวเตอร์กับกล้อง ESP32 ยังไม่สามารถทำงานร่วมกันได้

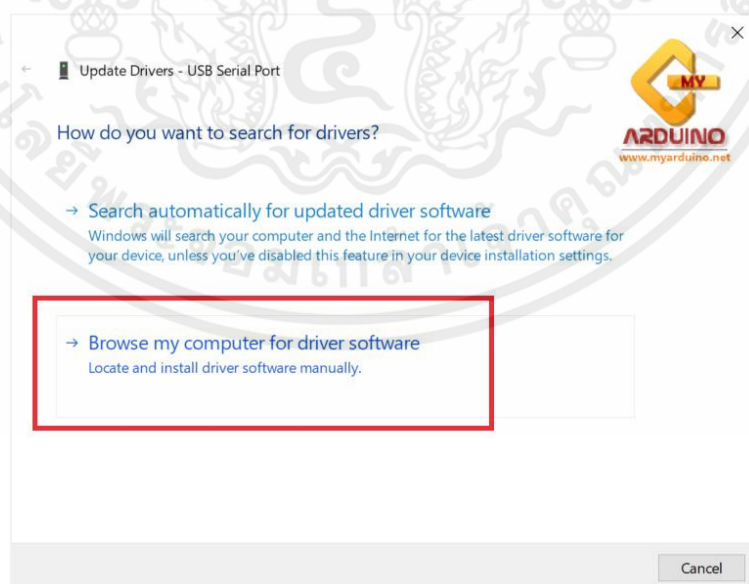
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4. คลิกขวาที่ USB Serial Port แล้วให้เลือก Update driver



รูปที่ 2.55 การอัปเดตไดรเวอร์ของกล่อง ESP32

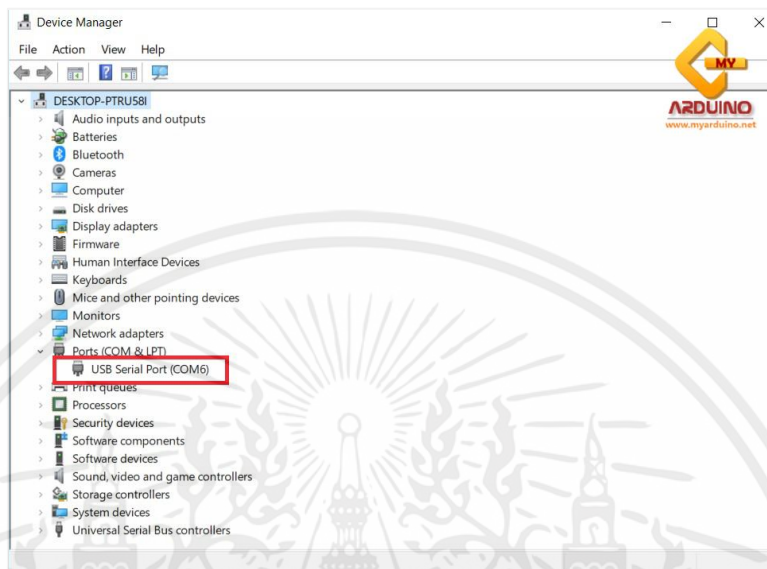
5. ให้คลิกที่ Browse my computer for driver software ต่อจากนั้นให้ไปตามหาว่าผู้อ่านได้เก็บไฟล์ของไดรเวอร์ FT232RLUSB ไว้ที่ใด ถ้าหากผู้อ่านยังไม่มีไฟล์ก็สามารถดาวน์โหลดได้ที่ลิงค์ <http://www.mediafire.com/file/36t9nmf402dh5bz/CDM-2.08.28-WHQL-Certified1.zip/file>



รูปที่ 2.56 การค้นหาไฟล์ไดรเวอร์ของ FT232RLUSB

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. เมื่อติดตั้งไดรเวอร์ FT232RLUSB สำเร็จแล้ว ก็จะปรากฏข้อความว่า USB Serial Port (COM6) นั่นหมายความว่า การเชื่อมต่อสมบูรณ์แล้ว ดังนั้นการเชื่อมต่อกับกล้อง ESP32 ก็จะเสร็จสมบูรณ์ตามไปด้วย ดังแสดงไว้ในรูปที่ 2.57



รูปที่ 2.57 การติดตั้งไดรเวอร์กล้อง ESP32 ที่เสร็จเรียบร้อยแล้ว

## 2.7.6 ตัวอย่างโค้ด

```
ESP32-CAM.cpp
1 #include<ESP32WIFI.h>
2 #include<stdio.h>
3
4 void setup()
5 {
6     WRITE_PERI_REG(RTC_CNTL_BROWN_OUT_REG, 0);
7     Serial.begin(115200); Serial.setDebugOutput(false);
8     camera_config_t config; config.ledc_channel = LEDC_CHANNEL_0;
9     config.ledc_timer = LEDC_TIMER_0; config.pin_d0 = Y2_GPIO_NUM;
10    config.pin_d1 = Y3_GPIO_NUM; config.pin_d2 = Y4_GPIO_NUM; config.pin_d3 = Y5_GPIO_NUM; config.pin_d4 = Y6_GPIO_NUM;
11    config.pin_d5 = Y7_GPIO_NUM; config.pin_d6 = Y8_GPIO_NUM; config.pin_d7 = Y9_GPIO_NUM; config.pin_xclk = XCLK_GPIO_NUM;
12    config.pin_pclk = PCLK_GPIO_NUM; config.pin_vsync = VSYNC_GPIO_NUM; config.pin_href = HREF_GPIO_NUM;
13    config.pin_sscb_sda = SIOD_GPIO_NUM; config.pin_sscb_scl = SIOC_GPIO_NUM; config.pin_pwdn = PMDN_GPIO_NUM;
14    config.pin_reset = RESET_GPIO_NUM; config.xclk_freq_hz = 20000000; config.pixel_format = PIXFORMAT_JPEG;
15    if(psramFound())
16    {
17        config.frame_size = FRAMESIZE_UXGA; config.jpeg_quality = 10; config.fb_count = 2;
18    }
19    else
20    {
21        config.frame_size = FRAMESIZE_SVGA; config.jpeg_quality = 12; config.fb_count = 1;
22    }
23    esp_err_t err = esp_camera_init(&config);
24    if (err != ESP_OK)
25    {
26        Serial.printf("Camera init failed with error 0x%x", err);
27        return;
28    }
29    WiFi.begin(ssid, password);
30    while (WiFi.status() != WL_CONNECTED)
31    {
32        delay(500); Serial.print(".");
33    }
34    Serial.println(""); Serial.println("WiFi connected"); Serial.print("Camera Stream Ready! Go to: http://");
35    Serial.print(WiFi.localIP()); startCameraServer();
36 }
37 void loop()
38 {
39    delay(1);
40 }
```

รูปที่ 2.58 ตัวอย่างโค้ดในการควบคุมให้กล้อง ESP32 ทำงานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.8 ภาษา HTML

HTML ย่อมาจาก Hypertext Markup Language [12] เป็นภาษาคอมพิวเตอร์รูปแบบหนึ่งที่มีโครงสร้างการเขียนโดยอาศัยตัวกำกับ (แท็ก: Tag) สำหรับควบคุมการแสดงผลข้อความ รูปภาพ หรือวัตถุอื่นๆ ผ่านเว็บเบราว์เซอร์ (Web Browser) แต่ละ Tag อาจจะมีส่วนขยายที่เรียกว่าแอตทริบิวต์ (Attribute) สำหรับระบุหรือควบคุมการแสดงผลของเว็บไซต์ได้ด้วย HTML เป็นภาษาที่ถูกพัฒนาโดย World Wide Web Consortium (W3C) ซึ่งมีร่างต้นแบบเป็นภาษา SGML (Standard Generalized Markup Language) โดยตัดความสามารถบางส่วนออกไป เพื่อให้สามารถทำความเข้าใจและเรียนรู้ได้ง่าย ด้วยความสะดวกรวดเร็วกว่าทำให้บริการ www เติบโตขยายตัวอย่างกว้างขวาง

### 2.8.1 ประวัติของภาษา HTML

จุดเริ่มต้นของ HTML [13] เริ่มมาจากการพัฒนาโดย ทิม เบอร์เนิร์ส ลี (Tim Berners Lee) แห่งศูนย์ปฏิบัติการวิจัยทางอนุภาคฟิสิกส์ของยุโรป (CERN) แห่งกรุงเจนีวา ประเทศสวิตเซอร์แลนด์ ซึ่งมีแนวความคิดที่จะให้นักวิทยาศาสตร์ภายในสถาบัน ค้นหาและสื่อสารข้อมูลถึงกัน และกันได้อย่างสะดวกขึ้น จึงคิดค้นวิธีการที่จะถ่ายทอดข้อมูลเอกสารในรูปแบบของไฮเปอร์เท็กซ์ (Hypertext) ซึ่งเป็นรูปแบบเอกสารที่แต่ละหน้าเชื่อมโยงถึงกันได้ จากนั้นได้มีการพัฒนาเครื่องมือที่เรียกว่าเว็บเบราว์เซอร์ เพื่อใช้ในการอ่านข้อมูลเอกสารไฮเปอร์เท็กซ์โดยมีโมเสค (MOSAIC) เป็นเว็บเบราว์เซอร์ตัวแรกถือกำเนิดขึ้นมา โดยใช้ HTTP (Hypertext Transport Protocol) เป็นโปรโตคอล (Protocol) ที่ใช้ในการรับส่งข้อมูล ซึ่งทำให้เอกสารไฮเปอร์เท็กซ์ได้ถูกพัฒนาขึ้นมาใหม่เรียกว่าภาษา HTML

ปัจจุบัน HTML ได้รับการดูแลพัฒนาจัดการและกำหนดมาตรฐาน ภายใต้องค์กรที่ชื่อ W3C ซึ่งที่ผ่านมา HTML ได้มีรุ่นต่างๆ ออกมาอย่างต่อเนื่อง ตั้งแต่ HTML Level 1, HTML2.0, HTML3.0, HTML3.2, HTML4.0, HTML4.01 และปัจจุบันคือ HTML5

### 2.8.2 คุณสมบัติเด่นของภาษา HTML

1. ภาษา HTML เป็นภาษาหลักในการสร้างเว็บไซต์ [14]
2. ภาษา HTML หรือบางครั้งเรียกว่าเอกสาร HTML ใช้สำหรับจัดการข้อความ ตัวอักษร รูปภาพ ไฟล์ และลิงก์ต่างๆ โดยการใช้แท็ก
3. สามารถเรียกเอกสาร HTML ได้ว่าเว็บเพจ (Web Page)
4. ภาษา HTML ไม่ใช่ภาษาโปรแกรมแต่เป็นภาษาแสดงข้อมูล (Markup)
5. ภาษาแสดงข้อมูล (Markup Language) อยู่ในกลุ่ม Markup Tags
7. ภาษา HTML จะใช้ Markup Tags ในการอธิบายหน้าเว็บเพจ

### 2.8.3 ส่วนประกอบที่สำคัญของภาษา HTML

ส่วนประกอบสำคัญ 2 อย่างนั่นคือ แท็กและแอตทริบิวต์

แท็ก คือคำสั่งที่ใช้ในภาษา HTML ซึ่งอยู่ในเครื่องหมาย < > ใช้สำหรับจัดรูปแบบ

ข้อความภาพหรือวัตถุอื่นๆ แท็กส่วนมากจะมีทั้งแท็กเปิดและแท็กปิด เช่น

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<h1>...</h1> ใช้เน้นหัวข้อเรื่อง

<p>...</p> ใช้จัดพารากราฟ

<b>...</b> ใช้กำหนดให้ตัวอักษรเป็นตัวหนา


แต่บางแท็กก็ไม่มีแท็กปิด เช่น

<hr> ใช้สร้างเส้นคั่น

<br> ใช้สำหรับการขึ้นบรรทัดใหม่

แท็กที่ใช้ในภาษา HTML มีอยู่มากมายซึ่งผู้เขียนขอยกตัวอย่างแท็กที่ได้พบเจอบ่อยๆ ในการสร้างเว็บไซต์มาพอสังเขป ดังที่แสดงไว้ในตารางที่ 2.8 หากผู้อ่านต้องการข้อมูลเพิ่มเติมสามารถไปค้นหาต่อได้อีกในเอกสารอ้างอิงหมายเลข 13 และ 14

ตารางที่ 2.8 แท็กคำสั่งในภาษา HTML

แท็ก	ความหมาย	ตัวอย่างการใช้	แสดงผลในเว็บเบราว์เซอร์
<CENTER>	จัดให้อยู่กึ่งกลาง	<CENTER>HelloWorld</CENTER>	HelloWorld
<IMG>	แทรกรูปภาพ	<IMG SRC="Earth.gif" BORDER="0">	
<U>	ขีดเส้นใต้	<U>Example</U>	<u>Example</u>
<I>	ตัวเอียง	<I>Example</I>	<i>Example</i>
<A>	การเชื่อมโยง	<A HREF="URL">Your Link</A>	Your Link
 	ขึ้นบรรทัดใหม่	Tom Jerry	Tom Jerry
<HR>	การสร้างเส้นในเอกสาร	<HR>	-----
<OL>	ลำดับรายการแบบ Ordered List	Example 1: <OL> <LI>Item 1 <LI>Item 2 <LI>Item 3 <LI>Item 4 </OL>	Example 1: 1. Item 1 2. Item 2 3. Item 3 4. Item 4

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.8.4 ส่วนประกอบของเอกสาร HTML (HTML Document)

1. เอกสาร HTML แบ่งออกเป็น 2 ส่วนนั่นคือส่วน head กับส่วน body
2. เอกสาร HTML จะต้องประกอบไปด้วยแท็กที่สำคัญอยู่ 3 แท็กนั่นคือ html, head และ body มีการวางรูปแบบแท็กดังนี้

- 1) แท็ก html บ่งบอกว่าเอกสารนี้เป็นเอกสาร HTML
- 2) แท็ก head เป็นส่วนที่ทำงานเกี่ยวกับส่วนหัวของเอกสารการกำหนดค่าต่าง ๆ และจะต้องอยู่ภายในแท็ก html เพื่อบ่งบอกว่าอยู่ภายในเอกสาร html
- 3) แท็ก body เป็นส่วนที่ทำงานเกี่ยวกับตัวของเอกสารที่จะแสดงในหน้าเว็บเพจ และจะต้องอยู่ภายในแท็ก html และอยู่หลังแท็ก head

```

1 <html>
2   <head>
3   </head>
4   <body>
5   </body>
6 </html>

```

รูปที่ 2.59 ส่วนประกอบของเอกสาร HTML ที่สำคัญ

## 2.8.5 การสร้างเอกสาร HTML

ในการสร้างเอกสาร HTML สามารถสร้างได้ง่ายๆโดยใช้โปรแกรมจำพวก Code Editor ยกตัวอย่างเช่น VS Code, Dreamweaver, EditPlus เป็นต้น หรือจะใช้โปรแกรมกลุ่ม Word Processor เช่น Notepad ในการเขียนภาษา HTML ก็ได้แต่ต้องเซฟเป็นไฟล์นามสกุล .html

### 1. การสร้างเว็บเพจด้วยโปรแกรม Notepad

- 1) ให้เปิด Notepad แล้วทำการเขียนโค้ดดังรูปที่ 2.60

```

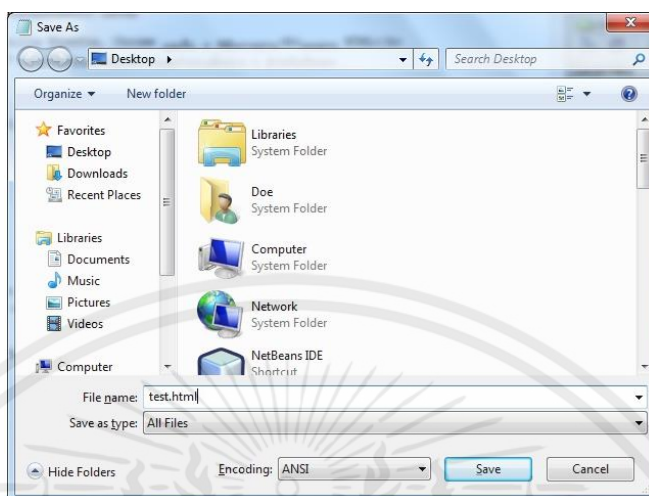
<html>
<head>
<title>Test by admin www.doesystem.com</title>
</head>
<body>
Test HTML
</body>
</html>

```

รูปที่ 2.60 ตัวอย่างการสร้างเอกสาร HTML ด้วยโปรแกรม Notepad

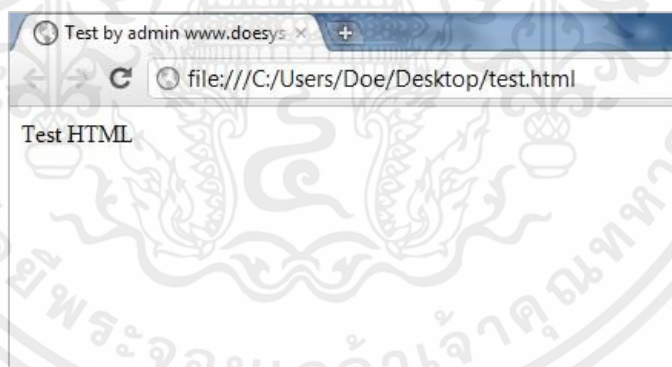
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) เสร็จแล้วทำการเซฟโดยใช้นามสกุล .html และเปลี่ยน Save as type ให้เป็น All Files ดังรูปที่ 2.61



รูปที่ 2.61 ตัวอย่างการเซฟไฟล์เป็นนามสกุล .html

3) ให้ลองเป็นไฟล์เอกสาร HTML ที่เซฟไปเมื่อครู จะได้ผลลัพธ์เป็นหน้าเว็บเพจตามที่ได้เขียนโค้ดไว้ ดังที่แสดงในรูปที่ 2.61

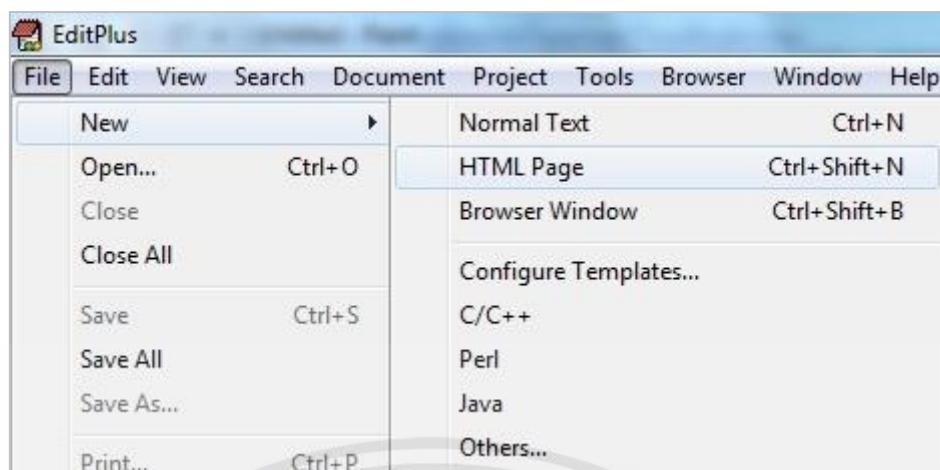


รูปที่ 2.62 ผลลัพธ์จากการเปิดไฟล์ test.html

## 2. การสร้างเว็บเพจด้วยโปรแกรม EditPlus

1) เปิดโปรแกรม EditPlus แล้วไปที่ File > New > HTML Page

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.63 การเริ่มใช้งานโปรแกรม EditPlus

2) เมื่อเปิดโปรแกรมเสร็จแล้ว จะเห็นว่ามิโค้ดบางส่วนมาให้เรียบร้อยแล้ว ซึ่งเป็นโค้ดตั้งต้นที่ใช้สำหรับสร้างเอกสาร HTML และผู้เขียนได้เติมโค้ด “Test HTML” ลงไปในแท็ก <BODY>...</BODY> ดังรูปที่ 2.64 ส่วนผลลัพธ์ที่ได้จะเหมือนกับรูป 2.62

```

1  <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
2  <HTML>
3  <HEAD>
4  <TITLE>Test by admin www.doesystem.com</TITLE>
5  <META NAME="Generator" CONTENT="EditPlus">
6  <META NAME="Author" CONTENT="">
7  <META NAME="Keywords" CONTENT="">
8  <META NAME="Description" CONTENT="">
9  </HEAD>
10
11 <BODY>
12   Test HTML
13 </BODY>
14 </HTML>
15 |

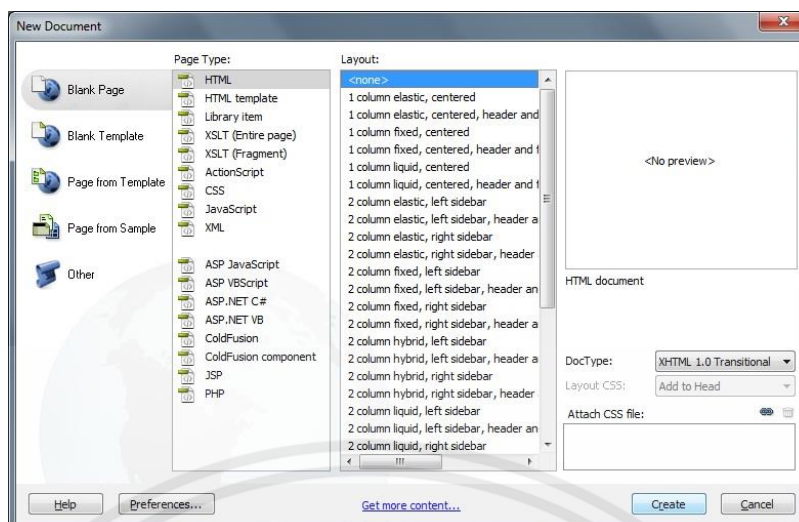
```

รูปที่ 2.64 การเริ่มเขียนโค้ด HTML ด้วยโปรแกรม EditPlus

### 3. การสร้างเว็บด้วยโปรแกรม Dreamweaver

1) เริ่มสร้างเอกสารโดยไปที่ File > New หรือกด Ctrl+N ก็ได้ ซึ่งจะแสดงหน้าต่าง New Document ให้เห็นดังที่แสดงให้เห็นในรูปที่ 2.65

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.65 การเริ่มใช้งานโปรแกรม Dreamweaver

2) จากรูปที่ 2.65 ให้เลือก Blank Page > HTML > none จากนั้นให้กด Create ก็จะมีการเขียนโค้ด HTML ซึ่งจะโค้ดส่วนที่สำคัญมาให้แล้ว ดังรูปที่ 2.66 ต่อจากนี้ผู้เขียนได้เติมโค้ด “Test HTML” ลงไปในแท็ก <BODY>...</BODY> ดังรูปที่ 2.66 ส่วนผลลัพธ์ที่ได้จะเหมือนกับรูป 2.62



รูปที่ 2.66 เอกสาร HTML โดย Dreamweaver

## 2.9 ภาษา CSS

CSS ย่อมาจาก Cascading Style Sheets [15] เป็นภาษาที่มีไวยากรณ์การเขียน (Syntax) ที่เฉพาะ และถูกกำหนดมาตรฐานโดย W3C เช่นเดียวกับ HTML และ XHTML ภาษามีจุดเด่นที่ใช้สำหรับตกแต่งเอกสาร HTML/XHTML ให้มีหน้าตา สี สัน ตัวอักษร เส้นขอบ พื้นหลัง ระยะห่าง ฯลฯ อย่างที่ต้องการ ด้วยการกำหนดคุณสมบัติให้กับ Element ต่างๆ ของ HTML เช่น <body>, <p>, <h1> เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.9.1 CSS คืออะไร

ภาษา CSS คือภาษาที่ใช้กำหนดรูปแบบการแสดงผล HTML และถูกสร้างขึ้นเพื่อใช้งานร่วมกับ HTML กล่าวคือ HTML ใช้สำหรับกำหนดโครงสร้างข้อมูล และ CSS ใช้สำหรับกำหนดรูปแบบการแสดงผล ตัวอย่างเช่น

HTML:

```
<h1>Welcome to my web site</h1>
```

CSS:

```
h1
```

```
{
```

```
    color: #0000FF; /* Blue */
```

```
}
```

ในตัวอย่างข้างต้นเราใช้แท็ก <h1> เพื่อกำหนดหัวข้อเท่านั้น และแยกส่วนรูปแบบการแสดงผลให้เป็นหน้าที่ของ CSS และกำหนดให้แสดงผลสีฟ้า ดังนั้นจึงกล่าวได้ว่า CSS คือภาษาที่ใช้ในการกำหนดรูปแบบหรือสไตล์ (Style) สำหรับ HTML

การแยกการแสดงผลออกจากข้อมูลช่วยให้ง่ายต่อการเปลี่ยนแปลงรูปแบบในอนาคต เช่น ถ้าเราต้องการเปลี่ยนสีให้กับหัวข้อในแท็ก <h1> ในทุก ๆ เว็บเพจ เราเพียงเปลี่ยนที่ไฟล์ CSS เท่านั้น (ไฟล์ CSS สามารถใช้ได้กับทุก ๆ เว็บเพจที่ต้องการ) ปัจจุบัน CSS เป็นที่นิยมอย่างมากในการจัดรูปแบบให้กับเว็บเพจ ซึ่งก่อนการมาของ CSS นั้นเราใช้ตารางในการจัดเลย์เอาต์ และใช้ properties ของ HTML ในการจัดรูปแบบ

### 2.9.2 รูปแบบการใช้คำสั่ง CSS

1. Comment คือการแทรกคำอธิบายลงไปในโค้ด ไม่มีผลต่อรูปแบบการแสดงผลใด ๆ ทั้งสิ้น เพื่อให้ผู้ที่มาอ่านโค้ดของเรา เข้าใจโค้ดได้ง่ายขึ้น หรือแม้แต่จะอธิบายตัวเองก็ได้ เพราะเราอาจลืมในสิ่งที่เราเขียนเมื่อเวลาผ่านไป คอมเมนต์มี Syntax ดังนี้

```
รูปแบบ /* . . . any comment here . . . */
```

```
ตัวอย่าง /* . . . ส่วนเพิ่มข้อมูลรายวิชา . . . */
```

2. Selectors คือตัวระบุแท็ก HTML ที่เราต้องการกำหนดสไตล์ให้ จากตัวอย่างโค้ด CSS แบบ Embedded และ External ที่ผ่านมา เราใช้ Syntax ดังนี้

```
รูปแบบ selector {
```

```
    property:value;
```

```
}
```

```
ตัวอย่าง p {
```

```
    color: #FFFFFF;
```

```
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Selector ไม่เพียงแต่จะระบุแท็กด้วยชื่อเท่านั้น แต่ยังสามารถระบุแท็กด้วยวิธีการอื่นอีกด้วย การใช้ Selector มีอยู่ 3 รูปแบบหลัก ๆ ด้วยกันคือ

1. ใช้ชื่อแท็ก (tag{ }) การระบุ Selector ด้วยชื่อแท็ก การระบุด้วยชื่อแท็กนั้นก็เป็นการวิธีการดังเช่นที่ได้ยกตัวอย่างมา โดยมี Syntax ดังนี้

```
รูปแบบ tagName {
    property:value;
}
```

การระบุด้วยชื่อแท็กนั้นจะส่งผลถึงทุก ๆ แท็กภายในเว็บเพจนั้น รูปแบบนี้เหมาะสำหรับกรณีที่แท็กนั้นๆ ใช้รูปแบบเดียวกันทั้งเว็บเพจ ตัวอย่างเช่น ถ้าต้องการเปลี่ยนสีทั้งหมดให้เป็นสีขาว ก็ทำได้ดังนี้

```
a {
    color: #FFFFFF; /* white */
}
```

ทุกลิงค์ในเว็บเพจนั้นจะเป็นสีขาวหมด แต่หากต้องการระบุเพียงแค่บางแท็ก ถึงแม้จะมีชื่อแท็กเหมือนกัน ก็ต้องใช้วิธีระบุด้วยแอตทริบิวต์ class

2. ใช้แอตทริบิวต์ id (#id{ }) การระบุด้วยแอตทริบิวต์ id ใช้ในกรณีที่ต้องการระบุถึงแท็กเดียวภายในเว็บเพจเท่านั้น เนื่องด้วยเราสามารถตั้ง id ให้กับแท็ก HTML ใด ๆ ก็ได้ และ id ก็จะต้องไม่ซ้ำกันด้วย ดังนั้นหากเราต้องการระบุถึงแท็กใดเป็นการเฉพาะเจาะจง ก็ให้ตั้ง id ให้กับแท็กนั้นๆ เสียก่อน

ตัวอย่าง <p id = "footer">kmitl 2018</p>

ถึงแม้ว่าจะมีแท็ก <p> มากกว่าหนึ่งแท็กในเว็บเพจเดียวกัน เราก็สามารถอ้างถึงแท็กดังกล่าวข้างต้นได้ด้วย id "footer" อย่างไรก็ตามในการระบุ Selector ด้วย id มีรูปแบบการเขียนที่แตกต่างจากการระบุด้วยชื่อแท็ก กล่าวคือเราจะต้องใช้เครื่องหมาย "#" นำหน้า Selector เสมอ ทั้งนี้เพื่อบอกให้เว็บเบราว์เซอร์รู้ว่า กำลังอ้างถึงด้วย id โดยมี Syntax ดังนี้

```
รูปแบบ #id {
    property:value;
}
```

ดังนั้นจากตัวอย่างข้างต้น เราสามารถกำหนดให้ย่อหน้าข้อความ "kmitl 2018" เป็นตัวหนา และมีการจัดเรียงข้อความให้อยู่กึ่งกลางได้ดังนี้

```
ตัวอย่าง #footer {
    font-weight: bold;
    text-align: center;
}
```

รูปแบบการระบุ Selector ด้วยแอตทริบิวต์ id นี้ เหมาะกับการปรับแต่งส่วนใดส่วนหนึ่ง เฉพาะภายในเว็บเพจ ซึ่งจะไม่ซ้ำกันโดยเด็ดขาด เช่นในการทำเลย์เอาต์เราอาจแบ่งเว็บเพจออกเป็น ส่วนต่าง ๆ เช่น Header, Content, Menu, Sidebar, Footer เป็นต้น ในแต่ละส่วนก็มีรูปแบบการ แสดงผลที่ไม่เหมือนกัน ในกรณีนี้เราควรใช้การระบุ Selector ด้วยแอตทริบิวต์ id (ในการจัดเลย์เอาต์ เราไม่ใช่แท็ก <p> แต่จะใช้แท็ก <div> แทน)

3. ใช้แอตทริบิวต์ class (.class{ }) การระบุ Selector ด้วยชื่อแท็กจะเปลี่ยนแปลง สไตล์ให้กับแท็กทุก ๆ แท็กที่ชื่อเดียวกัน ในขณะที่การระบุ Selector ด้วย id จะเปลี่ยนแปลงสไตล์ ให้กับแท็กใดแท็กหนึ่งเฉพาะเจาะจงเท่านั้น ในรูปแบบสุดท้ายนี้เราจะมาดูวิธีที่อยู่ตรงกลาง กล่าวคือ การเปลี่ยนแปลงสไตล์ให้กับแท็กไม่ทั้งหมดแต่ก็มากกว่าหนึ่งแท็กได้ นั่นก็คือการระบุ Selector ด้วย แอตทริบิวต์ class นั่นเอง วิธีการนี้มีประโยชน์มาก เช่น เราต้องการให้ลิงค์ภายในเว็บไซต์เรามีสีขา วแต่ลิงค์ไปยังเว็บไซต์อื่นเป็นสีส้ม จะเห็นได้ว่าเราต้องการให้แท็ก <a> มีรูปแบบการแสดงผลที่ แตกต่างกัน แม้จะเป็นแท็ก <a> เหมือนกัน เป็นต้น การระบุ Selector ด้วยแอตทริบิวต์ class เริ่มต้นด้วยการกำหนดแอตทริบิวต์ class ภายในแท็กนั้นๆ ก่อน เช่น

```
ตัวอย่าง <a class = "ex" href = "/about">about us</a>
```

```
<a class = "ex" href = "/contact">contact us</a>
```

HTML อนุญาตให้เราใส่แอตทริบิวต์ class ลงไปในแท็กใด ๆ ก็ได้ เพื่อประโยชน์ในการจัด พวกหรือประเภท ให้นี้ง่าย ๆ ว่า ถ้าแท็กใดมี class เดียวกัน ก็จัดอยู่ในพวกเดียวกัน ก็สามารถใช้ สไตล์แบบเดียวกันได้ ทั้งนี้มาดูวิธีการระบุ Selector ด้วยแอตทริบิวต์ class ใน CSS กัน ให้นำหน้า ด้วยเครื่องหมายจุด "." แล้วตามด้วยชื่อ class โดยมี Syntax ดังนี้

```
รูปแบบ .class {
    property:value;
}
```

จากตัวอย่างลิงค์ก่อนหน้านี้ เราสามารถระบุ Selector ได้ดังนี้

```
รูปแบบ .ex {
    color: #fe8809;
}
```

ในตัวอย่างข้างต้นเรากำหนดสไตล์ให้กับแท็ก <a> ที่มี class เป็น "ex" เท่านั้น แท็ก <a> อื่น ๆ ยังคงใช้ค่าสีเป็นค่าตั้งต้นคือสีส้ม หากต้องการให้เป็นสีอื่นนอกจากสีส้ม ก็ต้องกำหนด class เพิ่ม นอกจากนี้การกำหนดค่า class ยังสามารถซ้ำกันข้ามแท็กก็ทำได้ เช่น

```
ตัวอย่าง <h1 class = "ex">
```

```
<p class = "ex">
```

### 2.9.3 ประโยชน์ของ CSS

1. การใช้ CSS ในการจัดรูปแบบการแสดงผล จะช่วยลดการใช้ภาษา HTML ในการตกแต่งเอกสารเว็บเพจ ทำให้โค้ดภายในเอกสาร HTML เหลือเพียงส่วนเนื้อหา ทำให้เข้าใจง่ายขึ้น การแก้ไขเอกสารทำได้ง่ายและรวดเร็ว
2. หากมีการเขียนโค้ดภายในเอกสาร HTML ลดลง ทำให้ขนาดไฟล์เล็กลง จึงดาวน์โหลดได้เร็ว สามารถกำหนดรูปแบบการแสดงผลจากคำสั่ง style sheet ชุดเดียวกัน ให้มีผลกับเอกสาร HTML ทั้งหน้าหรือทุกหน้าได้ ทำให้เวลาแก้ไขหรือปรับปรุงทำได้ง่าย ไม่ต้องไล่ตามแก้ที่แท็กของ HTML ทั่วทั้งเอกสาร
3. สามารถควบคุมการแสดงผลให้เหมือนกัน หรือใกล้เคียงกันได้ ในหลายเว็บเบราว์เซอร์
4. สามารถกำหนดการแสดงผลในรูปแบบที่เหมาะสมกับสื่อชนิดต่าง ๆ ไม่ว่าจะเป็นเป็นการแสดงผลบนหน้าจอ บนกระดาษเมื่อสั่งพิมพ์ บนโทรศัพท์มือถือ โดยที่เป็นเนื้อหาเดียวกัน
5. ทำให้เป็นเว็บไซต์ที่มีมาตรฐาน ปัจจุบันการใช้แอตทริบิวต์ของ HTML ตกแต่งเอกสารเว็บเพจ นั้นล้าสมัยแล้ว W3C แนะนำให้เราใช้ CSS แทน ดังนั้นหากเราใช้ CSS กับเอกสาร HTML ของเรา ก็จะทำให้เข้ากับเว็บเบราว์เซอร์ในอนาคตได้ดี

### 2.10 ภาษา JavaScript

JavaScript คือภาษาคอมพิวเตอร์สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ต ที่กำลังได้รับความนิยมอย่างสูง Java JavaScript เป็น ภาษาสคริปต์เชิงวัตถุ (ที่เรียกกันว่า "สคริปต์" Script) ซึ่งในการสร้างและพัฒนาเว็บไซต์ (ใช้ร่วมกับ HTML) เพื่อให้เว็บไซต์ของเราดูมีการเคลื่อนไหว สามารถตอบสนองผู้ใช้งานได้มากขึ้น ซึ่งมีวิธีการทำงานในลักษณะ "แปลความและดำเนินงานไปที่ละคำสั่ง" (interpret) หรือเรียกว่า อ็อบเจ็กต์โอเรียนเตด (Object Oriented Programming) ที่มีเป้าหมายในการ ออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ต สำหรับผู้เขียนด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ โดยทำงานร่วมกับ ภาษา HTML และภาษา Java ได้ทั้งทางฝั่งไคลเอนต์ (Client) และ ทางฝั่งเซิร์ฟเวอร์ (Server) [16]

JavaScript ถูกพัฒนาขึ้นโดย เน็ตสเคปคอมมิวนิเคชันส์ (Netscape Communications Corporation) โดยใช้ชื่อว่า Live Script ออกมาพร้อมกับ Netscape Navigator 2.0 เพื่อใช้สร้างเว็บเพจโดยติดต่อกับเซิร์ฟเวอร์แบบ Live Wire ต่อมาเน็ตสเคปจึงได้ร่วมมือกับ บริษัทซันไมโครซิสเต็มส์ ปรับปรุงระบบของเบราว์เซอร์เพื่อให้สามารถติดต่อกับภาษาจาวาได้ และได้ปรับปรุง LiveScript ใหม่เมื่อ ปี 2538 แล้วตั้งชื่อใหม่ว่า JavaScript JavaScript สามารถทำให้ การสร้างเว็บเพจ มีลูกเล่น ต่าง ๆ มากมาย และยังสามารถโต้ตอบกับผู้ใช้ได้อย่างทันที เช่น การใช้เมาส์คลิก หรือ การกรอกข้อความในฟอร์ม เป็นต้น

เนื่องจาก JavaScript ช่วยให้ผู้พัฒนา สามารถสร้างเว็บเพจได้ตรงกับความต้องการ และมีเอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความน่าสนใจมากขึ้น ประกอบกับเป็นภาษาเปิด ที่ใครก็สามารถนำไปใช้ได้ ดังนั้นจึงได้รับความนิยมเป็นอย่างสูง มีการใช้งานอย่างกว้างขวาง รวมทั้งได้ถูกกำหนดให้เป็นมาตรฐานโดย ECMA การทำงานของ JavaScript จะต้องมีการแปลความคำสั่ง ซึ่งขั้นตอนนี้จะถูกจัดการโดยบราวเซอร์ (เรียกว่าเป็น client-side script) ดังนั้น JavaScript จึงสามารถทำงานได้ เฉพาะบนบราวเซอร์ที่สนับสนุน ซึ่งปัจจุบันบราวเซอร์เกือบทั้งหมดก็สนับสนุน JavaScript แล้ว อย่างไรก็ตาม สิ่งที่ต้องระวังคือ JavaScript มีการพัฒนาเป็นเวอร์ชันใหม่ๆ ออกมาด้วย (ปัจจุบันคือรุ่น 1.5) ดังนั้น ถ้านำโค้ดของเวอร์ชันใหม่ ไปรันบนบราวเซอร์รุ่นเก่าที่ยังไม่สนับสนุน ก็อาจจะทำให้เกิด error ได้

### 2.10.1 JavaScript ทำอะไรได้บ้าง

1. JavaScript ทำให้สามารถใช้เขียนโปรแกรมแบบง่ายๆ ได้ โดยไม่ต้องพึ่งภาษาอื่น
2. JavaScript มีคำสั่งที่ตอบสนองกับผู้ใช้งาน เช่น เมื่อผู้ใช้คลิกที่ปุ่ม หรือ Checkbox ก็สามารถสั่งให้เปิดหน้าต่างใหม่ได้ ทำให้เว็บไซต์ของเรามีปฏิสัมพันธ์กับผู้ใช้งานมากขึ้น นี่คือข้อดีของ JavaScript เลยก็ว่าได้ที่ทำให้เว็บไซต์ต่างๆ ทั้หลายเช่น Google Map ต่างหันมาใช้
3. JavaScript สามารถเขียนหรือเปลี่ยนแปลง HTML Element ได้ นั่นคือสามารถเปลี่ยนแปลงรูปแบบการแสดงผลของเว็บไซต์ได้ หรือหน้าแสดงเนื้อหาสามารถซ่อนหรือแสดงเนื้อหาได้แบบง่ายๆ นั่นเอง
4. JavaScript สามารถใช้ตรวจสอบข้อมูลได้ สังเกตว่าเมื่อเรากรอกข้อมูลบางเว็บไซต์ เช่น อีเมล เมื่อเรากรอกข้อมูลผิดจะมีหน้าต่างป๊อปขึ้นมาว่าเรากรอกผิด หรือลืมกรอกอะไรบางอย่าง เป็นต้น
5. JavaScript สามารถใช้ในการตรวจสอบผู้ใช้ได้เช่น ตรวจสอบว่าผู้ใช้กำลังใช้เว็บเบราว์เซอร์ตัวไหนอยู่
6. JavaScript สร้าง Cookies (เก็บข้อมูลของผู้ใช้ในคอมพิวเตอร์ของผู้ใช้เอง) ได้

### 2.10.2 ความสามารถของ JavaScript

1. ทำให้สามารถเขียนโปรแกรมแบบง่ายๆ ได้ โดยไม่ต้องพึ่งภาษาอื่น
2. มีคำสั่งที่ตอบสนองต่อผู้ใช้งาน
3. ใช้ตรวจสอบข้อมูลได้ เช่น เตือนเมื่อกรอกรูปแบบข้อมูลผิดพลาด
4. ใช้ตรวจสอบผู้ใช้ได้ เช่น ตรวจสอบว่าผู้ใช้งานใช้เว็บเบราว์เซอร์อะไร
5. สร้าง Cookies ได้
6. สร้างลูกเล่นให้เว็บไซต์ได้

ในบทเรียนเรื่อง JavaScript นี้ จะอธิบายพร้อมทั้งยกตัวอย่างประกอบ การใช้ JavaScript เพื่อให้ผู้ศึกษาเข้าใจการทำงานร่วมกันของ JavaScript กับ HTML และ CSS ซึ่งก่อนที่จะเริ่มเรียนเรื่อง JavaScript นั้น ผู้ศึกษาจำเป็นต้องมีความรู้พื้นฐานเรื่อง HTML และ CSS เสียก่อน

### 2.10.3 ข้อดีและข้อเสียของ JavaScript

การทำงานของ JavaScript เกิดขึ้นบนเบราว์เซอร์ (เรียกว่าเป็น Client-Side Script) ดังนั้นไม่ว่าคุณจะใช้เซิร์ฟเวอร์อะไร หรือที่ไหน ก็ยังคงสามารถใช้ JavaScript ในเว็บเพจได้ ต่างกับภาษาสคริปต์อื่น เช่น Perl, PHP หรือ ASP ซึ่งต้องแปลความและทำงานที่ตัวเครื่องเซิร์ฟเวอร์ (เรียกว่า Server-Side Script) ดังนั้นจึงต้องใช้บนเซิร์ฟเวอร์ ที่สนับสนุนภาษาเหล่านี้เท่านั้น อย่างไรก็ตาม ก็ดี จากลักษณะดังกล่าวก็ทำให้ JavaScript มีข้อจำกัด คือไม่สามารถรับและส่งข้อมูลต่างๆ กับเซิร์ฟเวอร์โดยตรง เช่น การอ่านไฟล์จากเซิร์ฟเวอร์ เพื่อนำมาแสดงบนเว็บเพจ หรือรับข้อมูลจากผู้ชม เพื่อนำไปเก็บบนเซิร์ฟเวอร์ เป็นต้น ดังนั้นงานลักษณะนี้ จึงยังคงต้องอาศัยภาษา Server-Side Script อยู่ (ความจริง JavaScript ที่ทำงานบนเซิร์ฟเวอร์เวอร์ก็มี ซึ่งต้องอาศัยเซิร์ฟเวอร์ที่สนับสนุน โดยเฉพาะเช่นกัน แต่ไม่เป็นที่ยิมนัก)

## 2.11 โปรแกรม MySQL

MySQL คือโปรแกรมระบบจัดการฐานข้อมูล [17] ที่พัฒนาโดยบริษัท MySQL AB มีหน้าที่เก็บข้อมูลอย่างเป็นระบบ รองรับคำสั่ง SQL เป็นเครื่องมือสำหรับเก็บข้อมูล ที่ต้องใช้ร่วมกับเครื่องมือหรือโปรแกรมอื่นอย่างบูรณาการ เพื่อให้ได้ระบบงานที่รองรับความต้องการของผู้ใช้ เช่น ทำงานร่วมกับเครื่องบริการเว็บ (Web Server) เพื่อให้บริการแก่ภาษาสคริปต์ที่ทำงานฝั่งเครื่องบริการ (Server-Side Script) เช่น ภาษา PHP ภาษา ASP.NET หรือภาษา JSP เป็นต้น หรือทำงานร่วมกับโปรแกรมประยุกต์ (Application Program) เช่น ภาษา Visual Basic.NET ภาษา Java หรือภาษา C# เป็นต้น โปรแกรมถูกออกแบบให้สามารถทำงานได้บนระบบปฏิบัติการที่หลากหลาย และเป็นระบบฐานข้อมูลโอเพนซอร์สที่ถูกนำไปใช้งานมากที่สุด

### 2.11.1 ความสามารถและการทำงานของโปรแกรม MySQL

MySQL ถือเป็นระบบจัดการฐานข้อมูล (Database Management System (DBMS) ฐานข้อมูลมีลักษณะเป็นโครงสร้างของการเก็บรวบรวมข้อมูล การที่จะเพิ่มเติม เข้าถึงหรือประมวลผลข้อมูลที่เก็บในฐานข้อมูลจำเป็นต้องอาศัยระบบจัดการ ฐานข้อมูล ซึ่งจะทำหน้าที่เป็นตัวกลางในการจัดการกับข้อมูลในฐานข้อมูลทั้งสำหรับการ ใช้งานเฉพาะ และรองรับการทำงานของแอปพลิเคชันอื่นๆ ที่ต้องการใช้งานข้อมูลในฐานข้อมูล เพื่อให้ได้รับความสะดวกในการจัดการกับข้อมูลจำนวนมาก MySQL ทำหน้าที่เป็นทั้งตัวฐานข้อมูลและระบบจัดการฐานข้อมูล

โดย MySQL เป็นระบบจัดการฐานข้อมูลแบบ Relational ฐานข้อมูลแบบ relational จะทำการเก็บข้อมูลทั้งหมดในรูปแบบของตารางแทนการเก็บข้อมูลทั้งหมดลงในไฟล์เพียงไฟล์เดียว ทำให้ทำงานได้รวดเร็วและมีความยืดหยุ่น นอกจากนี้ แต่ละตารางที่เก็บข้อมูลสามารถเชื่อมโยงเข้าหากันทำให้สามารถรวมหรือจัด กลุ่มข้อมูลได้ตามต้องการ โดยอาศัยภาษา SQL ที่เป็นส่วนหนึ่งของโปรแกรม MySQL ซึ่งเป็นภาษามาตรฐานในการเข้าถึงฐานข้อมูล

### 2.11.2 ข้อควรระวังที่เกี่ยวข้องกับความปลอดภัยของโปรแกรม MySQL

การใช้งานโปรแกรม MySQL ให้มีความปลอดภัยนั้น ผู้ดูแลจะต้องพิจารณาถึงวิธีการที่ผู้ใช้หรือผู้อื่น ๆ จะเข้าถึงข้อมูลต่าง ๆ ในฐานข้อมูล และจำกัดสิทธิ์การใช้งานของผู้ที่จะเข้าใช้งานให้ได้รับสิทธิ์ให้น้อยที่สุด เท่าที่จะเป็นไปได้ เท่าที่จำเป็นต่อการใช้งานเท่านั้น มีข้อควรระวังดังต่อไปนี้

1. นอกจากผู้ดูแลฐานข้อมูลแล้ว ไม่ควรให้ผู้ใช้งานคนอื่นๆ เข้าถึงตาราง user ของฐานข้อมูล
2. ผู้ดูแลฐานข้อมูลควรศึกษาถึงระบบการให้สิทธิ์การเข้าถึงฐานข้อมูลโดยละเอียด
3. รหัสผ่านที่ใช้งานจะต้องเป็นรหัสผ่านที่ดี และการเก็บค่ารหัสผ่านจะต้องได้รับการเข้ารหัส
4. ถ้ามีการส่งข้อมูลผ่านอินเทอร์เน็ต ข้อมูลที่ส่งจะต้องได้รับการเข้ารหัสทุกครั้ง
5. ตรวจสอบการส่งข้อมูลโดยใช้คำสั่ง tcpdump และ strings ของระบบปฏิบัติการ
6. ต้องกำหนดให้ผู้ใช้งานฐานข้อมูลทุกคนมีรหัสผ่านในการเข้าใช้งาน

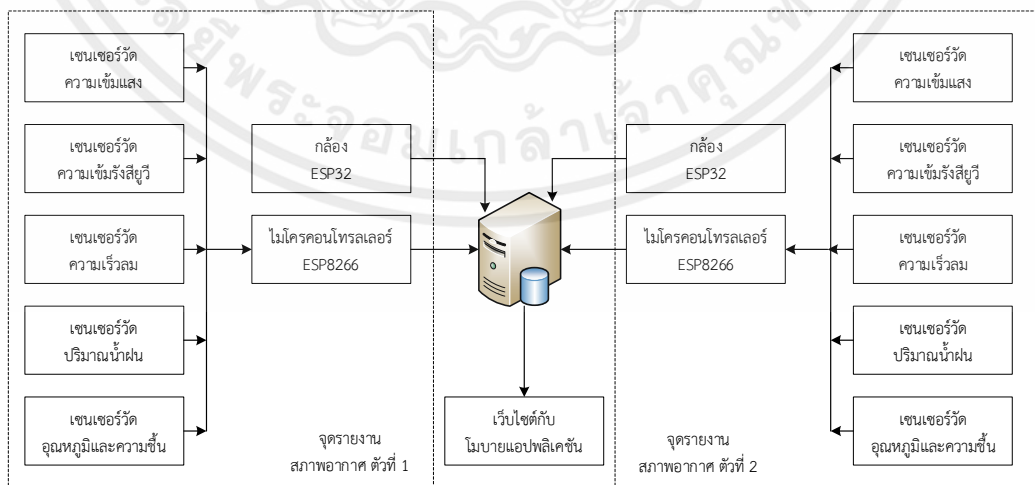
## บทที่ 3

### หลักการและการออกแบบ

ในบทนี้จะกล่าวถึง ภาพรวมของระบบ การออกแบบโครงสร้างจตุรายนงานสภาพอากาศ การออกแบบเชื่อมต่อวงจรเซนเซอร์ การประกอบเสาจตุรายนงานสภาพอากาศ การติดตั้งจตุรายนงานสภาพอากาศในสถานที่จริง การออกแบบและสร้างฐานข้อมูล แผนผังการทำงานของโปรแกรมและซอร์สโค้ด และการสร้างเว็บไซต์และโมบายแอปพลิเคชัน

#### 3.1 ภาพรวมของระบบ (Block Diagram)

จตุรายนงานสภาพอากาศจะมีอยู่ 2 จุด ซึ่งแต่ละที่จะมีอุปกรณ์อิเล็กทรอนิกส์ การทำงานและคุณสมบัติเหมือนกันทุกอย่าง หลักการทำงานของจตุรายนงานสภาพอากาศเริ่มจากเซนเซอร์ทั้ง 5 ชนิด นั่นคือ เซนเซอร์วัดความเข้มแสง เซนเซอร์วัดความเข้มรังสียูวี เซนเซอร์วัดความเร็วลม เซนเซอร์วัดปริมาณน้ำฝน ตัวสุดท้ายคือเซนเซอร์วัดอุณหภูมิและความชื้นในอากาศ จะส่งค่าที่วัดได้ไปให้ไมโครคอนโทรลเลอร์ ESP8266 เพื่อส่งค่าทั้งหมดไปเก็บไว้ในฐานข้อมูล ในขณะที่เดียวกันกล่อง ESP32 จะถ่ายรูปสถานที่จริงทุกๆ 1 ชั่วโมงแล้วส่งไฟล์รูปไปเก็บไว้ในฐานข้อมูลเช่นเดียวกัน ส่วนสุดท้ายคือเว็บไซต์กับโมบายแอปพลิเคชัน ซึ่งจะทำหน้าที่แสดงข้อมูลทั้งให้ผู้ใช้งานได้รับทราบ ข้อมูลที่นำแสดงก็จะเรียกมาจากฐานข้อมูล ผู้ใช้งานจะเลือกดูข้อมูลได้ 2 สถานที่ แต่ละที่จะแสดงข้อมูลลักษณะเดียวกันนั่นคือ ความเข้มแสง ความเข้มรังสียูวี ความเร็วลม ปริมาณน้ำฝน อุณหภูมิและความชื้นในอากาศ ซึ่งจะเป็นข้อมูลของแต่ละสถานที่ ผู้ใช้งานก็เลือกได้ว่าจะดูข้อมูลจากจตุรายนงานสภาพอากาศตัวไหน ภาพรวมของระบบจตุรายนงานสภาพอากาศมีลักษณะ ดังแสดงไว้ในรูปที่ 3.1

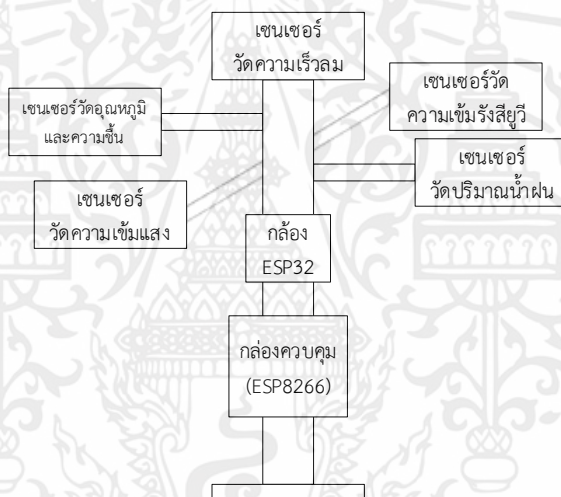


รูปที่ 3.1 ภาพรวมของระบบจตุรายนงานสภาพอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

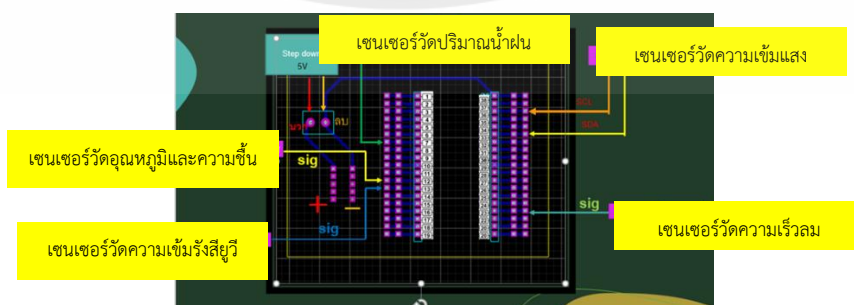
### 3.2 การออกแบบโครงสร้างจตุรกายงานสภาพอากาศ

แนวคิดในการออกแบบจตุรกายงานสภาพอากาศ คือการนำเซนเซอร์ทั้งหมด กล้อง ESP32 และไมโครคอนโทรลเลอร์ ESP8266 ติดตั้งไว้บนเสาตัวเดียวกัน แล้วจึงนำเสาดังนี้ไปติดตั้งตามสถานที่ที่ต้องการ เพื่อให้เป็นอุปกรณ์วัดสภาพอากาศและถ่ายรูปตามที่เรียกชื่อว่า จตุรกายงานสภาพอากาศ เซนเซอร์ตัวที่อยู่ด้านบนสุดจะต้องเป็นเซนเซอร์วัดความเร็วลม เพราะจะไม่โดนเสาหรือเซนเซอร์ตัวอื่นๆ ขวางทิศทางลม ส่วนเซนเซอร์ตัวอื่นๆ ก็จะติดตั้งบนแขนของเสาโดยติดลดหลั่นกันลงมาแต่ต้องไม่กีดขวางกัน เซนเซอร์วัดปริมาณน้ำฝนเป็นเซนเซอร์ที่ต้องติดตั้งให้ได้ระดับแนวอนที่แม่นยำ (ระดับน้ำเสมอกัน) เพราะจะทำให้วัดปริมาณน้ำฝนได้แม่นยำ พร้อมด้วยการติดตั้งกล้อง ESP32 เอาไว้ตรงกลางเสา (เรียกว่าด้านหน้า) ส่วนสุดท้ายต้องมีกล่องพลาสติกกันน้ำเพื่อใส่ ESP8266 ซึ่งเป็นตัวประมวลผลเอาไว้ในกล่อง ด้วยแนวคิดที่กล่าวมาทั้งหมด ทำให้ผู้จัดทำสร้างเสาสำหรับเป็นจตุรกายงานสภาพอากาศ ดังรูปที่ 3.2



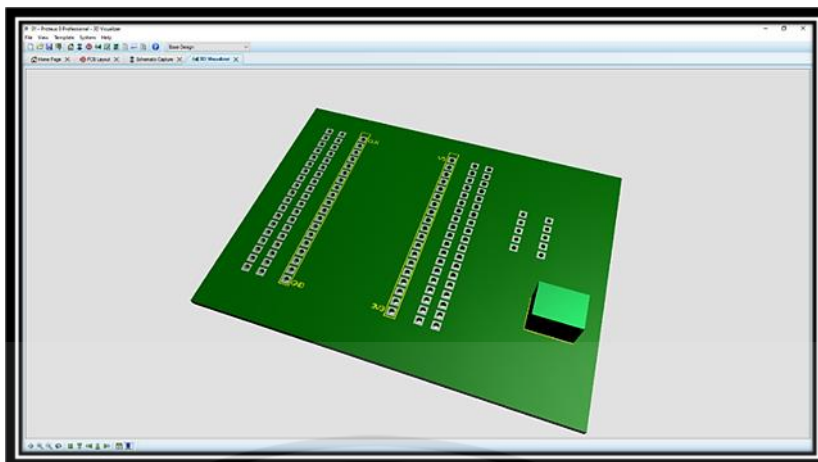
รูปที่ 3.2 การออกแบบเสาของจตุรกายงานสภาพอากาศ

### 3.3 การออกแบบเชื่อมต่อวงจรเซนเซอร์



รูปที่ 3.3 การออกแบบวงจรเพื่อเชื่อมต่อเซนเซอร์ทุกตัว

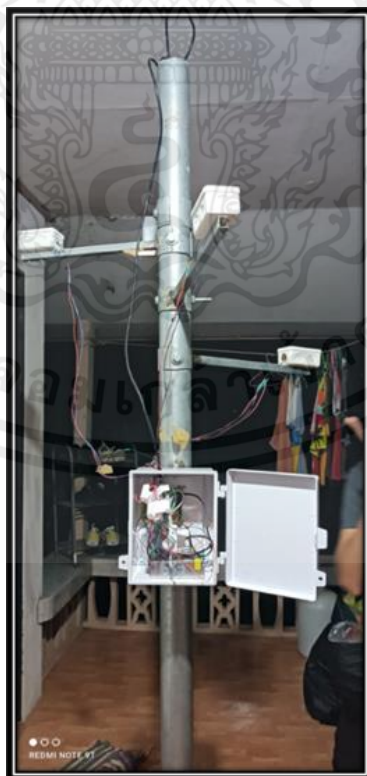
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 รูปของวงจรพีซีบีของไมโครคอนโทรลเลอร์ ESP8266

จากรูปที่ 3.3 และ 3.4 เป็นการออกแบบวงจรสำหรับการเชื่อมต่อเซนเซอร์ทุกชนิดให้เข้ากับไมโครคอนโทรลเลอร์ ESP8266 โดยการใช้โปรแกรม Proteus 2019 ในการออกแบบ เมื่อได้รูปแบบวงจรพีซีบีแล้ว จากนั้นผู้จัดทำจะนำไปทำวงจรจริงด้วยแผ่นทองแดง

### 3.4 การประกอบเสาจุดรายงานสภาพอากาศ



รูปที่ 3.5 การประกอบเสาจุดรายงานสภาพอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 การประกอบภายในกล่องควบคุมอุปกรณ์



รูปที่ 3.7 รูปด้านหลังของเสาจุดรายงานสภาพอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.5-3.7 คือการประกอบเซนเซอร์ทุกชนิดให้เชื่อมต่อกับ ESP8266 และได้ติดตั้งกล่อง ESP32 จากนั้นนำอุปกรณ์ต่างๆ ติดตั้งไว้ตรงตำแหน่งที่ได้ออกแบบไว้ สำหรับภายในกล่องควบคุมจะมี ESP8266 เก็บเอาไว้ในกล่องพร้อมด้วยแหล่งจ่ายแรงดันไฟฟ้ากระแสตรงแบบสวิตซิ่ง

### 3.5 การติดตั้งจุดรายงานสภาพอากาศในสถานที่จริง

การติดตั้งจุดรายงานสภาพอากาศในเบื้องต้นจะติดตั้งที่หน้าหอพักของจัดทำและข้างหอพักของอาจารย์ที่ปรึกษาก่อน เพื่อเป็นการทดสอบระบบและเก็บผลการทดลอง สาเหตุที่ติดตั้งไว้ใกล้กับที่พักอาศัยเพราะจะทำให้ซ่อมบำรุงได้ง่ายและมีสัญญาณอินเทอร์เน็ตจากไว-ไฟ เมื่อทดสอบระบบนี้เป็นที่แน่ใจแล้วว่าอุปกรณ์ต่างๆ ทำงานได้อย่างแน่นอน ผู้จัดทำก็จะย้ายจุดรายงานสภาพอากาศไปยังสถานที่จริง นั่นคือหาดพระจอม

#### 3.5.1 การติดตั้งจุดรายงานสภาพอากาศ ตัวที่ 1 (หน้าหอพักของผู้จัดทำ)



รูปที่ 3.8 ด้านหน้าของจุดรายงานสภาพอากาศ ตัวที่ 1

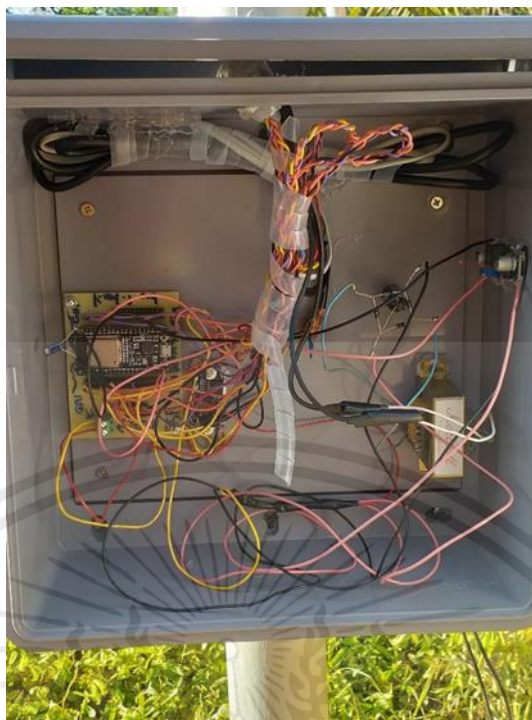


รูปที่ 3.9 ด้านข้างของจุดรายสภาพอากาศ ตัวที่ 1



รูปที่ 3.10 ด้านหน้าของจุดรายสภาพอากาศ ตัวที่ 1 แบบถ่ายระยะใกล้

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**รูปที่ 3.11** ภายในกล่องควบคุมของจุดรายงานสภาพอากาศ ตัวที่ 1

จากรูปที่ 3.8-3.11 คือรูปการติดตั้งจุดรายงานสภาพอากาศที่บริเวณหอพักของผู้จัดทำ ซึ่งสะดวกต่อการบำรุงรักษาและเพื่อทดสอบระบบและเก็บผลการทดลอง อีกทั้งยังแก้ไขข้อผิดพลาดได้อย่างง่ายดายและรวดเร็ว

### 3.5.2 การติดตั้งจุดรายงานสภาพอากาศ ตัวที่ 2 (ด้านข้างหอพักของอาจารย์)

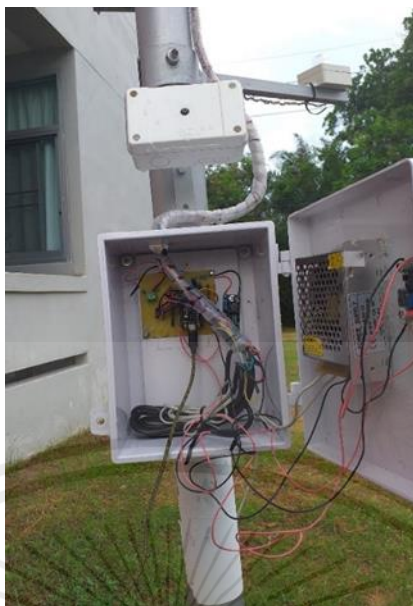


รูปที่ 3.12 ด้านหน้าของจุดรายสภาพอากาศ ตัวที่ 2



รูปที่ 3.13 ด้านหลังของจุดรายสภาพอากาศ ตัวที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 ภายในกล่องควบคุมของจุดรายงานสภาพอากาศ ตัวที่ 2

จากรูปที่ 3.12-3.14 แสดงให้เห็นการติดตั้งจุดรายงานสภาพอากาศ ตัวที่ 2 ซึ่งเหตุผลที่เลือกบริเวณนี้สำหรับการทดสอบระบบนั้นคือ มีสัญญาณไว-ไฟที่ดี เข้าซ่อมบำรุง แก้ไขข้อผิดพลาดได้สะดวก และง่ายต่อการเปลี่ยนต่อย้ายสถานที่

### 3.5.3 การติดตั้งจุดรายงานสภาพอากาศ ตัวที่ 1 (ณ หากพระจอม)



รูปที่ 3.15 การติดตั้งจุดรายงานสภาพอากาศที่บริเวณหาดพระจอม

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.15 เป็นการเคลื่อนย้ายจตุรกายงานสภาพอากาศ ตัวที่ 1 ไปยังหาพระจอมเพื่อเก็บผลการทดลองและเป็นการทดสอบระบบ สาเหตุที่ผู้จัดทำเลือกตำแหน่งตามรูปเพราะว่ามีปลั๊กไฟซึ่งสามารถใช้ไฟฟ้าได้อย่างสะดวก อีกทั้งยังสี่สัญญาณไว-ไฟจากบ้านพักของอาจารย์อีกด้วย

### 3.6 การออกแบบและสร้างฐานข้อมูล

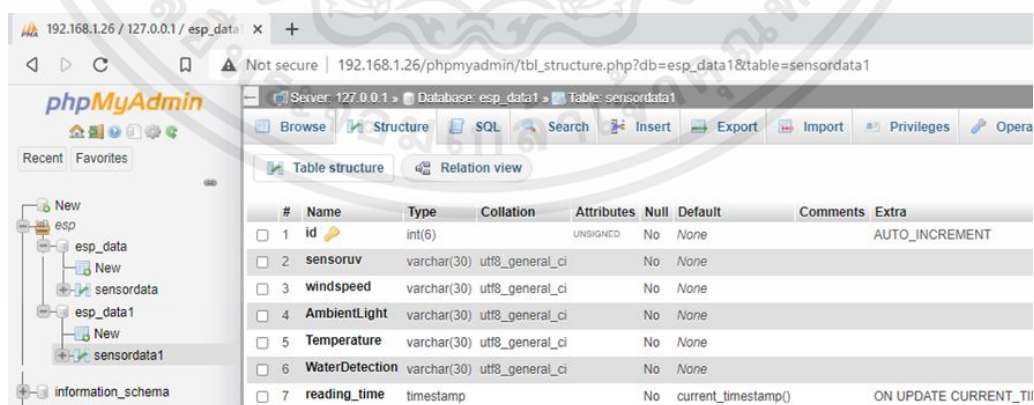
แนวคิดในการออกแบบฐานข้อมูลจะมาจากจำนวนของจตุรกายงานสภาพอากาศและจำนวนของอุปกรณ์ ร่วมกับวันและเวลาที่เก็บข้อมูลด้วย ซึ่งจะทำให้ผู้จัดทำได้ออกแบบฐานข้อมูลดังต่อไปนี้

1. ในฐานข้อมูล 1 ตัวจะแบ่งออกเป็น 2 ตาราง
  - 1) ตารางของจตุรกายงานสภาพอากาศ ตัวที่ 1
  - 2) ตารางของจตุรกายงานสภาพอากาศ ตัวที่ 2
2. ในแต่ละตารางจะมีทั้งหมด 7 คอลัมน์ หรือเรียกว่า 7 ฟิลด์ (Field) ได้แก่
  - 1) id กำหนดรหัสให้กับข้อมูลที่เข้ามา
  - 2) sensoruv เก็บข้อมูลมาจากเซนเซอร์วัดความเข้มรังสียูวี
  - 3) windspeed เก็บข้อมูลมาจากเซนเซอร์วัดความเร็วลม
  - 4) AmbientLight เก็บข้อมูลมาจากเซนเซอร์วัดความเข้มแสง
  - 5) Temperature เก็บข้อมูลมาจากเซนเซอร์วัดอุณหภูมิและความชื้น
  - 6) WaterDetection เก็บข้อมูลมาจากเซนเซอร์วัดปริมาณน้ำฝน
  - 7) reading\_time วันและเวลาที่บันทึกข้อมูล

จากแนวคิดที่ได้ ผู้จัดทำจึงนำไปออกแบบและสร้างฐานข้อมูล โดยใช้โปรแกรม MySQL เป็นโปรแกรมสร้างฐานข้อมูล และใช้โปรแกรม phpMyAdmin สำหรับจัดการฐานข้อมูล

#### 3.6.1 ฐานข้อมูลสำหรับจตุรกายงานสภาพอากาศ ตัวที่ 1 และ 2

1. การสร้างฐานข้อมูลเพื่อใช้เก็บข้อมูลจากจตุรกายงานสภาพอากาศ ตัวที่ 1



รูปที่ 3.16 การสร้างตารางเก็บข้อมูลสำหรับจตุรกายงานสภาพอากาศ ตัวที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	id	sensoruv	windspeed	AmbientLight	Temperature	WaterDetection	reading_time
<input type="checkbox"/>	667	0.00	0	0.00	26.37	0.00000000	2021-06-04 00:00:04
<input type="checkbox"/>	668	0.00	0	0.00	26.12	0.00000000	2021-06-04 01:00:04
<input type="checkbox"/>	669	0.00	0	0.00	25.50	0.00000000	2021-06-04 03:00:04
<input type="checkbox"/>	670	0.00	0	0.00	26.06	0.00000000	2021-06-04 13:02:03
<input type="checkbox"/>	671	0.00	0	0.00	26.75	0.00000000	2021-06-04 07:00:05
<input type="checkbox"/>	672	0.65	0	0.00	33.00	0.00000000	2021-06-04 13:01:44
<input type="checkbox"/>	673	2.70	0	0.00	33.44	0.00000000	2021-06-04 09:00:03
<input type="checkbox"/>	674	4.25	0	0.00	35.31	0.00000000	2021-06-04 12:00:04
<input type="checkbox"/>	675	1.12	0	44.94	35.06	0.00000000	2021-06-05 01:28:38
<input type="checkbox"/>	676	0.00	0	2.70	31.19	0.00000000	2021-06-05 01:28:54
<input type="checkbox"/>	677	0.00	0	0.00	29.81	0.00000000	2021-06-04 18:59:49
<input type="checkbox"/>	678	0.00	0	0.00	29.81	0.00000000	2021-06-05 01:28:17
<input type="checkbox"/>	679	0.00	0	0.00	27.44	0.00000000	2021-06-04 22:59:49
<input type="checkbox"/>	680	0.00	0	0.00	26.87	0.00000000	2021-06-04 23:59:49
<input type="checkbox"/>	681	0.00	0	0.00	26.44	0.00000000	2021-06-05 00:59:49
<input type="checkbox"/>	682	0.00	0	0.00	26.06	0.00000000	2021-06-05 01:59:49
<input type="checkbox"/>	683	0.00	0	0.00	25.88	0.00000000	2021-06-05 03:59:49
<input type="checkbox"/>	684	0.00	0	0.00	25.50	0.00000000	2021-06-05 04:59:49
<input type="checkbox"/>	685	0.62	0	21.43	33.44	0.00600000	2021-06-05 20:38:39
<input type="checkbox"/>	686	0.02	0	13.47	29.37	0.00000000	2021-06-05 15:59:50
<input type="checkbox"/>	687	0.00	0	0.00	29.00	0.00000000	2021-06-05 20:38:30

รูปที่ 3.17 ข้อมูลที่ได้จากจตุรายนสภาพอากาศ ตัวที่ 1

## 2. การสร้างฐานข้อมูลเพื่อใช้เก็บข้อมูลจากจตุรายนสภาพอากาศ ตัวที่ 2

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	int(6)		UNSIGNED	No	None		AUTO_INCREMENT
2	UV	varchar(30)	utf8_general_ci		Yes	NULL		
3	windspeed	varchar(30)	utf8_general_ci		Yes	NULL		
4	AmbientLight	varchar(10)	utf8_general_ci		Yes	NULL		
5	Temperature	varchar(10)	utf8_general_ci		Yes	NULL		
6	WaterDetection	varchar(10)	utf8_general_ci		Yes	NULL		
7	reading_time	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP

รูปที่ 3.18 การสร้างตารางเก็บข้อมูลสำหรับจตุรายนสภาพอากาศ ตัวที่ 2

จากรูปที่ 3.16 และ 3.18 คือตารางเก็บข้อมูลของจตุรายนสภาพอากาศ ตัวที่ 1 และ 2 ตามลำดับ ส่วนตารางที่ 3.17 และ 3.19 เป็นตัวอย่างข้อมูลที่เก็บไว้ในตารางของฐานข้อมูลของจตุรายนสภาพอากาศ ตัวที่ 1 และ 2 ตามลำดับเช่นกัน

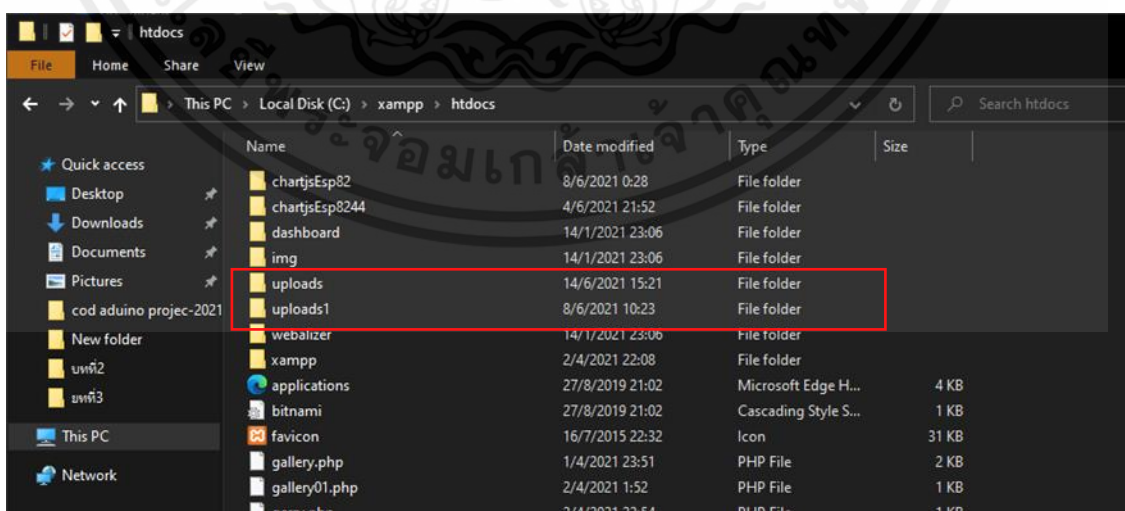
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

+ Options		id	UV	windspeed	AmbientLight	Temperature	WaterDetection	reading_time
☐ Edit Copy Delete	1747	0.31	0	4.99	31.80	0.00000000	2021-06-03 09:31:38	
☐ Edit Copy Delete	1748	0.64	0	8.29	32.90	0.00000000	2021-06-03 10:31:37	
☐ Edit Copy Delete	1749	0.31	0	7.25	33.00	0.00000000	2021-06-03 11:31:37	
☐ Edit Copy Delete	1750	6.68	0	8.12	36.20	0.00000000	2021-06-03 12:31:37	
☐ Edit Copy Delete	1751	5.25	0	54.61	37.30	0.00000000	2021-06-03 13:31:40	
☐ Edit Copy Delete	1752	2.47	0	54.61	37.60	0.00000000	2021-06-03 14:31:37	
☐ Edit Copy Delete	1753	0.15	0	51.44	37.30	0.00000000	2021-06-03 15:31:37	
☐ Edit Copy Delete	1754	1.11	0	29.11	37.40	0.00000000	2021-06-03 16:31:37	
☐ Edit Copy Delete	1755	0.00	0	10.55	34.40	0.00000000	2021-06-03 17:31:37	
☐ Edit Copy Delete	1756	0.00	0	0.51	30.80	0.00000000	2021-06-03 18:31:38	
☐ Edit Copy Delete	1757	0.00	0	0.00	29.40	0.00000000	2021-06-03 19:31:37	
☐ Edit Copy Delete	1758	0.00	0	0.00	28.10	0.00000000	2021-06-03 20:31:37	
☐ Edit Copy Delete	1759	0.00	0	0.00	27.80	0.00000000	2021-06-03 21:31:37	
☐ Edit Copy Delete	1760	0.00	0	0.00	27.50	0.00000000	2021-06-03 22:31:38	
☐ Edit Copy Delete	1761	0.00	0	0.00	30.00	0.00000000	2021-06-05 08:44:40	
☐ Edit Copy Delete	1762	1.35	1	21.02	34.00	0.00000000	2021-06-05 11:51:51	
☐ Edit Copy Delete	1763	0.00	0	2.10	29.00	0.00000000	2021-06-05 18:13:02	
☐ Edit Copy Delete	1764	0.00	0	0.00	28.00	0.00000000	2021-06-05 19:58:47	
☐ Edit Copy Delete	1765	0.00	0	0.00	28.00	0.00000000	2021-06-05 20:58:48	
☐ Edit Copy Delete	1766	0.00	0	0.00	27.00	0.00000000	2021-06-05 21:58:48	
☐ Console Copy Delete	1767	0.00	0	0.00	27.00	0.00000000	2021-06-05 22:58:49	

รูปที่ 3.19 ข้อมูลที่ได้จากจตุรกายงานสภาพอากาศ ตัวที่ 2

### 3.6.2 ฐานข้อมูลเพื่อใช้เก็บภาพถ่ายจากกล้อง ESP32

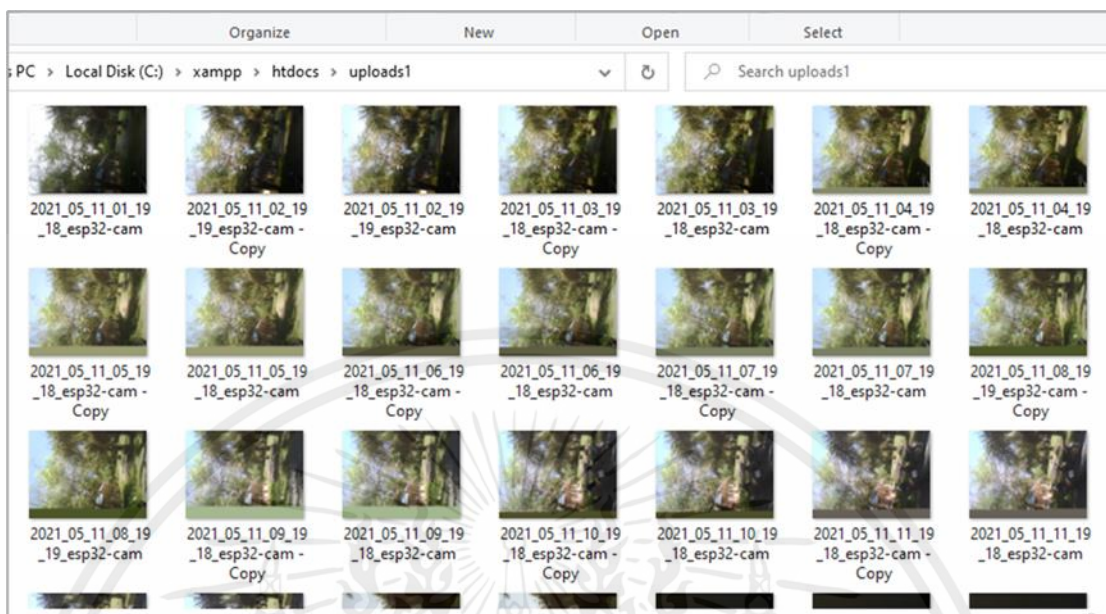
รูปถ่ายที่ได้จากกล้อง ESP32 จะส่งไปเก็บไว้ในฐานข้อมูลโดยตรง เนื่องจาก ESP32 ก็สามารถเชื่อมต่อไอ-ไฟได้เช่นกัน โพลเตอร์ที่เก็บไฟล์รูปภาพจะมีอยู่ 2 โพลเตอร์ คือโพลเตอร์ uploads1 จะเก็บไฟล์รูปจากจตุรกายงานสภาพอากาศ ตัวที่ 1 และโพลเตอร์ uploads จะเก็บไฟล์รูปจากจตุรกายงานสภาพอากาศ ตัวที่ 2 ดังแสดงไว้ในรูปที่ 3.20



รูปที่ 3.20 โพลเตอร์สำหรับเก็บไฟล์รูปไว้ในฐานข้อมูล

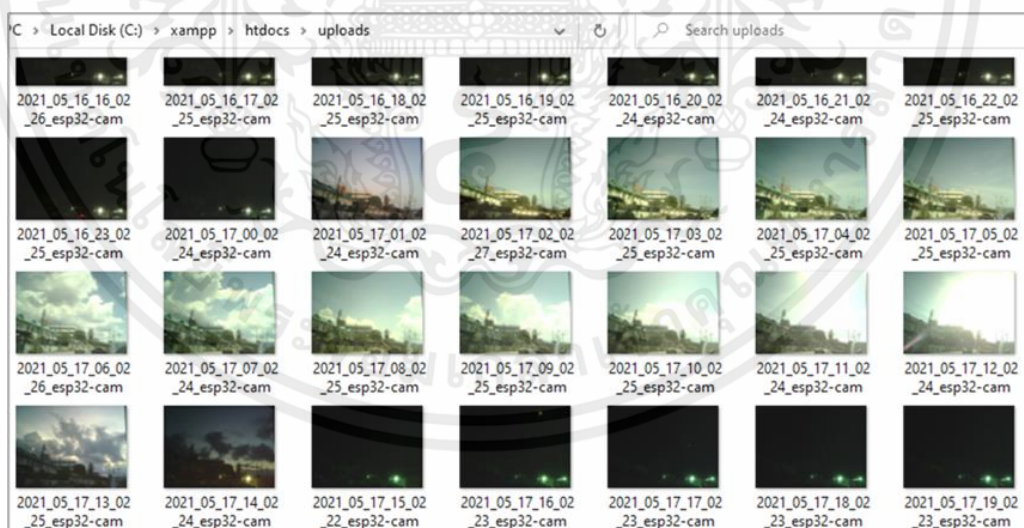
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 1. ไฟล์รูปที่ได้จากจตุรายนงานสภาพอากาศ ตัวที่ 1



รูปที่ 3.21 การเก็บไฟล์รูปจากจตุรายนงานสภาพอากาศ ตัวที่ 1

### 2. ไฟล์รูปที่ได้จากจตุรายนงานสภาพอากาศ ตัวที่ 2



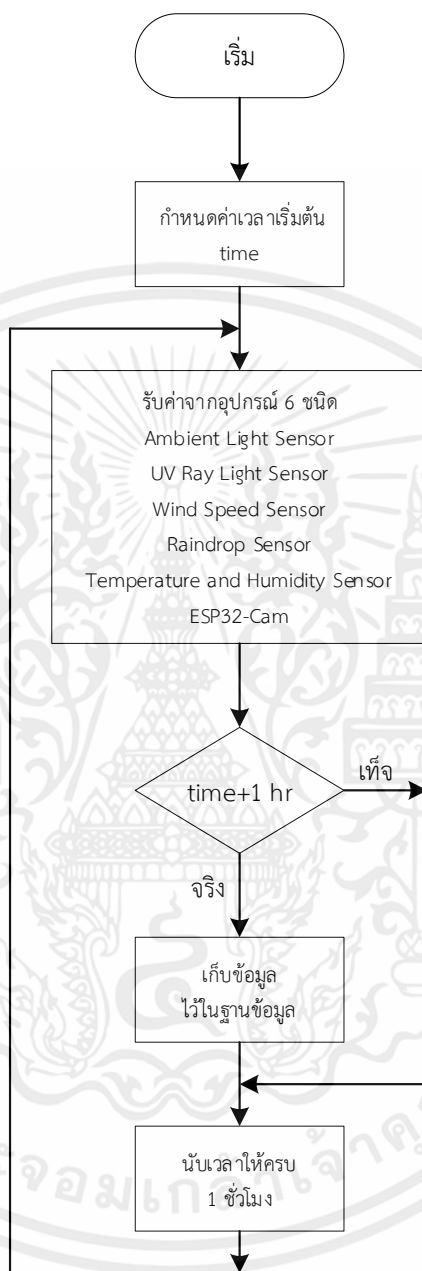
รูปที่ 3.22 การเก็บไฟล์รูปจากจตุรายนงานสภาพอากาศ ตัวที่ 2

จากรูปที่ 3.21 และ 3.22 คือรูปภาพที่กล้อง ESP32 ของจตุรายนงานสภาพอากาศ ตัวที่ 1 และ 2 ถ่ายเก็บเป็นไฟล์เอาไว้ในฐานข้อมูล ซึ่งรูปที่ได้เกิดจากการถ่ายรูปทุกๆ 1 ชั่วโมง

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.7 แผนผังการทำงานของโปรแกรมและซอร์สโค้ด (Flowchart and Source Code)

#### 3.7.1 แผนผังการทำงานของโปรแกรม



รูปที่ 3.23 แผนผังการทำงานของโปรแกรมของจุดรายงานสภาพอากาศ

จากรูปที่ 3.23 คือแผนผังการทำงานของโปรแกรมสำหรับจุดรายงานสภาพอากาศทั้ง 2 ตัว ซึ่งมีลักษณะการทำงานเหมือนกันทุกประการ การทำงานเริ่มจากเมื่อเปิดสวิตช์จุดรายงานสภาพอากาศแล้ว ตัวเครื่องจะทำการตั้งเวลาเริ่มต้น (เพื่อให้รู้ว่าอีก 1 ชั่วโมงข้างหน้าเป็นเวลาเท่าไร)

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นรับค่าจากอุปกรณ์ทุกตัวแล้วนำข้อมูลเหล่านี้ไปเก็บไว้ในฐานข้อมูล ต่อจากนั้นจะรอเวลาอีก 1 ชั่วโมงจึงจะรับค่าจากอุปกรณ์ทุกตัวอีกครั้ง และทำงานวนรอบแบบนี้ไปตลอด

### 3.7.2 ซอร์สโค้ดสำหรับเซนเซอร์ของจตุรายนสภาพอากาศ

```

WRP01.cpp
1 #ifndef ESP32
2 #include <WiFi.h>
3 #include <HTTPClient.h>
4 #include "DHT.h"
5 #include "RTCLib.h"
6 #include <time.h>
7 #else
8 #include <ESP8266WiFi.h>
9 #include <ESP8266HTTPClient.h>
10 #include <WiFiClient.h>
11 #endif
12 #include <Wire.h>
13 #include <Adafruit_Sensor.h>
14 #include <BH1750FVI.h> //Ligh
15 #define DHTPIN 14 //Temp and Humid
16 #define DHTTYPE DHT11 //Temp and Humid
17 #define RainPin 32 //Rain
18
19 DHT dht(DHTPIN,DHTTYPE); //Temp and Humid
20 BH1750FVI LightSensor(BH1750FVI :: k_DevModeContLowRes); //Ligh
21 RTC_Millis rtc; //rain
22 const char* ssid = "GNT-00";
23 const char* password = "zgmFx2oa";
24 const String month_name[12] = {"Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"};
25 const String day_name[7] = {"Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"};
26 const char* serverName = "http://1d9e9a95d2c.ngrok.io/post-esp-data1.php";
27 String apiKeyValue = "tPmAT5Ab3j7F8";
28 int timezone = 7 * 3600; //TimeZone
29 int dst = 0; //Date Swing Time
30 bool bucketPositionA = false; //Rain
31 const double bucketAmount = 0.01610595; //Rain
32 double dailyRain = 0.0; //Rain
33 double hourlyRain = 0.0; //Rain
34 double dailyRain_till_LastHour = 0.0; //Rain
35 bool first;
36 int UVOUT = 35; //UV
37 int REF_3V3 = 2; //UV
38

```

รูปที่ 3.24 ซอร์สโค้ดของจตุรายนสภาพอากาศ ส่วนที่ 1

```

39 int averageAnalogRead(int pinToRead)
40 {
41     byte numberOfReadings = 8;
42     unsigned int runningValue = 0;
43     for(int x = 0; x < numberOfReadings; x++)
44         runningValue += analogRead(pinToRead);
45     runningValue /= numberOfReadings;
46     return(runningValue);
47 }
48
49 float mapfloat(float x, float in_min, float in_max, float out_min, float out_max)
50 {
51     return (x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min;
52 }
53 void reedSwitch_ISR() //rain
54 {
55     bucketPositionA=true;
56     dailyRain+=bucketAmount;
57 }
58

```

รูปที่ 3.25 ซอร์สโค้ดของจตุรายนสภาพอากาศ ส่วนที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

59 void setup()
60 {
61   Serial.begin(115200);
62   Serial.setDebugOutput(true);
63   WiFi.mode(WIFI_STA); //WiFi
64   WiFi.begin(ssid, password);
65   pinMode(34, INPUT);
66   pinMode(UVOUT, INPUT); //uv
67   pinMode(REF_3V3, INPUT); //uv
68   dht.begin();
69   LightSensor.begin(); //Ligh
70   rtc.begin(DateTime(_DATE, _TIME_)); //rain
71   pinMode(RainPin, INPUT_PULLUP); //rain
72   attachInterrupt(digitalPinToInterrupt(RainPin), readSwitch_ISR, FALLING); //rain
73   Serial.println("Connecting");
74
75   while(WiFi.status() != WL_CONNECTED)
76   {
77     Serial.print(".");
78     delay(100);
79   }
80
81   configTime(timezone, dst, "pool.ntp.org", "time.nist.gov");
82   Serial.println("");
83   Serial.print("Connected to WiFi network with IP Address: ");
84   Serial.println(WiFi.localIP());
85   while (!time(nullptr))
86   {
87     Serial.print("**");
88     delay(100);
89   }
90   // (you can also pass in a Wire library object like &Wire2)
91   // bool status = bme.begin(0x76);
92 }

```

รูปที่ 3.26 ซอร์สโค้ดของจตุรายนงานสภาพอากาศ ส่วนที่ 3

```

93 void loop()
94 {
95   //delay(100);
96   //DateTime now = rtc.now(); //rain
97   time_t now = time(nullptr);
98   struct tm* p_tm = localtime(&now);
99   float lux = LightSensor.GetLightIntensity(); //Ligh
100  int uvLevel = averageAnalogRead(UVOUT); //uv
101  int refLevel = averageAnalogRead(REF_3V3); //uv
102  float outputVoltage = 3.3 * uvLevel/4095; //uv
103  float uvIntensity = mapfloat(outputVoltage, 0.99, 2.8, 0.0, 15.0); //uv
104  if (uvIntensity < 0)
105  {
106    uvIntensity = 0;
107  }
108  int sensorValue = analogRead(34);
109  float outvoltage = sensorValue * (3.3/4095);
110  int level = 6*outvoltage;
111  float h = dht.readHumidity();
112  float t = dht.readTemperature();
113  float f = dht.readTemperature(true);
114  if (uvIntensity <= 0)
115  {
116    float uvIntensity = 0;
117  }
118  //Serial.print("Light: ");
119  //Serial.print(lux);
120  //Serial.println(" Lux");
121  Serial.print("uvLevel=" + (String)uvLevel + " ");
122  Serial.print("voltage=" + (String)outputVoltage + " ");
123  Serial.print("uv=" + (String)uvIntensity + "(mW/cm^2)");
124  Serial.println();
125  Serial.print(p_tm->tm_hour);
126  Serial.print(":");
127  Serial.print(p_tm->tm_min);
128  Serial.print(":");
129  Serial.print(p_tm->tm_sec);
130  Serial.println();
131  delay(1000);

```

รูปที่ 3.27 ซอร์สโค้ดของจตุรายนงานสภาพอากาศ ส่วนที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

132 if((bucketPositionA == true) && (digitalRead(RainPin) == LOW))
133 bucketPositionA=false; //rain
134 if(p_tm->tm_sec!= 0)
135 first = true; //rain
136 if (WiFi.status()== WL_CONNECTED)
137 {
138     if(p_tm->tm_sec == 0 && p_tm->tm_min == 0 && first == true)
139     {
140         hourlyRain = dailyRain - dailyRain_till_LastHour; // rain
141         dailyRain_till_LastHour = dailyRain; // rain
142
143
144         HTTPClient http;
145         http.begin(serverName);
146         http.addHeader("Content-Type", "application/x-www-form-urlencoded");
147         String httpRequestData = "api_key = " + apiKeyValue + "&sensoruv="(String)uvIntensity
148             + "&windspeed="+Level+"&AmbientLight="+String(Lux/1000)
149             + "&Temperature=" + t + " &WaterDetection=" +String(hourlyRain,0)+ " ";
150         Serial.print("httpRequestData: ");
151         Serial.println(httpRequestData);
152         int httpResponseCode = http.POST(httpRequestData);
153         delay(900);
154         if (httpResponseCode>0)
155         {
156             Serial.print("HTTP Response code: ");
157             Serial.println(httpResponseCode);
158         }
159         else
160         {
161             Serial.print("Error code: ");
162             Serial.println(httpResponseCode);
163         }
164         http.end();
165         first = false;
166     }
167 }

```

### รูปที่ 3.28 ซอร์สโค้ดของจตุรรายงานสภาพอากาศ ส่วนที่ 5

```

168 else
169 {
170     Serial.println("WiFi Disconnected");
171 }
172 if(p_tm->tm_hour==0&&p_tm->tm_sec==0)
173 {
174     dailyRain = 0.0; //rain
175     dailyRain_till_LastHour = 0.0; //rain
176 }
177 }

```

### รูปที่ 3.29 ซอร์สโค้ดของจตุรรายงานสภาพอากาศ ส่วนที่ 6

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.24-3.29 คือซอร์สโค้ดสำหรับควบคุมให้เซนเซอร์ทั้ง 5 ชนิดให้เชื่อมต่อกับ ESP8266 แล้วทำงานได้อย่างอัตโนมัติ ซึ่งซอร์สโค้ดชุดนี้เป็นซอร์สโค้ดของจตุรายนสภาพอากาศ ตัวที่ 1 หากแต่ซอร์สโค้ดของจตุรายนสภาพอากาศ ตัวที่ 1 กับตัวที่ 2 ก็มีลักษณะใกล้เคียงกันมาก ผู้เขียนจะขออธิบายการทำงานของซอร์สโค้ดอย่างพอสังเขป อย่างเป็นข้อๆ ดังต่อไปนี้

1. รูปที่ 3.24 เป็นการเรียกใช้งานไลบรารี (Library) ที่เกี่ยวข้อง พร้อมด้วย การประกาศตัวแปรที่ใช้งานได้ทุกฟังก์ชัน (ตัวแปรแบบ Global)

2. รูปที่ 3.25 เป็นการสร้างฟังก์ชันสำหรับเรียกใช้งาน มีอยู่ 3 ฟังก์ชันนั่นคือ การอ่านข้อมูลแบบแอนะล็อก การหาค่าสูงสุดและต่ำสุด ฟังก์ชันสุดท้ายคือการอ่านค่าจากเซนเซอร์วัดปริมาณน้ำฝน

3. รูปที่ 3.26 เป็นการกำหนดค่าเริ่มต้นให้กับเซนเซอร์ การเชื่อมต่อไว-ไฟ และการกำหนดวัน/เวลาเริ่มต้น

4. รูปที่ 3.27-3.29 เป็นการทำงานอัตโนมัติของระบบนั่นคือ การเก็บค่าจากเซนเซอร์ชนิดต่างๆ การถ่ายรูปแล้วนำข้อมูลทั้งหมดที่ได้ไปเก็บไว้ในฐานข้อมูล ซึ่งระบบจะทำงานอยู่อย่างนี้ไปเรื่อยๆ จนกว่าจะปิดสวิทซ์การทำงาน

### 3.7.3 ซอร์สโค้ดสำหรับกล่อง ESP32 ของจตุรายนสภาพอากาศ

```

WRP02_ESP32Cam.cpp
1 #include<Arduino.h>
2 #include<WiFi.h>
3 #include<time.h>
4 #include"soc/soc.h"
5 #include"soc/rtc_cntl_reg.h"
6 #include"esp_camera.h"
7 #include"RTCLib.h"
8 const String month_name[12] = {"Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"};
9 const String day_name[7] = {"Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"};
10 int timezone = 7 * 3600;
11 int dst = 0;
12 bool first, bucketPositionA = false;
13 RTC_Millis rtc;
14 const char* ssid = "GNT-00";
15 const char* password = "zgmfx2oa";
16 String serverName = "f3f94cbcd7a5.ngrok.io"; //String serverName = "example.com";
17 String serverPath = "/upload1.php"; // The default serverPath should be upload.php
18 const int serverPort = 80;
19 WiFiClient client;
20 // CAMERA_MODEL_AI_THINKER
21 #define PWDN_GPIO_NUM 32
22 #define RESET_GPIO_NUM -1
23 #define XCLK_GPIO_NUM 0
24 #define SIOD_GPIO_NUM 26
25 #define SIOC_GPIO_NUM 27
26 #define V9_GPIO_NUM 35
27 #define V8_GPIO_NUM 34
28 #define V7_GPIO_NUM 39
29 #define V6_GPIO_NUM 36
30 #define V5_GPIO_NUM 21
31 #define V4_GPIO_NUM 19
32 #define V3_GPIO_NUM 18
33 #define V2_GPIO_NUM 5
34 #define VSYNC_GPIO_NUM 25
35 #define HREF_GPIO_NUM 23
36 #define PCLK_GPIO_NUM 22
37 const int timerInterval = 30000; // time between each HTTP POST image
38 unsigned long previousMillis = 0; // Last time image was sent

```

### รูปที่ 3.30 ซอร์สโค้ดสำหรับกล่อง ESP32 ของจตุรายนสภาพอากาศ ส่วนที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

39 void setup()
40 {
41   WRITE_PERI_REG(RTC_CNTL_BROWN_OUT_REG, 0);
42   Serial.begin(115200);
43   Serial.setDebugOutput(true);
44   #ifndef ESP8266
45   while (!Serial); // wait for serial port to connect. Needed for native USB
46   #endif
47   rtc.begin(DateTime(F(__DATE__), F(__TIME__)));
48   WiFi.mode(WIFI_STA);
49   Serial.println();
50   Serial.print("Connecting to ");
51   Serial.println(ssid);
52   WiFi.begin(ssid, password);
53   while (WiFi.status() != WL_CONNECTED)
54   {
55     Serial.print(".");
56     delay(500);
57   }
58   configTime(timezone, dst, "pool.ntp.org", "time.nist.gov"); //*,**@k Server
59   Serial.println("Loading time");
60   Serial.println();
61   Serial.print("ESP32-CAM IP Address: ");
62   Serial.println(WiFi.localIP());
63   while (!time(nullptr))
64   {
65     Serial.print("**");
66     delay(1000);
67   }

```

รูปที่ 3.31 ซอร์สโค้ดสำหรับกล้อง ESP32 ของจตุรายนงานสภาพอากาศ ส่วนที่ 2

```

68 camera_config_t config;
69 config.ledc_channel = LEDC_CHANNEL_0;
70 config.ledc_timer = LEDC_TIMER_0;
71 config.pin_d0 = Y2_GPIO_NUM;
72 config.pin_d1 = Y3_GPIO_NUM;
73 config.pin_d2 = Y4_GPIO_NUM;
74 config.pin_d3 = Y5_GPIO_NUM;
75 config.pin_d4 = Y6_GPIO_NUM;
76 config.pin_d5 = Y7_GPIO_NUM;
77 config.pin_d6 = Y8_GPIO_NUM;
78 config.pin_d7 = Y9_GPIO_NUM;
79 config.pin_xclk = XCLK_GPIO_NUM;
80 config.pin_pclk = PCLK_GPIO_NUM;
81 config.pin_vsync = VSYNC_GPIO_NUM;
82 config.pin_href = HREF_GPIO_NUM;
83 config.pin_sscb_sda = SIOD_GPIO_NUM;
84 config.pin_sscb_scl = SIOC_GPIO_NUM;
85 config.pin_pwdn = PWDN_GPIO_NUM;
86 config.pin_reset = RESET_GPIO_NUM;
87 config.xclk_freq_hz = 20000000;
88 config.pixel_format = PIXFORMAT_JPEG;
89 // init with high specs to pre-allocate larger buffers
90 if(psramFound())
91 {
92   config.frame_size = FRAMESIZE_SVGA;
93   config.jpeg_quality = 10; //0-63 lower number means higher quality
94   config.fb_count = 2;
95 }
96 else
97 {
98   config.frame_size = FRAMESIZE_CIF;
99   config.jpeg_quality = 12; //0-63 lower number means higher quality
100   config.fb_count = 1;
101 }

```

รูปที่ 3.32 ซอร์สโค้ดสำหรับกล้อง ESP32 ของจตุรายนงานสภาพอากาศ ส่วนที่ 3

```

102 // camera init
103 esp_err_t err = esp_camera_init(&config);
104 if (err != ESP_OK)
105 {
106   Serial.printf("Camera init failed with error 0x%x", err);
107   delay(1000);
108   ESP.restart();
109 }
110 sendPhoto();
111 }
112 void loop()
113 {
114   DateTime now = rtc.now();
115   unsigned long currentMillis = millis();
116   Serial.print(now.hour(), DEC);
117   Serial.print(":");
118   Serial.print(now.minute(), DEC);
119   Serial.print(":");
120   Serial.print(now.second(), DEC);
121   Serial.println();
122   delay(1000);
123   if (currentMillis - previousMillis >= timerInterval)
124   {
125     if(now.second() == 0 && now.minute()==0)
126     {
127       sendPhoto();
128       previousMillis = currentMillis;
129     }
130     first = false;
131   }
132 }

```

รูปที่ 3.33 ซอร์สโค้ดสำหรับกล้อง ESP32 ของจตุรายนงานสภาพอากาศ ส่วนที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

133 String sendPhoto()
134 {
135     String getAll;
136     String getBody;
137     camera_fb_t * fb = NULL;
138     fb = esp_camera_fb_get();
139     if(!fb)
140     {
141         Serial.println("Camera capture failed");
142         delay(1000);
143         ESP.restart();
144     }
145     Serial.println("Connecting to server: " + serverName);
146     if (client.connect(serverName.c_str(), serverPort))
147     {
148         Serial.println("Connection successful!");
149         String head = "--RandomNerdTutorials\r\nContent-Disposition: form-data; name=\"imageFile\"; filename=\"esp32-cam.jp
150         String tail = "\r\n--RandomNerdTutorials--\r\n";
151         uint32_t imageLen = fb->len;
152         uint32_t extraLen = head.length() + tail.length();
153         uint32_t totalLen = imageLen + extraLen;
154         client.println("POST " + serverPath + " HTTP/1.1");
155         client.println("Host: " + serverName);
156         client.println("Content-Length: " + String(totalLen));
157         client.println("Content-Type: multipart/form-data; boundary=RandomNerdTutorials");
158         client.println();
159         client.print(head);
160         uint8_t *fbBuf = fb->buf;
161         size_t fbLen = fb->len;

```

รูปที่ 3.34 ซอร์สโค้ดสำหรับกล้อง ESP32 ของจตุรายนงานสภาพอากาศ ส่วนที่ 5

```

162     for (size_t n=0; n<fbLen; n+=1024)
163     {
164         if (n+1024 < fbLen)
165         {
166             client.write(fbBuf, 1024);
167             fbBuf += 1024;
168         }
169         else if (fbLen-1024>0)
170         {
171             size_t remainder = fbLen-1024;
172             client.write(fbBuf, remainder);
173         }
174     }
175     client.print(tail);
176     esp_camera_fb_return(fb);
177     int timeoutTimer = 10000;
178     long startTimer = millis();
179     boolean state = false;
180     while ((startTimer + timeoutTimer) > millis())
181     {
182         Serial.print(",");
183         delay(100);
184         while (client.available())
185         {
186             char c = client.read();
187             if (c == '\n')
188             {
189                 if (getAll.length()==0)
190                 {
191                     state=true;
192                 }
193                 getAll = "";
194             }
195             else if (c != '\r')
196             {
197                 getAll += String(c);
198             }

```

รูปที่ 3.35 ซอร์สโค้ดสำหรับกล้อง ESP32 ของจตุรายนงานสภาพอากาศ ส่วนที่ 6

```

199         if (state==true)
200         {
201             getBody += String(c);
202         }
203         startTimer = millis();
204     }
205     if (getBody.length(>>0))
206     {
207         break;
208     }
209 }
210 Serial.println();
211 client.stop();
212 Serial.println(getBody);
213 }
214 else
215 {
216     getBody = "Connection to " + serverName + " failed.";
217     Serial.println(getBody);
218 }
219 return getBody;
220 }

```

รูปที่ 3.36 ซอร์สโค้ดสำหรับกล้อง ESP32 ของจตุรายนงานสภาพอากาศ ส่วนที่ 7

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

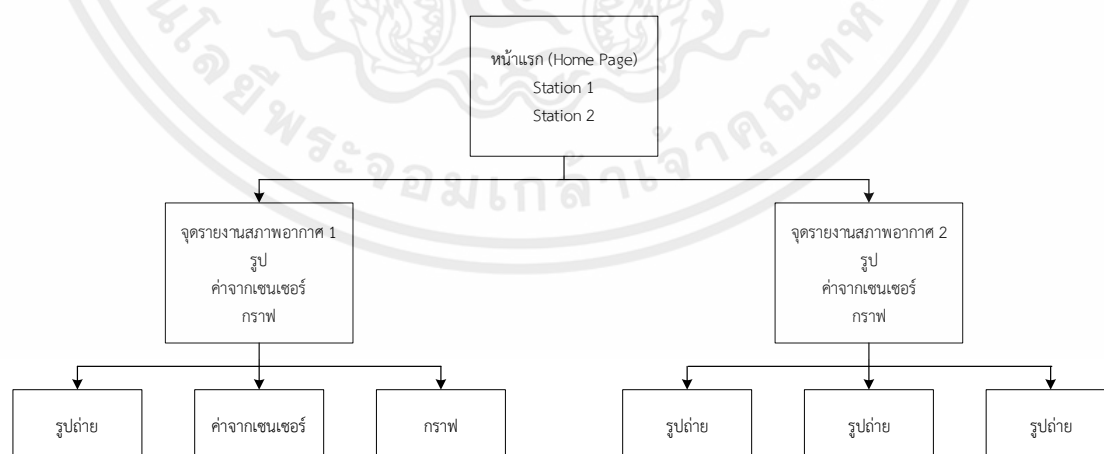
จากรูปที่ 3.30-3.36 คือซอร์สโค้ดสำหรับควบคุมให้กล้อง ESP32 ถ่ายรูปตามที่โปรแกรมที่กำหนดไว้อย่างอัตโนมัติ และกล้อง ESP32 ที่ติดตั้งอยู่บนจุดรายงานสภาพอากาศทั้ง 2 ตัว ซึ่งลักษณะของซอร์สโค้ดก็จะเหมือนกัน แตกต่างกันแค่ส่วนของคำสั่งให้รูปไปเก็บไว้ที่โฟลเดอร์ไหน ผู้จัดทำจะขออธิบายการทำงานของซอร์สโค้ดโดยสรุป ดังต่อไปนี้

1. รูปที่ 3.30 เป็นการประกาศใช้งานไลบรารี การประกาศค่าตัวแปรที่ใช้งานได้ทุกฟังก์ชันและตัวแปรแบบค่าคงที่
2. รูปที่ 3.31-3.32 รวมทั้งรูปที่ 3.33 ถึงแค่บรรทัดที่ 111 เป็นการกำหนดขบวนการทำงานสำหรับรับและส่งข้อมูล รวมทั้งการเชื่อมต่อไวไฟ เพื่อให้ ESP32 สามารถส่งรูปถ่ายไปเก็บไว้ในฐานข้อมูลได้
3. รูปที่ 3.33 บรรทัดที่ 112-132 เป็นการสั่งให้ทำการนับเวลาและทำการถ่ายรูป แล้วนำไปเก็บไว้ฐานข้อมูลโดยอัตโนมัติ
4. รูปที่ 3.34-3.36 เป็นการเขียนฟังก์ชันเพื่อให้ระบบทำการถ่ายรูป ซึ่งฟังก์ชันนี้จะถูกเรียกใช้งานมาจาก void loop()

### 3.8 การสร้างเว็บไซต์และโมบายแอปพลิเคชัน

เว็บไซต์และโมบายแอปพลิเคชันที่ได้สร้างขึ้นมาในโครงการนี้ มีจุดประสงค์อย่างเดียวกันั้นคือการแสดงข้อมูลสภาพอากาศของบริเวณที่ติดตั้งจุดรายงานสภาพอากาศ ซึ่งเว็บไซต์และโมบายแอปพลิเคชันจะมีคุณสมบัติเหมือนกัน แตกต่างกันแค่สร้างการแสดงผลข้อมูลให้เหมาะสมกับอุปกรณ์ที่ผู้ใช้งานต้องการเรียกดูข้อมูล

#### 3.8.1 แผนผังเว็บไซต์และโมบายแอปพลิเคชัน



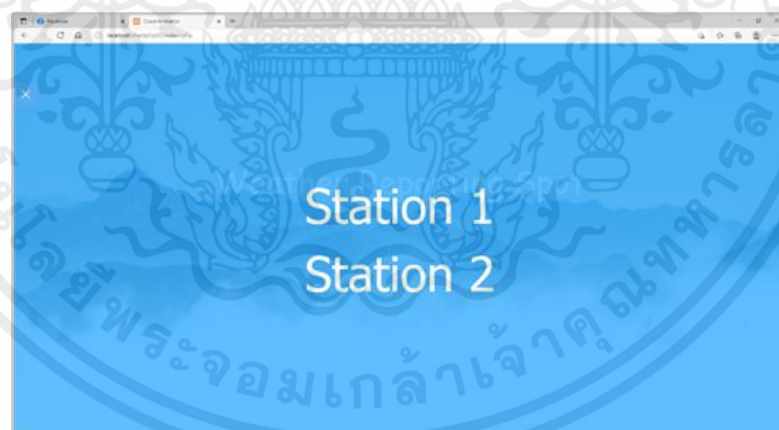
รูปที่ 3.37 แผนผังเว็บไซต์และโมบายแอปพลิเคชัน

จากรูปที่ 3.37 อธิบายได้ว่า เมื่อผู้ใช้งานเข้าสู่หน้าแรกของเว็บไซต์และโมบายแอปพลิเคชัน จะสามารถเลือกได้ว่าจะดูข้อมูลของจุดรายงานสภาพอากาศ ตัวที่ 1 หรือ 2 เมื่อเลือกแล้วว่าจะดูข้อมูลที่จุดใด จากนั้นจะพบว่าผู้ใช้สามารถเลือกดูข้อมูลได้ 3 อย่างคือ รูป ค่าที่วัดได้จากเซนเซอร์ และดูข้อมูลเป็นกราฟ หากผู้ใช้งานเลือกดูข้อมูลอย่างใดอย่างหนึ่งก็จะเข้าไปสู่หน้าต่างที่บอกรายละเอียดของข้อมูลแต่ละแบบอย่างชัดเจน

### 3.8.2 เว็บไซต์สำหรับรายงานสภาพอากาศ

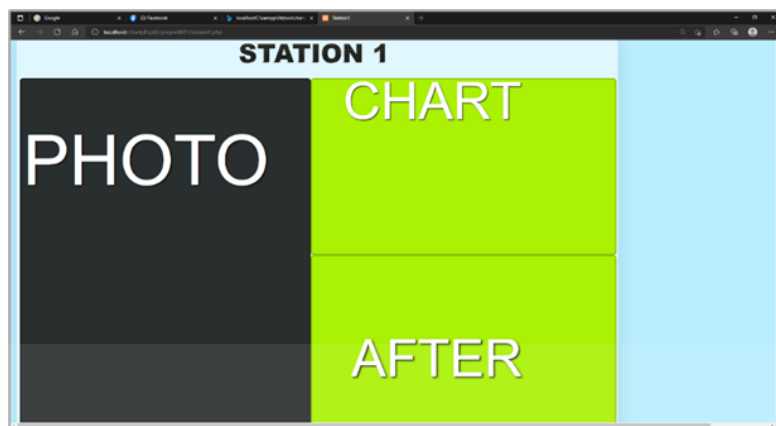


รูปที่ 3.38 หน้าแรกของเว็บไซต์

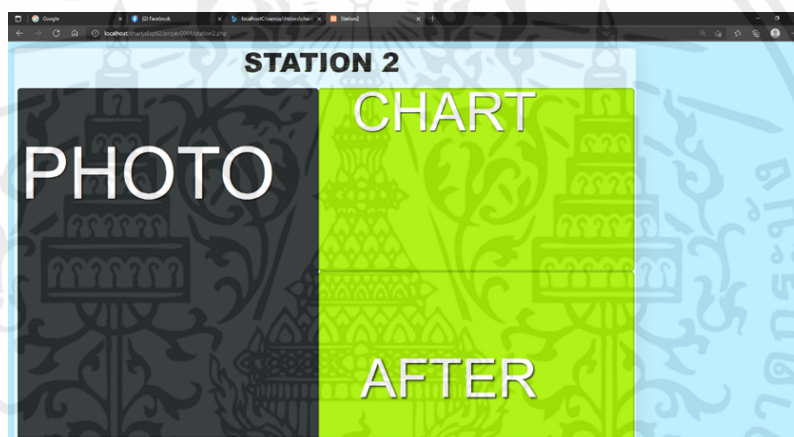


รูปที่ 3.39 การเลือกดูข้อมูลจากจุดรายงานสภาพอากาศ ตัวที่ 1 หรือ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.40 เมนูสำหรับเลือกดูข้อมูลของจตุรายนงานสภาพอากาศ ตัวที่ 1

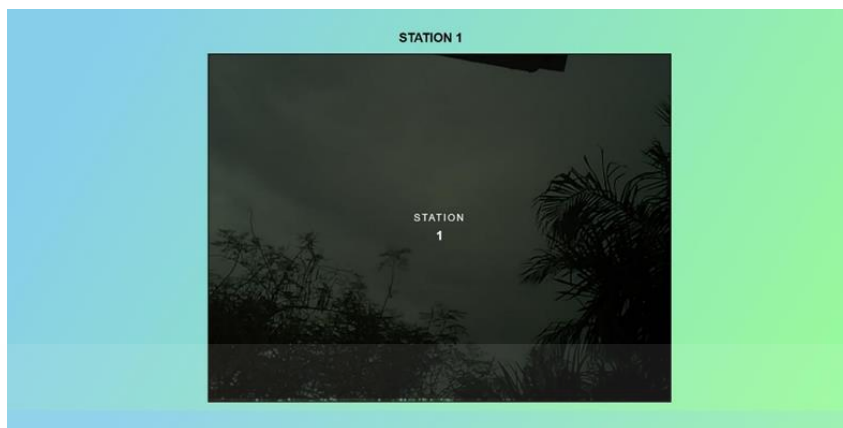


รูปที่ 3.41 เมนูสำหรับเลือกดูข้อมูลของจตุรายนงานสภาพอากาศ ตัวที่ 2

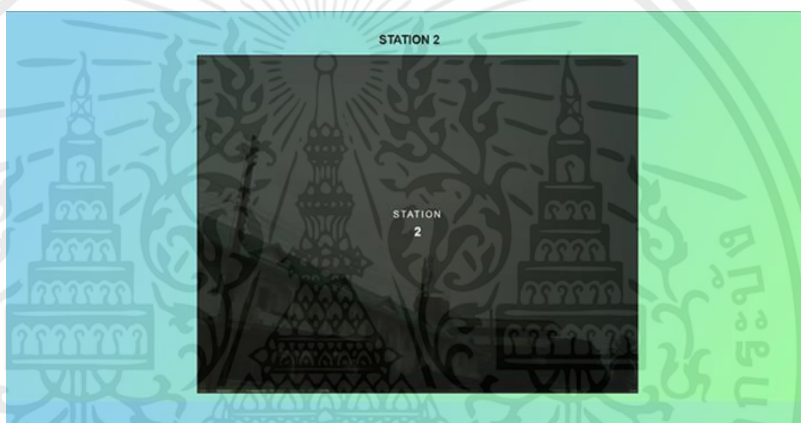
จากรูปที่ 3.40-3.41 เป็นหน้าต่างเมนูที่จะแสดงข้อมูลสามส่วนซึ่งสามารถเลือกได้ว่าจะดูข้อมูลส่วนไหน หากที่คำว่า PHOTO จะเป็นการเลือกดูภาพถ่ายทางอากาศปัจจุบันที่ใช้เวลาในการถ่ายทุกๆ 1 ชั่วโมง ในส่วนของส่วนของ CHART จะเป็นการดูการส่งค่าของเซนเซอร์ทั้ง 5 ชนิดนั้นคือ เซนเซอร์วัดความเข้มแสง เซนเซอร์วัดความเข้มรังสียูวี เซนเซอร์วัดอุณหภูมิและความชื้น เซนเซอร์ความเร็วลม และเซนเซอร์วัดปริมาณน้ำฝนซึ่งเป็นข้อมูลล่าสุดไม่เกิน 1 ชั่วโมงที่ผ่านมา และส่วนสุดท้ายคือ AFTER เป็นการเลือกดูค่าจากเซนเซอร์ทั้ง 5 ชนิดในรูปแบบกราฟ ซึ่งจะดูข้อมูลย้อนหลังได้ถึง 7 วัน

1. กรณีที่ผู้ใช้งานเลือกเมนู PHOTO ก็จะสามารถดูรูปถ่ายจากสถานที่เหล่านั้นได้ทันที ซึ่งเป็นรูปปัจจุบันไม่เกิน 1 ชั่วโมงที่ผ่านมา ดังแสดงไว้ในรูปที่ 3.42-3.43

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.42 รูปถ่ายจากจุดรายงานสภาพอากาศ ตัวที่ 1



รูปที่ 3.43 รูปถ่ายจากจุดรายงานสภาพอากาศ ตัวที่ 2

2. กรณีที่ผู้ใช้งานเลือกเมนู CHART ผู้ใช้งานจะสามารถดูข้อมูลจากเซนเซอร์ทั้ง 5 ชนิดในรูปแบบกราฟได้ ซึ่งจะเป็นกราฟที่แสดงข้อมูลช่วง 7 วันที่ผ่านมา ในเมนูนี้ยังมีวิธีการดูข้อมูล หรือเรียกว่า คู่มือในการดูข้อมูล ให้ผู้ใช้งานได้เข้าใจอย่างสะดวกอีกด้วย ดังรูปที่ 3.44-3.45

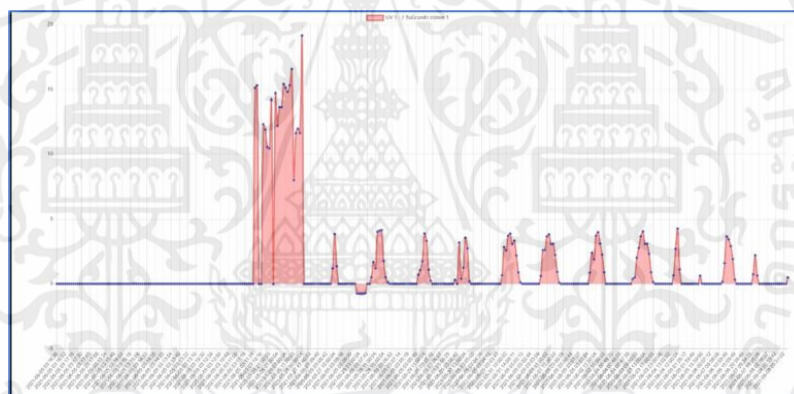


รูปที่ 3.44 หน้าต่างข้อมูลของเซนเซอร์แบบกราฟจากจุดรายงานสภาพอากาศ ตัวที่ 1

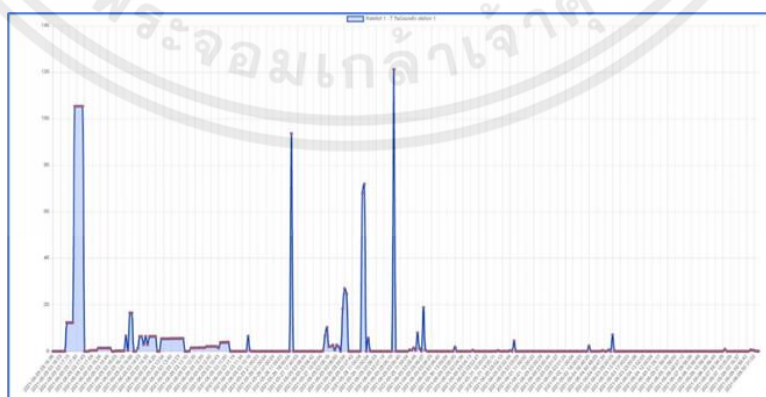
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.45 หน้าต่างข้อมูลของเซนเซอร์แบบกราฟจากจุดรายงานสภาพอากาศ ตัวที่ 2  
จากรูปที่ 3.46-3.50 เป็นการแสดงข้อมูลที่วัดค่าได้จากเซนเซอร์ทุกประเภทของจุด  
รายงานสภาพอากาศ ซึ่งจะแสดงเป็นกราฟย้อนหลัง 7 วัน โดยแบ่งเป็นกราฟของเซนเซอร์แต่ละชนิด

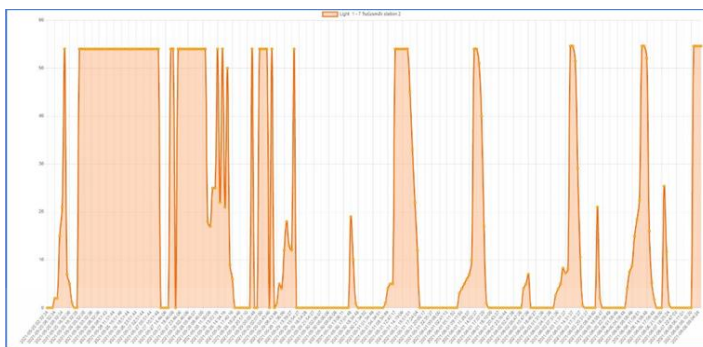


รูปที่ 3.46 กราฟวัดค่าความชื้นรังสีย้อนหลังเจ็ดวัน

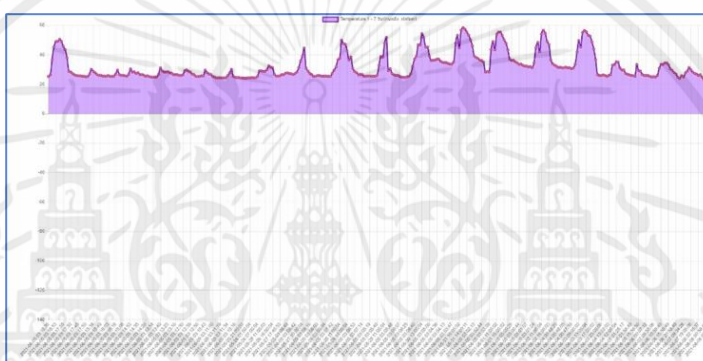


รูปที่ 3.47 กราฟวัดค่าปริมาณน้ำฝนย้อนหลังเจ็ดวัน

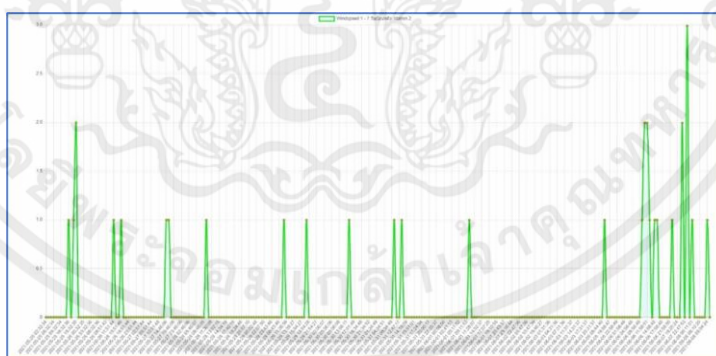
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.48 กราฟวัดค่าความเข้มแสงย้อนหลังเจ็ดวัน



รูปที่ 3.49 กราฟวัดค่าอุณหภูมิและความชื้นย้อนหลังเจ็ดวัน



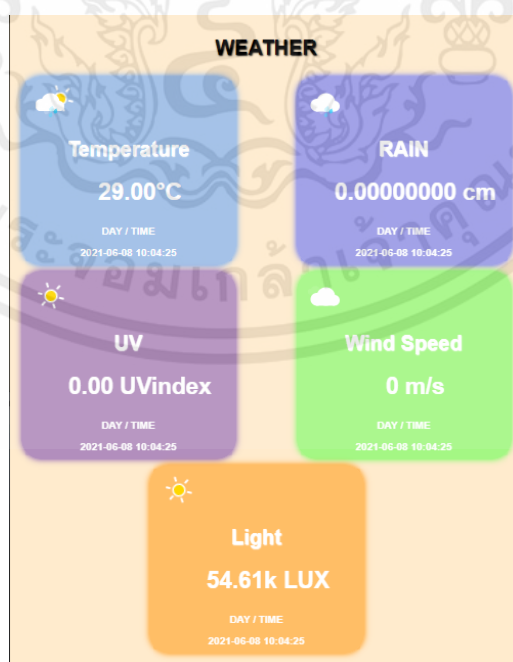
รูปที่ 3.50 กราฟวัดค่าความเร็วลมย้อนหลังเจ็ดวัน

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. กรณีที่ผู้ใช้งานเลือกเมนู AFTER ก็จะเข้าสู่การรายงานสภาพอากาศด้วยเซนเซอร์ ทั้ง 5 ชนิด ซึ่งเป็นข้อมูลที่วัดได้ในเวลาไม่เกิน 1 ชั่วโมง ดังรูปที่ 3.51-3.52



รูปที่ 3.51 หน้าต่างข้อมูลของเซนเซอร์จากจตุรรายงานสภาพอากาศ ตัวที่ 1



รูปที่ 3.52 หน้าต่างข้อมูลของเซนเซอร์จากจตุรรายงานสภาพอากาศ ตัวที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หากผู้ใช้งานคลิกไปที่ไอคอนที่แสดงอยู่บนหน้าเว็บ (จากรูปที่ 3.51-3.52) ก็จะแสดงคู่มือการทำความเข้าใจในระดับของการอ่านค่าจากเซนเซอร์ว่า ค่าที่วัดได้อยู่ในระดับใด โดยปรากฏเป็นป๊อปอัพขึ้นมาให้เห็นอย่างชัดเจน (Pop-up) ดังรูปที่ 3.53-3.57



รูปที่ 3.53 ข้อมูลระดับของปริมาณน้ำฝน



รูปที่ 3.54 ข้อมูลระดับความเข้มรังสียูวี



รูปที่ 3.55 ข้อมูลระดับอุณหภูมิ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เงื่อนไข	ฟลักซ์สว่าง (lux)
แสงแดด	107,527
กลางวันเต็มที่	10,752
วันที่ฟ้าครึ้ม	1,075
วันที่มืดมาก	107
สนธยา (Twilight) (ช่วงเวลาก่อนดวงอาทิตย์ขึ้นหรือหลังจากดวงอาทิตย์ตก)	10.8
พระจันทร์เต็มดวง	0.108
คืนที่มืดครึ้ม	0.0001

รูปที่ 3.56 ข้อมูลระดับความเข้มแสง

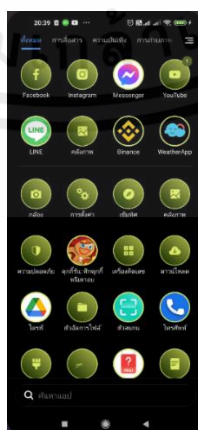


รูปที่ 3.57 ข้อมูลระดับความเร็วลม

### 3.8.3 โมบายแอปพลิเคชันสำหรับรายงานสภาพอากาศ

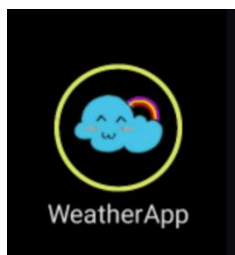
โมบายแอปพลิเคชันที่ผู้จัดทำได้สร้างเสร็จเรียบร้อยแล้ว จะเป็นไฟล์ .apk ซึ่งผู้ใช้สามารถนำไปติดตั้งในโทรศัพท์แบบสมาร์ตได้ ดังรูปที่ 3.58-3.

1. แอปพลิเคชันที่ติดตั้งเสร็จแล้วบนโทรศัพท์



รูปที่ 3.58 แอปพลิเคชันที่ติดตั้งบนโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



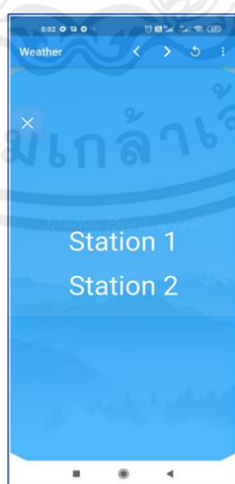
รูปที่ 3.59 ไอคอนของแอปพลิเคชัน WeatherApp

## 2. การใช้งานแอปพลิเคชัน WeatherApp

เมื่อผู้ใช้งานกดที่ไอคอนก็จะเข้าสู่หน้าแรกของแอปพลิเคชัน ที่สามารถเลือกได้ว่าจะดูข้อมูลของจุดรายงานสภาพอากาศ ตัวที่ 1 หรือ 2



รูปที่ 3.60 หน้าแรกของแอปพลิเคชัน WeatherApp



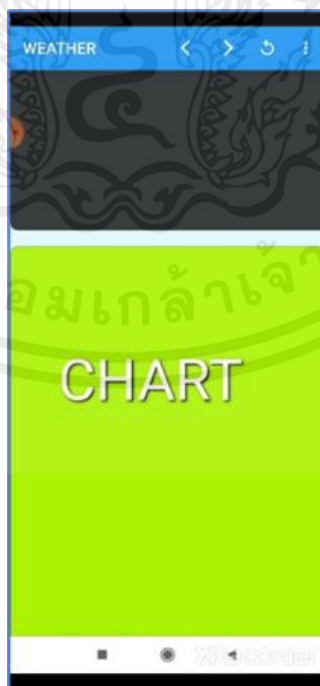
รูปที่ 3.61 การเลือกดูข้อมูลของจุดรายงานสภาพอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เมื่อผู้ใช้งานตัดสินใจได้ว่าจะดูข้อมูลจากจุดรายงานสภาพอากาศตัวใด จะปรากฏเมนู 3 อย่างนั่นคือ PHOTO, CHART และ AFTER เหมือนกับเว็บไซต์ ดังรูปที่ 3.62-3.64



รูปที่ 3.62 เมนู PHOTO



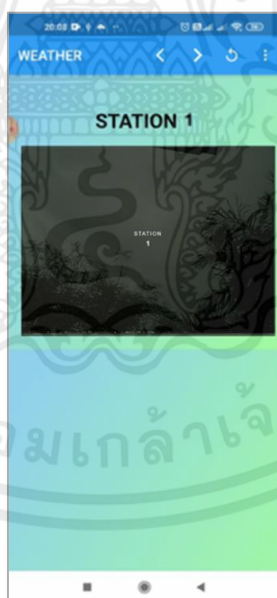
รูปที่ 3.63 เมนู CHART

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



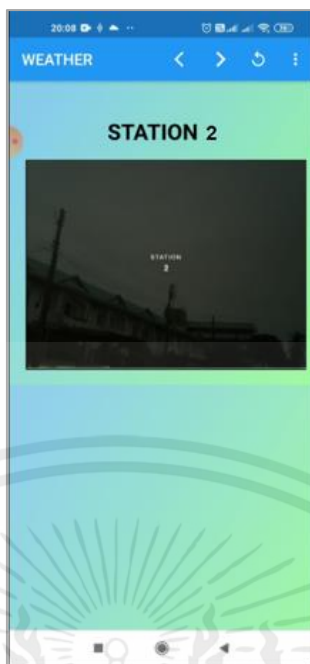
รูปที่ 3.64 เมนู AFTER

4. กรณีที่ผู้ใช้งานเลือกเมนู PHOTO ผู้ใช้งานสามารถเลือกดูรูปจากจตุรายนงานสภาพอากาศ ตัวที่ 1 หรือ 2 ได้ตามสะดวกซึ่งจะเป็นภาพถ่ายไม่เกิน 1 ชั่วโมง



รูปที่ 3.65 ภาพถ่ายจากจตุรายนงานสภาพอากาศ ตัวที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



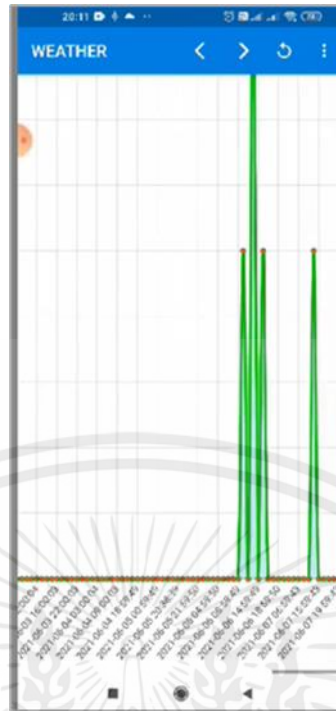
รูปที่ 3.66 ภาพถ่ายจากจุดรายงานสภาพอากาศ ตัวที่ 2

5. กรณีที่ผู้ใช้งานเลือกเมนู CHART ผู้ใช้งานสามารถเลือกดูกราฟจากจุดรายงานสภาพอากาศ ตัวที่ 1 หรือ 2 ได้ตามสะดวก เมื่อเลือกได้แล้วว่าจะดูข้อมูลจากตัวที่ 1 หรือ 2 ก็จะปรากฏเมนูให้เลือกได้อีกว่า ผู้ใช้ต้องการดูข้อมูลจากเซนเซอร์ตัวใด ก็จะแสดงข้อมูลของเซนเซอร์ตัวนั้นย้อนหลัง 7 วัน ดังรูปที่ 3.67-3.72

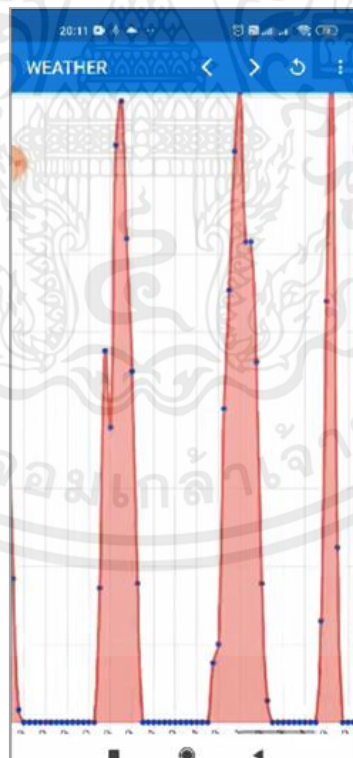


รูปที่ 3.67 หน้าแรกของกราฟข้อมูลการรายงานสภาพอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

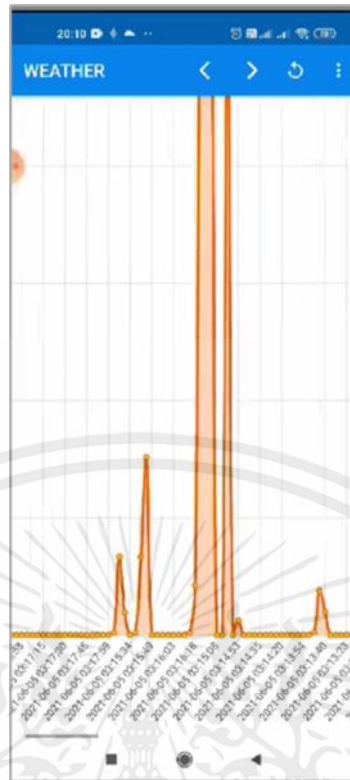


รูปที่ 3.68 กราฟความเร็วลมย้อนหลังเจ็ดวัน

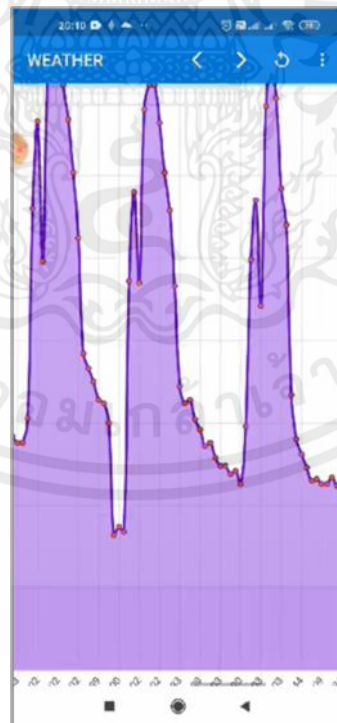


รูปที่ 3.69 กราฟความชื้นรังสีย้อนหลังเจ็ดวัน

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

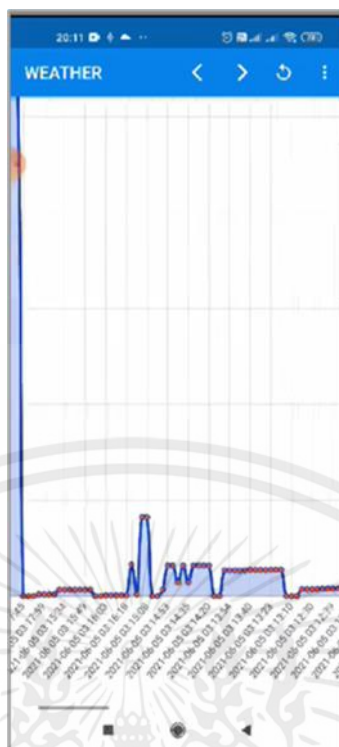


รูปที่ 3.70 กราฟความเข้มแสงย้อนหลังเจ็ดวัน



รูปที่ 3.71 กราฟพหุนหภูมิย้อนหลังเจ็ดวัน

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.72 กราฟปริมาณน้ำฝนย้อนหลังเจ็ดวัน

6. กรณีที่ผู้ใช้งานเลือกเมนู AFTER ผู้ใช้งานสามารถเลือกดูข้อมูลจากเซนเซอร์ตัวใดก็ได้จากเซนเซอร์ทั้ง 5 แบบ ดังรูปที่ 3.73-3.77



รูปที่ 3.73 การแสดงข้อมูลจากเซนเซอร์วัดอุณหภูมิและความชื้น

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

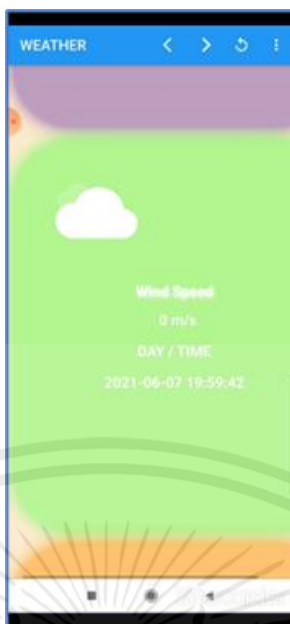


รูปที่ 3.74 การแสดงข้อมูลจากเซนเซอร์วัดปริมาณน้ำฝน



รูปที่ 3.75 การแสดงข้อมูลจากเซนเซอร์วัดความเข้มรังสียูวี

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.76 การแสดงข้อมูลจากเซนเซอร์วัดความเร็วลม



รูปที่ 3.77 การแสดงข้อมูลจากเซนเซอร์วัดความเข้มแสง

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หากผู้ใช้งานกดเข้าไปที่ไอคอนที่แสดงบนโทรศัพท์ (จากรูปที่ 3.73-3.77) ก็จะแสดงคู่มือการทำ ความเข้าใจในระดับของการอ่านค่าจากเซนเซอร์ว่า ค่าที่วัดได้อยู่ในระดับใด โดยปรากฏเป็น Pop-up ขึ้นมาให้เห็นอย่างชัดเจน ดังรูปที่ 3.78-3.82



รูปที่ 3.78 Pop-up คู่มือรายงานปริมาณน้ำฝนย้อนหลัง 7 วัน



รูปที่ 3.79 Pop-up คู่มือรายงานความชื้นสัมพัทธ์ย้อนหลัง 7 วัน

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.80 Pop-up คู่มือรายงานอุณหภูมิย้อนหลัง 7 วัน



รูปที่ 3.81 Pop-up คู่มือรายงานความเข้มแสงย้อนหลัง 7 วัน

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.82 Pop-up คู่มือรายงานความเร็วลมย้อนหลัง 7 วัน

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### ผลการทดลอง

ในบทนี้จะกล่าวถึงผลของการดำเนินงาน การเก็บรวบรวมข้อมูลและวิเคราะห์ข้อมูลของการทดสอบเซนเซอร์เพื่อวัดค่าความเข้มของรังสีอัลตราไวโอเล็ต GUVVA S12SD การทดสอบเซนเซอร์วัดความเข้มแสง BH1750 การทดสอบเซนเซอร์วัดความเร็วลม การทดสอบเซนเซอร์วัดปริมาณน้ำฝน การทดสอบเซนเซอร์วัดอุณหภูมิ การทดสอบแสดงค่าเซนเซอร์ใช้แบบร่วมกัน ทดสอบกล่อง ESP-32 CAM ทดสอบเว็บแอปพลิเคชันจุดรายงานสภาพอากาศ และการทดสอบโมบายแอปพลิเคชันจุดรายงานสภาพอากาศ ได้ข้อมูลดังต่อไปนี้

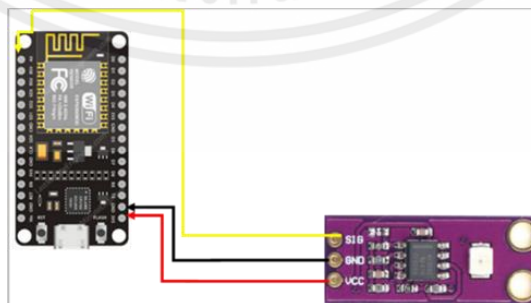
#### 4.1 การทดสอบเซนเซอร์วัดค่าความเข้มของรังสีอัลตราไวโอเล็ต GUVVA S12SD

##### 4.1.1 การทดสอบการทำงานของเซนเซอร์วัดค่าความเข้มรังสีอัลตราไวโอเล็ต วัตถุประสงค์ในการทดลอง

1. เพื่อทดสอบการแสดงค่าของเซนเซอร์ว่าแสดงค่าได้ตลอดเวลาหรือไม่
2. เพื่อทดสอบการแสดงค่าของเซนเซอร์ว่าสามารถแสดงค่าได้จริงไม่

##### ขั้นตอนการทดลอง

1. ต่อขา VCC เข้ากับแหล่งจ่ายไฟฟ้ากระแสตรงด้วยแรงดันไฟฟ้า 5 V ดังรูปที่ 4.1
2. ต่อขาสัญญาณเข้ากับพีอิน 8266 ที่ตั้งค่าไว้ให้อ่านค่าของเซนเซอร์ ดังรูปที่ 4.1
3. จากนั้นนำเซนเซอร์ไปไว้บริเวณที่แสงแจ้ง
4. สังเกตการแสดงค่าของเซนเซอร์
5. โดยใช้ระยะเวลาที่ใช้ในการทดสอบตั้งแต่วันที่ 1 พ.ค 2564 – 5 พ.ค 2564



รูปที่ 4.1 การต่อใช้งานพีอิน 8266 กับเซนเซอร์เพื่อวัดค่าความเข้มของรังสีอัลตราไวโอเล็ต  
GUVVA S12SD

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

sketch_jul14a.ino
1 void setup()
2 {
3   Serial.begin(9600);
4 }
5
6 void loop()
7 {
8   float sensorVoltage;
9   float sensorValue;
10
11  sensorValue = analogRead(A0);
12  sensorVoltage = sensorValue/1024*3.3;
13  Serial.print("sensor reading = ");
14  Serial.print(sensorValue);
15  Serial.println("");
16  Serial.print("sensor voltage = ");
17  Serial.print(sensorVoltage);
18  Serial.println(" V");
19  delay(1000);
20 }

```

รูปที่ 4.2 โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์เพื่อวัดค่าความชื้นของรังสีอัลตราไวโอเล็ต

#### GUVA S12SD

ตารางที่ 4.1 ตารางแสดงผลทดสอบเซนเซอร์เพื่อวัดค่าความชื้นของรังสีอัลตราไวโอเล็ต GUVA S12SD

วันที่	ช่วงเวลา(นาฬิกา)	การแสดงผลของเซนเซอร์		ค่าที่แสดงของเซนเซอร์(ดัชนี)
		แสดง	ไม่แสดง	
1/5/2564	0.00 น	✓		0.00
	4.00 น	✓		0.00
	8.00 น	✓		3.25
	12.00 น	✓		7.45
	16.00 น	✓		4.50
	20.00 น	✓		0.00
ค่าเฉลี่ย(รวม)				2.53
2/5/2564	0.00 น	✓		0.00
	4.00 น	✓		0.00
	8.00 น	✓		3.50
	12.00 น	✓		7.70
	16.00 น	✓		4.00
	20.00 น	✓		0.00
ค่าเฉลี่ย(รวม)				2.53
3/5/2564	0.00 น	✓		0.00
	4.00 น	✓		0.00
	8.00 น	✓		2.80
	12.00 น	✓		7.50

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำไปเผยแพร่โดยไม่ได้รับอนุญาตให้เผยแพร่เป็นการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วันที่	ช่วงเวลา(นาฬิกา)	การแสดงผลของเซนเซอร์		ค่าที่แสดงผลของเซนเซอร์(ดัชนี)
		แสดง	ไม่แสดง	
3/5/2564	16.00 น	✓		4.30
	20.00 น	✓		0.00
ค่าเฉลี่ย(รวม)				2.43
4/5/2564	0.00 น	✓		0.00
	4.00 น	✓		0.00
	8.00 น	✓		2.68
	12.00 น	✓		6.90
	16.00 น	✓		3.54
	20.00 น	✓		0.00
ค่าเฉลี่ย(รวม)				2.18
5/5/2564	0.00 น	✓		0.00
	4.00 น	✓		0.00
	8.00 น	✓		2.48
	12.00 น	✓		7.76
	16.00 น	✓		3.62
	20.00 น	✓		0.00
ค่าเฉลี่ย(รวม)				2.31

#### 4.1.2 การทดสอบนำค่าเซนเซอร์ที่วัดค่าความเข้มของรังสีอัลตราไวโอเล็ตเข้า

##### ฐานข้อมูล

##### วัตถุประสงค์ในการทดลอง

1. เพื่อทดสอบการแสดงผลค่าของเซนเซอร์ว่าแสดงผลค่าได้ตลอดเวลาได้จริงไม่
2. เพื่อทดสอบการแสดงผลค่าของเซนเซอร์ว่าสามารถแสดงผลค่าได้จริงไม่
3. เพื่อทดสอบการส่งค่าของเซนเซอร์ว่ามีการส่งเข้าฐานข้อมูลได้จริงไม่

##### ขั้นตอนการทดลอง

1. ต่อขา VCC เข้ากับแหล่งจ่ายไฟฟ้ากระแสตรงด้วยแรงดันไฟฟ้า 5 V ดังรูปที่ 4.1
2. ต่อขาสัญญาณเข้ากับขาอีเอสพี 8266 ที่ตั้งค่าไว้เพื่ออ่านค่าของเซนเซอร์ ดังรูปที่ 4.1
3. นำเซนเซอร์ไปไว้บริเวณที่โล่งแจ้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. สังเกตการแสดงค่าของเซนเซอร์ในการส่งข้อมูลไปยังฐานข้อมูล ดังรูปที่ 4.5
5. โดยใช้ระยะเวลาที่ใช้ในการทดสอบตั้งแต่วันที่ 1 พ.ค 2564 – 5 พ.ค 2564

```

56 sensorValue = analogRead(A5);
57 sensorVoltage = sensorValue/4095*3.3;
58 int sensorValuep = analogRead(A7);
59 float outputTemp = sensorValuep/1.3 /4095.0;
60 int level = #ofAnalogInputs;
61
62 HTTPClient http;
63 // Your device runs with URL path or IP address with path
64 http.begin(serverName);
65 http.addHeader("Content-Type", "application/x-www-form-urlencoded");
66
67 // Prepare your HTTP POST request data
68 String httpRequestData = "api_key="+ apiKeyValue + "&pin="+sensorVoltagep;
69 Serial.println(httpRequestData);
70 http.POST(httpRequestData);
71 int httpResponseCode = http.POST(httpRequestData);
72 // If you need an HTTP request with a content type: text/plain
73 //http.addHeader("Content-Type", "text/plain");
74 //int httpResponseCode = http.POST("hello, world!");
75
76 // If you need an HTTP request with a content type: application/json, use the following:
77 //http.addHeader("Content-Type", "application/json");
78 //int httpResponseCode = http.POST("{\"value1\":\"10\",\"value2\":\"57\",\"value3\":\"78\"}");
79
80 if (httpResponseCode) {
81   Serial.println("HTTP Response code: ");
82   Serial.println(httpResponseCode);
83 }
84 else {
85   Serial.println("Error code: ");
86 }
    
```

รูปที่ 4.3 โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์เพื่อวัดค่าความเข้มของรังสีอัลตราไวโอเล็ต GUYA S12SD บนฐานข้อมูล

```

HTTP Requests
-----
POST /post-esp-data.php      200 OK
POST /post-esp-data1.php    200 OK
POST /upload1.php           200 OK
POST /post-esp-data.php      200 OK
POST /post-esp-data.php      200 OK
POST /post-esp-data1.php    200 OK
POST /upload1.php           200 OK
POST /post-esp-data.php      200 OK
POST /post-esp-data.php      200 OK
POST /post-esp-data1.php    200 OK
    
```

รูปที่ 4.4 แสดงการส่งข้อมูลเซนเซอร์วัดค่าความเข้มของรังสีอัลตราไวโอเล็ต GUYA S12SD ไปยังฐานข้อมูล

ตารางที่ 4.2 ตารางแสดงผลการทดสอบนำค่าเซนเซอร์ที่ได้มาวัดค่าความเข้มของรังสีอัลตราไวโอเล็ต GUYA S12SD บนฐานข้อมูล

วันที่	ช่วงเวลา(นาฬิกา)	การแสดงผลของเซนเซอร์		ค่าที่แสดงของเซนเซอร์(ดัชนี)
		แสดง	ไม่แสดง	
1/5/2564	0.00 น	✓		0.00
	4.00 น	✓		0.00
	8.00 น	✓		3.25
	12.00 น	✓		7.45
	16.00 น	✓		4.50
	20.00 น	✓		0.00
ค่าเฉลี่ย(รวม)				2.53

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วันที่	ช่วงเวลา(นาฬิกา)	การแสดงผลของเซนเซอร์		ค่าที่แสดงของเซนเซอร์(ดัชนี)
		แสดง	ไม่แสดง	
2/5/2564	0.00 น	✓		0.00
	4.00 น	✓		0.00
	8.00 น	✓		3.50
	12.00 น	✓		7.70
	16.00 น	✓		4.00
	20.00 น	✓		0.00
ค่าเฉลี่ย(รวม)				2.53
3/5/2564	0.00 น	✓		0.00
	4.00 น	✓		0.00
	8.00 น	✓		2.80
	12.00 น	✓		7.50
	16.00 น	✓		4.30
	20.00 น	✓		0.00
ค่าเฉลี่ย(รวม)				2.43
4/5/2564	0.00 น	✓		0.00
	4.00 น	✓		0.00
	8.00 น	✓		2.68
	12.00 น	✓		6.90
	16.00 น	✓		3.54
	20.00 น	✓		0.00
ค่าเฉลี่ย(รวม)				2.18
5/5/2564	0.00 น	✓		0.00
	4.00 น	✓		0.00
	8.00 น	✓		2.48
	12.00 น	✓		7.76
	16.00 น	✓		3.62
	20.00 น	✓		0.00
ค่าเฉลี่ย(รวม)				2.31

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.2 การทดสอบเซนเซอร์วัดความเข้มของแสง BH1750

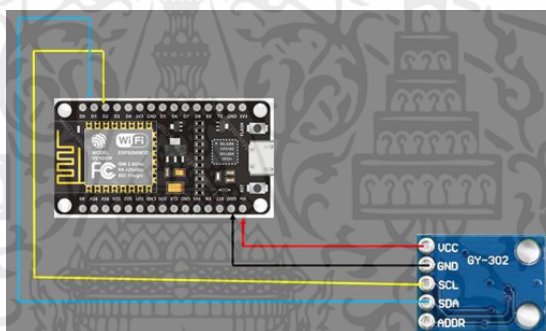
### 4.2.1 การทดสอบการทำงานของเซนเซอร์

#### วัตถุประสงค์ในการทดลอง

1. เพื่อทดสอบการแสดงผลค่าของเซนเซอร์ว่าแสดงค่าได้ตลอดเวลาหรือไม่
2. เพื่อทดสอบการแสดงผลค่าของเซนเซอร์ว่าสามารถแสดงค่าได้จริงไม่

#### ขั้นตอนการทดลอง

1. ต่อขา VCC เข้ากับแหล่งจ่ายไฟฟ้ากระแสตรงด้วยแรงดันไฟฟ้า 5 V ดังรูปที่ 4.6
2. ต่อขาสัญญาณเข้ากับพีอิน 8266 ที่ตั้งค่าไว้ให้อ่านค่าของเซนเซอร์ ดังรูปที่ 4.6
3. จากนั้นนำเซนเซอร์ไปไว้บริเวณที่โล่งแจ้ง
4. สังเกตการแสดงผลค่าของเซนเซอร์
5. โดยใช้ระยะเวลาที่ใช้ในการทดสอบตั้งแต่วันที่ 1 พ.ค 2564 – 5 พ.ค 2564



รูปที่ 4.5 การต่อใช้งานไอเอสพี 8266 กับเซนเซอร์วัดค่าความเข้มของแสง BH1750

```

sketch_jul14a | Arduino IDE 2.0.0-beta.9
File Edit Sketch Tools Help
Arduino Uno
sketch_jul14a.ino
1 #include <BH1750FVI.h>
2 BH1750FVI LightSensor(BH1750FVI::k_DeVNodeContLowRes);
3 void setup()
4 {
5   Serial.begin(115200);
6   LightSensor.begin();
7 }
8 void loop()
9 {
10  uint16_t lux = LightSensor.getLightIntensity();
11  Serial.print("Light: ");
12  Serial.println(lux);
13  Serial.println(" lux");
14  delay(250);
15 }
Output
Compilation error: Error: 2 UNKNOWN: exit status 1
Error: Request compile failed with message: 2 UNKNOWN: exit status 1
  
```

รูปที่ 4.6 โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์เพื่อวัดค่าความเข้มของแสง BH1750

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 ตารางแสดงทดสอบเซนเซอร์วัดค่าความเข้มของแสง BH1750

วันที่	ช่วงเวลา(นาฬิกา)	การแสดงผลของเซนเซอร์		ค่าที่แสดงของเซนเซอร์(klux)
		แสดง	ไม่แสดง	
1/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		3
	12.00 น	✓		58
	16.00 น	✓		40
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				16.83
2/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		3
	12.00 น	✓		56
	16.00 น	✓		14
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				12.16
3/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		2
	12.00 น	✓		54
	16.00 น	✓		14
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				11.66
4/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		2
	12.00 น	✓		55
	16.00 น	✓		10
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				11.16

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วันที่	ช่วงเวลา(นาฬิกา)	การแสดงผลของเซนเซอร์		ค่าที่แสดงผลของเซนเซอร์(klux)
		แสดง	ไม่แสดง	
5/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		1
	12.00 น	✓		52
	16.00 น	✓		12
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				10.83

#### 4.2.2 การทดสอบการนำค่าเซนเซอร์ที่วัดค่าความเข้มแสง BH1750 เข้า

##### ฐานข้อมูล

##### วัตถุประสงค์ในการทดลอง

1. เพื่อทดสอบการแสดงผลค่าของเซนเซอร์ว่าแสดงผลค่าได้ตลอดเวลาได้จริงไม่
2. เพื่อทดสอบการแสดงผลค่าของเซนเซอร์ว่าสามารถแสดงผลค่าได้จริงไม่
3. เพื่อทดสอบการส่งค่าของเซนเซอร์ว่ามีการส่งเข้าฐานข้อมูลได้จริงไม่

##### ขั้นตอนการทดลอง

1. ต่อขา VCC เข้ากับแหล่งจ่ายไฟฟ้ากระแสตรงด้วยแรงดันไฟฟ้า 5 V ดังรูปที่ 4.6
2. ต่อขาสัญญาณเข้ากับขาไอเอสพี 8266 ที่ตั้งค่าไว้เพื่ออ่านค่าของเซนเซอร์ ดังรูปที่ 4.6
3. นำเซนเซอร์ไปไว้บริเวณที่โล่งแจ้ง
4. สังเกตการแสดงผลค่าของเซนเซอร์ในการส่งข้อมูลไปยังฐานข้อมูล ดังรูปที่ 4.8
5. โดยใช้ระยะเวลาที่ใช้ในการทดสอบตั้งแต่วันที่ 1 พ.ค 2564 – 5 พ.ค 2564

```

1 // Include the library that provides a simple interface to the sensor.
2 #include <SPI.h>
3 #include <BH1750.h>
4
5 // Define pins for VCC, GND, and the sensor's SDA and SCL pins.
6 #define VCC_PIN 5
7 #define GND_PIN 0
8 #define SDA_PIN 4
9 #define SCL_PIN 5
10
11 // Define the sensor's I2C address.
12 #define I2C_ADDR 0x23
13
14 void setup() {
15   // Initialize the sensor.
16   BH1750 sensor(I2C_ADDR);
17   sensor.begin();
18   Serial.begin(9600);
19   Serial.println(" Lux:");
20 }
21
22 void loop() {
23   // Read the sensor's data.
24   float lux = sensor.readLightIntensity();
25   Serial.println(lux);
26 }

```

รูปที่ 4.7 โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์ที่วัดค่าความเข้มของแสง BH1750 บนฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

POST /post-esp-data.php 200 OK
POST /post-esp-data1.php 200 OK
POST /upload1.php 200 OK
POST /post-esp-data.php 200 OK
POST /post-esp-data1.php 200 OK
POST /upload1.php 200 OK
POST /post-esp-data.php 200 OK
POST /post-esp-data1.php 200 OK
POST /upload1.php 200 OK

```

รูปที่ 4.8 แสดงการส่งข้อมูลวัดค่าความเข้มแสง BH1750 ไปยังฐานข้อมูล

ตารางที่ 4.4 ตารางแสดงผลการทดสอบการนำค่าเซนเซอร์ที่วัดค่าความเข้มของแสง BH1750 บนฐานข้อมูล

วันที่	ช่วงเวลา(นาฬิกา)	การแสดงผลของเซนเซอร์		ค่าที่แสดงของเซนเซอร์(klux)
		แสดง	ไม่แสดง	
1/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		3
	12.00 น	✓		58
	16.00 น	✓		40
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				16.83
2/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		3
	12.00 น	✓		56
	16.00 น	✓		14
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				12.16
3/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		2
	12.00 น	✓		54
	16.00 น	✓		14
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				11.66

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วันที่	ช่วงเวลา(นาฬิกา)	การแสดงผลของเซนเซอร์		ค่าที่แสดงของเซนเซอร์(klux)
		แสดง	ไม่แสดง	
4/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		2
	12.00 น	✓		55
	16.00 น	✓		10
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				11.16
5/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		2
	12.00 น	✓		55
	16.00 น	✓		10
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				10.83

### 4.3 การทดสอบเซนเซอร์วัดความเร็วลม

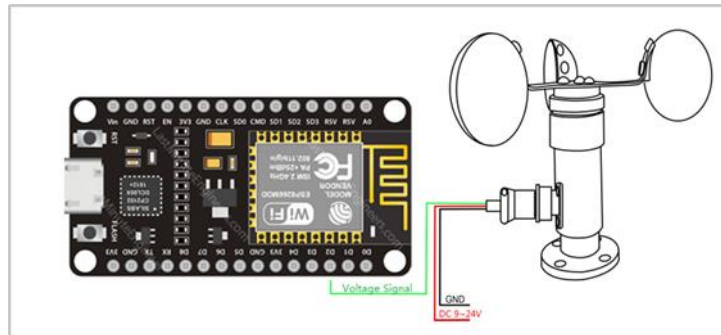
#### 4.3.1 การทดสอบการทำงานของเซนเซอร์วัดความเร็วลม

##### วัตถุประสงค์ในการทดลอง

1. เพื่อทดสอบการแสดงผลค่าของเซนเซอร์ว่าแสดงผลค่าได้ตลอดเวลาหรือไม่
2. เพื่อทดสอบการแสดงผลค่าของเซนเซอร์ว่าสามารถแสดงผลค่าได้จริงหรือไม่

##### ขั้นตอนการทดลอง

1. ต่อขา VCC เข้ากับแหล่งจ่ายไฟฟ้ากระแสตรงด้วยแรงดันไฟฟ้า 12-24 V ดังรูปที่ 4.9
2. ต่อขาสัญญาณเข้ากับออสซิลิโอสโคป 8266 ที่ตั้งค่าไว้ให้อ่านค่าของเซนเซอร์ ดังรูปที่ 4.9
3. จากนั้นนำเซนเซอร์ไปไว้บริเวณที่โล่งแจ้ง
4. สังเกตการแสดงผลค่าของเซนเซอร์
5. โดยใช้ระยะเวลาวันที่ใช้ในการทดสอบตั้งแต่วันที่ 1 พ.ค 2564 – 5 พ.ค 2564



รูปที่ 4.9 การต่อใช้งานไอเอสพี 8266 เข้ากับเซนเซอร์วัดความเร็วลม

```

File Edit Sketch Tools Help
Arduino Uno
sketch_jul14a.ino
3 Serial.begin(9600);
4 }
5
6 void loop()
7 {
8   int sensorValue = analogRead(A0);
9   float outvoltage = sensorValue * (5.0 / 1023.0);
10  Serial.print("outvoltage = ");
11  Serial.print(outvoltage);
12  Serial.println("V");
13  int Level = 6*outvoltage;//The level of wind speed is proportional
14  Serial.print("wind speed is ");
15  Serial.print(Level);
16  Serial.println(" level now");
17  Serial.println();
18  delay(500);
19 }
20

```

รูปที่ 4.10 โค้ดใช้สำหรับการทดสอบเซนเซอร์วัดความเร็วลม

ตารางที่ 4.5 ตารางแสดงผลการทดสอบเซนเซอร์วัดความเร็วลม

วันที่	ช่วงเวลา(นาฬิกา)	การแสดงผลของเซนเซอร์		ค่าที่แสดงของเซนเซอร์(m/s)
		แสดง	ไม่แสดง	
1/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		0
	12.00 น	✓		1
	16.00 น	✓		1
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				3

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

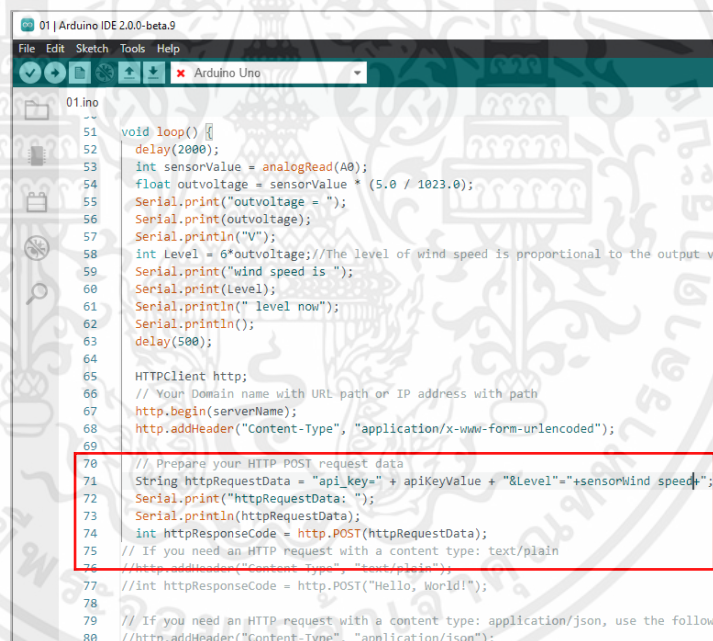
วันที่	ช่วงเวลา(นาฬิกา)	การแสดงผลของเซนเซอร์		ค่าที่แสดงของเซนเซอร์(m/s)
		แสดง	ไม่แสดง	
2/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		1
	12.00 น	✓		0
	16.00 น	✓		0
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				0.16
3/5/2564	0.00 น	✓		1
	4.00 น	✓		0
	8.00 น	✓		0
	12.00 น	✓		0
	16.00 น	✓		2
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				0.5
4/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		0
	12.00 น	✓		1
	16.00 น	✓		0
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				0.16
5/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		0
	12.00 น	✓		0
	16.00 น	✓		0
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				0

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.3.2 การทดสอบการนำค่าเซนเซอร์วัดความเร็วลมเข้าฐานข้อมูล วัตถุประสงค์ในการทดลอง

1. เพื่อทดสอบการแสดงค่าของเซนเซอร์ว่าแสดงค่าได้ตลอดเวลาได้จริงไม่
  2. เพื่อทดสอบการแสดงค่าของเซนเซอร์ว่าสามารถแสดงค่าได้จริงไม่
  3. เพื่อทดสอบการส่งค่าของเซนเซอร์ว่ามีการส่งเข้าฐานข้อมูลได้จริงไม่
- ขั้นตอนการทดลอง

1. ต่อขา VCC เข้ากับแหล่งจ่ายไฟฟ้ากระแสตรงด้วยแรงดันไฟฟ้า 12-24 V ดังรูปที่ 4.9
2. ต่อขาสัญญาณเข้ากับขาไอเอสพี 8266 ที่ตั้งค่าไว้เพื่ออ่านค่าของเซนเซอร์ ดังรูปที่ 4.9
3. นำเซนเซอร์ไปไว้บริเวณที่โล่งแจ้ง
4. สังเกตการแสดงค่าของเซนเซอร์ในการส่งข้อมูลไปยังฐานข้อมูล ดังรูปที่ 4.8
5. โดยใช้ระยะเวลาที่ใช้ในการทดสอบตั้งแต่วันที่ 1 พ.ค 2564 – 5 พ.ค 2564

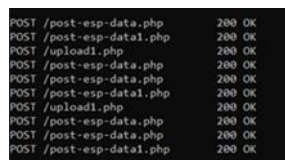


```

01.ino
--
51
52 delay(2000);
53 int sensorValue = analogRead(A0);
54 float outvoltage = sensorValue * (5.0 / 1023.0);
55 Serial.print("outvoltage = ");
56 Serial.print(outvoltage);
57 Serial.println("V");
58 int Level = 6*outvoltage;//The level of wind speed is proportional to the output voltage
59 Serial.print("wind speed is ");
60 Serial.print(Level);
61 Serial.println(" Level now");
62 Serial.println();
63 delay(500);
64
65 HTTPClient http;
66 // Your Domain name with URL path or IP address with path
67 http.begin(serverName);
68 http.addHeader("Content-Type", "application/x-www-form-urlencoded");
69
70 // Prepare your HTTP POST request data
71 String httpRequestData = "api_key="+ apiKeyValue + "&Level="+sensorWind speed";
72 Serial.print("httpRequestData: ");
73 Serial.println(httpRequestData);
74 int httpResponseCode = http.POST(httpRequestData);
75 // If you need an HTTP request with a content type: text/plain
76 //http.addHeader("Content-Type", "text/plain");
77 //int httpResponseCode = http.POST("Hello, World!");
78
79 // If you need an HTTP request with a content type: application/json, use the following
80 //http.addHeader("Content-Type", "application/json");

```

รูปที่ 4.11 โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์วัดความเร็วลมบนฐานข้อมูล



```

POST /post-esp-data.php 200 OK
POST /post-esp-data1.php 200 OK
POST /upload1.php 200 OK
POST /post-esp-data.php 200 OK
POST /post-esp-data.php 200 OK
POST /post-esp-data1.php 200 OK
POST /upload1.php 200 OK
POST /post-esp-data.php 200 OK
POST /post-esp-data.php 200 OK
POST /post-esp-data1.php 200 OK

```

รูปที่ 4.12 แสดงการส่งข้อมูลวัดความเร็วลมไปยังฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.6 ตารางแสดงผลการทดสอบการนำค่าเซนเซอร์วัดความเร็วลมบนฐานข้อมูล

วันที่	ช่วงเวลา(นาฬิกา)	การแสดงผลของเซนเซอร์		ค่าที่แสดงของเซนเซอร์(m/s)
		แสดง	ไม่แสดง	
1/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		0
	12.00 น	✓		1
	16.00 น	✓		1
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				3
2/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		1
	12.00 น	✓		0
	16.00 น	✓		0
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				0.16
3/5/2564	0.00 น	✓		1
	4.00 น	✓		0
	8.00 น	✓		0
	12.00 น	✓		0
	16.00 น	✓		2
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				0.5
4/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		0
	12.00 น	✓		1
	16.00 น	✓		0
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				0.16

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วันที่	ช่วงเวลา(นาฬิกา)	การแสดงผลของเซนเซอร์		ค่าที่แสดงของเซนเซอร์(m/s)
		แสดง	ไม่แสดง	
5/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		0
	12.00 น	✓		0
	16.00 น	✓		0
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				0

#### 4.4 การทดสอบเซนเซอร์วัดปริมาณน้ำฝน

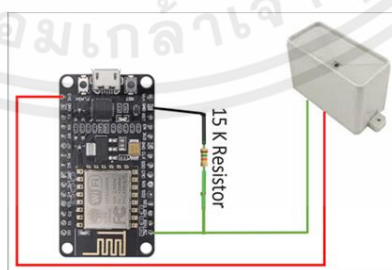
##### 4.4.1 การทดสอบการทำงานของเซนเซอร์วัดปริมาณน้ำฝน

###### วัตถุประสงค์ในการทดลอง

1. เพื่อทดสอบการแสดงผลค่าของเซนเซอร์ว่าแสดงค่าได้ตลอดเวลาได้จริงหรือไม่
2. เพื่อทดสอบการแสดงผลค่าของเซนเซอร์ว่าสามารถแสดงค่าได้จริงไม่

###### ขั้นตอนการทดลอง

1. ต่อขา VCC เข้ากับแหล่งจ่ายไฟที่กระแสตรงด้วยแรงดันไฟฟ้า 5 V ดังรูปที่ 4.13
2. ต่อขาสัญญาณเข้ากับขาไอเอสพี 8266 ที่ตั้งค่าไว้เพื่ออ่านค่าของเซนเซอร์ ดังรูปที่ 4.13
3. นำเซนเซอร์ไปไว้บริเวณที่โล่งแจ้ง
4. สังเกตการแสดงผลค่าของเซนเซอร์
5. โดยใช้ระยะเวลาที่ใช้ในการทดสอบตั้งแต่วันที่ 1 พ.ค 2564 – 5 พ.ค 2564



รูปที่ 4.13 การต่อใช้งานไอเอสพี 8266 กับเซนเซอร์วัดปริมาณน้ำฝน

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

15 bucketPosition=0;
16 dailyRain=0;
17 }
18 }
19 void setup(void) {
20   Serial.begin(9600); // start the serial port
21   rtc.begin(DateTime(__DATE__, __TIME__)); // start the RTC
22   pinMode(PIN_A, INPUT); // set the rain pin as
23   attachInterrupt(digitalPinToInterrupt(PIN_A), readRain, FALLING);
24   digitalWrite(PIN_A, HIGH); // it's time to starting the
25   Serial.println("Ready!!!"); // not necessary line
26 }
27 }
28
29 void loop(void) {
30   digitalWrite(PIN_A, HIGH);
31 }
32 // ===== Count the bucket flip =====
33 if (bucketPosition==HIGH) digitalWrite(PIN_A, LOW);
34 bucketPosition=LOW;
35 }
36 // ===== after the first =====
37 if (now.second() != 0) first = true; // after the first
38 if (now.second() == 0 && first == true) {
39   hourRain = dailyRain - dailyRain_kill_LastHour; // calculate the
40   dailyRain_kill_LastHour = dailyRain; // update the val
41 }
42 // Every 5 days the bucket is completed
43 Serial.println(now.Hour());
44 Serial.println(" ");
45 Serial.println(now.Minute());
46 Serial.println("Total Rain for the day = ");
47 Serial.println(dailyRain);
48 Serial.println(" inches"); // in 5' inches
49 Serial.println(" ");
50 Serial.println("Rain in Last Hour = ");
51 Serial.println(hourRain);
52 Serial.println(" inches");
53 Serial.println(" ");
54 }

```

รูปที่ 4.14 โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์วัดปริมาณน้ำฝน

ตารางที่ 4.7 ตารางแสดงผลทดสอบของเซนเซอร์วัดปริมาณน้ำฝน

วันที่	ช่วงเวลา(นาฬิกา)	การแสดงผลของเซนเซอร์		ค่าที่แสดงของเซนเซอร์(cm)
		แสดง	ไม่แสดง	
1/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		0
	12.00 น	✓		0
	16.00 น	✓		0
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				0
2/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		0
	12.00 น	✓		0
	16.00 น	✓		0
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				0
3/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		0
	12.00 น	✓		0

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วันที่	ช่วงเวลา(นาฬิกา)	การแสดงผลของเซนเซอร์		ค่าที่แสดงผลของเซนเซอร์(cm)
		แสดง	ไม่แสดง	
3/5/2564	16.00 น	✓		0
	20.00 น	✓		0
4/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		0
	12.00 น	✓		0
	16.00 น	✓		0
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				0
5/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		0
	12.00 น	✓		0
	16.00 น	✓		0
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				0

#### 4.4.2 การทดสอบการนำค่าเซนเซอร์วัดปริมาณน้ำฝนเข้าฐานข้อมูล วัตถุประสงค์ในการทดลอง

1. เพื่อทดสอบการแสดงผลค่าของเซนเซอร์ว่าแสดงผลค่าได้ตลอดเวลาได้จริงหรือไม่
2. เพื่อทดสอบการแสดงผลค่าของเซนเซอร์ว่าสามารถแสดงผลค่าได้จริงหรือไม่
3. เพื่อทดสอบการส่งค่าของเซนเซอร์ว่ามีการส่งเข้าฐานข้อมูลได้จริงหรือไม่

#### ขั้นตอนการทดลอง

1. ต่อขา VCC เข้ากับแหล่งจ่ายไฟฟ้ากระแสตรงด้วยแรงดันไฟฟ้า 5 V ดังรูปที่ 4.13
2. ต่อขาสัญญาณเข้ากับขาไอเอสพี 8266 ที่ตั้งค่าไว้เพื่ออ่านค่าของเซนเซอร์ ดังรูปที่ 4.13
3. นำเซนเซอร์ไปไว้บริเวณที่โล่งแจ้ง
4. สังเกตการแสดงผลค่าของเซนเซอร์ในการส่งข้อมูลไปยังฐานข้อมูล ดังรูปที่ 4.8
5. โดยใช้ระยะเวลาที่ใช้ในการทดสอบตั้งแต่วันที่ 1 พ.ค 2564 – 5 พ.ค 2564

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

84
85
86 // Sets $url for users to connect
87 $url = "http://192.168.1.100:8080";
88 $url = "http://192.168.1.100:8080";
89 $url = "http://192.168.1.100:8080";
90 $url = "http://192.168.1.100:8080";
91 $url = "http://192.168.1.100:8080";
92 $url = "http://192.168.1.100:8080";
93 $url = "http://192.168.1.100:8080";
94 $url = "http://192.168.1.100:8080";
95 $url = "http://192.168.1.100:8080";
96 $url = "http://192.168.1.100:8080";
97 $url = "http://192.168.1.100:8080";
98 $url = "http://192.168.1.100:8080";
99 $url = "http://192.168.1.100:8080";
100 $url = "http://192.168.1.100:8080";
101 $url = "http://192.168.1.100:8080";
102 $url = "http://192.168.1.100:8080";
103 $url = "http://192.168.1.100:8080";
104 $url = "http://192.168.1.100:8080";
105 $url = "http://192.168.1.100:8080";
106 $url = "http://192.168.1.100:8080";
107 $url = "http://192.168.1.100:8080";
108 $url = "http://192.168.1.100:8080";
109 $url = "http://192.168.1.100:8080";
110 $url = "http://192.168.1.100:8080";
111 $url = "http://192.168.1.100:8080";

```

รูปที่ 4.15 โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์วัดปริมาณน้ำฝนบนฐานข้อมูล

```

POST /post-esp-data.php 200 OK
POST /post-esp-data1.php 200 OK
POST /upload1.php 200 OK
POST /post-esp-data.php 200 OK
POST /post-esp-data1.php 200 OK
POST /upload1.php 200 OK
POST /post-esp-data.php 200 OK
POST /post-esp-data1.php 200 OK
POST /upload1.php 200 OK

```

รูปที่ 4.16 แสดงการส่งข้อมูลวัดปริมาณน้ำฝนไปยังฐานข้อมูล

ตารางที่ 4.8 ตารางแสดงผลของทดสอบนำค่าเซนเซอร์วัดปริมาณน้ำฝนบนฐานข้อมูล

วันที่	ช่วงเวลา(นาฬิกา)	การแสดงผลของเซนเซอร์		ค่าที่แสดงของเซนเซอร์(cm)
		แสดง	ไม่แสดง	
1/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		0
	12.00 น	✓		0
	16.00 น	✓		0
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				0
2/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		0

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	12.00 น	✓		0
วันที่	ช่วงเวลา(นาฬิกา)	การแสดงผลของเซนเซอร์		ค่าที่แสดงของเซนเซอร์(cm)
		แสดง	ไม่แสดง	
2/5/2564	16.00 น	✓		0
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				0
3/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		0
	12.00 น	✓		0
	16.00 น	✓		0
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				0
4/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		0
	12.00 น	✓		0
	16.00 น	✓		0
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				0
5/5/2564	0.00 น	✓		0
	4.00 น	✓		0
	8.00 น	✓		0
	12.00 น	✓		0
	16.00 น	✓		0
	20.00 น	✓		0
ค่าเฉลี่ย(รวม)				0

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.5 การทดสอบเซนเซอร์วัดอุณหภูมิ

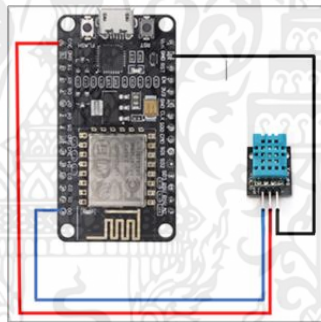
### 4.5.1 การทดสอบการทำงานของเซนเซอร์วัดอุณหภูมิ

#### วัตถุประสงค์ในการทดลอง

1. เพื่อทดสอบการแสดงผลค่าของเซนเซอร์ว่าแสดงค่าได้ตลอดเวลาได้จริงหรือไม่
2. เพื่อทดสอบการแสดงผลค่าของเซนเซอร์ว่าสามารถแสดงผลค่าได้จริงหรือไม่

#### ขั้นตอนการทดลอง

1. ต่อขา VCC เข้ากับแหล่งจ่ายไฟฟ้ากระแสตรงด้วยแรงดันไฟฟ้า 5 V ดังรูปที่ 4.21
2. ต่อขาสัญญาณเข้ากับขาไอเอสพี 8266 ที่ตั้งค่าไว้เพื่ออ่านค่าของเซนเซอร์ ดังรูปที่ 4.21
3. นำเซนเซอร์ไปไว้บริเวณที่โล่งแจ้ง
4. สังเกตการแสดงผลค่าของเซนเซอร์
5. โดยใช้ระยะเวลาที่ใช้ในการทดสอบตั้งแต่วันที่ 1 พ.ค 2564 – 5 พ.ค 2564



รูปที่ 4.17 การต่อใช้งานไอเอสพี 8266 กับเซนเซอร์วัดอุณหภูมิ

```

Arduino Uno
sketch_jul14a.ino
1 #include "DHT.h"
2
3 DHT dht;
4
5 void setup()
6 {
7   Serial.begin(9600);
8   Serial.println();
9   Serial.println("Status\tHumidity (%)\tTemperature (C)\t(F)");
10
11  dht.setup(2); // data pin 2
12 }
13
14 void loop()
15 {
16   delay(dht.getMinimumSamplingPeriod());
17
18   float humidity = dht.getHumidity(); // ดึงค่าความชื้น
19   float temperature = dht.getTemperature(); // ดึงค่าอุณหภูมิ
20
21   Serial.print(dht.getStatusString());
22   Serial.print("\t");
23   Serial.print(humidity, 1);
24   Serial.print("\t");
25   Serial.print(temperature, 1);
26   Serial.print("\t");
27   Serial.println(dht.toFahrenheit(temperature), 1);
28 }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## รูปที่ 4.18 โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์วัดอุณหภูมิ

ตารางที่ 4.9 ตารางแสดงผลการทดสอบของเซนเซอร์วัดอุณหภูมิ

วันที่	ช่วงเวลา(นาฬิกา)	การแสดงผลของเซนเซอร์		ค่าที่แสดงของเซนเซอร์(°C)
		แสดง	ไม่แสดง	
1/5/2564	0.00 น	✓		25.50
	4.00 น	✓		25.60
	8.00 น	✓		29.10
	12.00 น	✓		43.10
	16.00 น	✓		35.10
	20.00 น	✓		26.70
ค่าเฉลี่ย(รวม)				30.85
2/5/2564	0.00 น	✓		25.00
	4.00 น	✓		25.20
	8.00 น	✓		29.00
	12.00 น	✓		42.00
	16.00 น	✓		34.90
	20.00 น	✓		26.70
ค่าเฉลี่ย(รวม)				30.46
3/5/2564	0.00 น	✓		24.50
	4.00 น	✓		23.00
	8.00 น	✓		28.00
	12.00 น	✓		43.00
	16.00 น	✓		35.01
	20.00 น	✓		26.71
ค่าเฉลี่ย(รวม)				30.03
4/5/2564	0.00 น	✓		20.00
	4.00 น	✓		20.50
	8.00 น	✓		30.00
	12.00 น	✓		40.30
	16.00 น	✓		33.00

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



## รูปที่ 4.19 โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์วัดอุณหภูมิบนฐานข้อมูล

ตารางที่ 4.10 ตารางแสดงผลทดสอบของการนำค่าเซนเซอร์วัดอุณหภูมิบนฐานข้อมูล

วันที่	ช่วงเวลา(นาฬิกา)	การแสดงผลของเซนเซอร์		ค่าที่แสดงของเซนเซอร์(°C)
		แสดง	ไม่แสดง	
1/5/2564	0.00 น	✓		25.50
	4.00 น	✓		25.60
	8.00 น	✓		29.10
	12.00 น	✓		43.10
	16.00 น	✓		35.10
	20.00 น	✓		26.70
ค่าเฉลี่ย(รวม)				30.85
2/5/2564	0.00 น	✓		25.00
	4.00 น	✓		25.20
	8.00 น	✓		29.00
	12.00 น	✓		42.00
	16.00 น	✓		34.90
	20.00 น	✓		26.70
ค่าเฉลี่ย(รวม)				30.46
3/5/2564	0.00 น	✓		24.50
	4.00 น	✓		23.00
	8.00 น	✓		28.00
	12.00 น	✓		43.00
	16.00 น	✓		35.01
	20.00 น	✓		26.71
ค่าเฉลี่ย(รวม)				30.03
4/5/2564	0.00 น	✓		20.00
	4.00 น	✓		20.50
	8.00 น	✓		30.00
	12.00 น	✓		40.30
	16.00 น	✓		33.00

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	20.00 น	✓		28.00
วันที่	ช่วงเวลา(นาฬิกา)	การแสดงผลของเซนเซอร์		ค่าที่แสดงของเซนเซอร์(°C)
		แสดง	ไม่แสดง	
ค่าเฉลี่ย(รวม)				28.63
5/5/2564	0.00 น	✓		21.00
	4.00 น	✓		20.00
	8.00 น	✓		33.00
	12.00 น	✓		41.00
	16.00 น	✓		34.10
	20.00 น	✓		30.20
ค่าเฉลี่ย(รวม)				29.88

#### 4.6 การทดสอบแสดงผลค่าเซนเซอร์ใช้แบบร่วมกัน

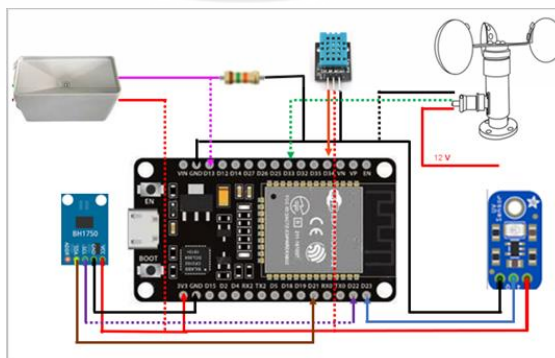
##### 4.6.1 การทดสอบการทำงานของเซนเซอร์ใช้แบบร่วมกัน

###### วัตถุประสงค์ในการทดลอง

1. เพื่อทดสอบการแสดงผลค่าของเซนเซอร์ใช้แบบร่วมกันว่าแสดงผลค่าได้ตลอดเวลาได้จริงหรือไม่
2. เพื่อทดสอบการแสดงผลค่าของเซนเซอร์ใช้แบบร่วมกันว่าสามารถแสดงผลค่าได้จริงหรือไม่

###### ขั้นตอนการทดลอง

1. ทำการต่อแหล่งจ่ายไฟดีซีให้กับตัวเซนเซอร์ทุกเซนเซอร์ดังรูปที่ 4.26
2. ต่อขาของสัญญาณเข้ากับพีอิน 8266 ที่ตั้งค่าไว้อ่านค่าของเซนเซอร์แต่ละตัวดังรูปที่ 4.26
3. นำเซนเซอร์ไปไว้บริเวณที่โล่งแจ้ง
4. สังเกตการแสดงผลค่าของเซนเซอร์
5. โดยใช้ระยะเวลาที่ใช้ในการทดสอบตั้งแต่วันที่ 1 พ.ค 2564 – 5 พ.ค 2564



เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## รูปที่ 4.20 การต่อใช้งานอีเอสพี 8266 กับเซนเซอร์รวมกันทุกตัว

```

sketch_jul14a.ino
48   uateime now = rtc.now();
49   float h = dht.readHumidity();
50   float t = dht.readTemperature();
51   float f = dht.readTemperature(true);
52   uint16_t lux = LightSensor.GetLightIntensity();
53   int sensorValue = analogRead(15);
54   int uvLevel = averageAnalogRead(UVOUT);
55   int refLevel = averageAnalogRead(REF_3V3);
56   float outputVolty = 3.3 *uvLevel/4095;
57   float uvIntensity = mapFloat(outputVoltageuv,0.99,2.8,0.0,15.0);
58   float outvoltage = sensorValue * (3.3 /4095.0);
59   int level = 6*outvoltage;
60   if ((bucketPositionA==true)&&(digitalRead(RainPin)==LOW)){
61     bucketPositionA=false;
62   }
63   if(now.second() != 0) first = true;
64   if(now.second() == 0 && first == true){
65     hourlyRain = dailyRain - dailyRain_till_LastHour; // calculate the last hour's rain
66     dailyRain_till_LastHour = dailyRain;
67     Serial.print(now.hour());
68     Serial.print(":");
69     Serial.print(now.minute());
70     Serial.print("Light: ");
71     Serial.print(lux);
72     Serial.println(" lux");
73     Serial.print("Humidity: ");
74     Serial.print(h);
75     Serial.print(" %");
76     Serial.print("Temperature: ");
77     Serial.print(t);
78     Serial.print(" f");
79     Serial.print(f);

```

## รูปที่ 4.21 โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์รวมกันทุกตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.11 ตารางแสดงผลการทดสอบของเซนเซอร์ใช้แบบร่วมกัน

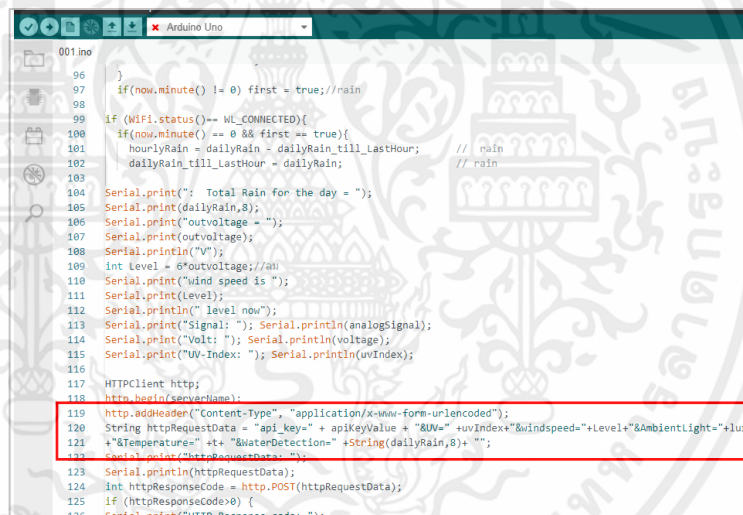
วันที่	ช่วงเวลา (นาฬิกา)	การแสดงผลของเซนเซอร์														
		ค่าความเข้มของรังสีอัลตราไวโอเล็ต			ค่าความเข้มของแสง			ค่าความเร็วลม			ปริมาณน้ำฝน			อุณหภูมิ		
		แสดง	ไม่แสดง	ค่าที่แสดง (ดัชนี)	แสดง	ไม่แสดง	ค่าที่แสดง (klux)	แสดง	ไม่แสดง	ค่าที่แสดง (m/s)	แสดง	ไม่แสดง	ค่าที่แสดง (cm)	แสดง	ไม่แสดง	ค่าที่แสดง (°C)
1/5/2564	0.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		25.50
	4.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		25.60
	8.00 น	✓		3.25	✓		3	✓		0	✓		0	✓		29.10
	12.00 น	✓		7.45	✓		58	✓		1	✓		0	✓		43.10
	16.00 น	✓		4.50	✓		40	✓		1	✓		0	✓		35.10
	20.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		26.70
ค่าเฉลี่ย(รวม)				2.53			16.83			3			0			30.85
2/5/2564	0.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		25.00
	4.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		25.20
	8.00 น	✓		3.50	✓		3	✓		1	✓		0	✓		29.00
	12.00 น	✓		7.70	✓		56	✓		0	✓		0	✓		42.00
	16.00 น	✓		4.00	✓		14	✓		0	✓		0	✓		34.90
	20.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		26.70
ค่าเฉลี่ย(รวม)				2.53			12.16			0.16			0			30.46
3/5/2564	0.00 น	✓		0.00	✓		0	✓		1	✓		0	✓		24.50
	4.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		23.00
	8.00 น	✓		2.80	✓		2	✓		0	✓		0	✓		28.00
	12.00 น	✓		7.50	✓		54	✓		0	✓		0	✓		43.00
	16.00 น	✓		4.30	✓		14	✓		2	✓		0	✓		35.01
	20.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		26.71
ค่าเฉลี่ย(รวม)				2.43			11.66			0.5			0			30.03
4/5/2564	0.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		20.00
	4.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		20.50
	8.00 น	✓		2.68	✓		2	✓		0	✓		0	✓		30.00

วันที่	ช่วงเวลา (นาฬิกา)	การแสดงผลของเซนเซอร์														
		ค่าความเข้มของรังสีอัลตราไวโอเล็ต			ค่าความเข้มของแสง			ค่าความเร็วลม			ปริมาณน้ำฝน			อุณหภูมิ		
		แสดง	ไม่แสดง	ค่าที่แสดง (ดัชนี)	แสดง	ไม่แสดง	ค่าที่แสดง()	แสดง	ไม่แสดง	ค่าที่แสดง (m/s)	แสดง	ไม่แสดง	ค่าที่แสดง (cm)	แสดง	ไม่แสดง	ค่าที่แสดง (°C)
4/5/2564	12.00 น	✓		6.90	✓		55	✓		1	✓		0	✓		40.30
	16.00 น	✓		3.54	✓		10	✓		0	✓		0	✓		33.00
	20.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		28.00
ค่าเฉลี่ย(รวม)				2.18			11.66			0.16			0			28.63
5/5/2564	0.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		21.00
	4.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		20.00
	8.00 น	✓		2.48	✓		1	✓		0	✓		0	✓		33.00
	12.00 น	✓		7.76	✓		52	✓		0	✓		0	✓		41.00
	16.00 น	✓		3.62	✓		12	✓		0	✓		0	✓		34.10
	20.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		30.20
ค่าเฉลี่ย(รวม)				2.31			10.83			0			0			29.88

#### 4.6.2 การทดสอบนำค่าเซนเซอร์ของเซนเซอร์ใช้แบบร่วมกันเข้าฐานข้อมูล

วัตถุประสงค์ในการทดลอง

1. เพื่อทดสอบการแสดงค่าของเซนเซอร์ใช้แบบร่วมกันว่าแสดงค่าได้ตลอดเวลาได้จริงหรือไม่
  2. เพื่อทดสอบการแสดงค่าของเซนเซอร์ใช้แบบร่วมกันว่าสามารถแสดงค่าได้จริงหรือไม่
  3. เพื่อทดสอบการส่งค่าของเซนเซอร์ใช้แบบร่วมกันว่ามีการส่งเข้าฐานข้อมูลได้จริงหรือไม่
- ขั้นตอนการทดลอง
1. ทำการต่อแหล่งจ่ายไฟดีซีให้กับตัวเซนเซอร์ทุกเซนเซอร์ดังรูปที่ 4.26
  2. ต่อขาของสัญญาณเข้ากับขาไอเอสพี 8266 ที่ตั้งค่าไว้อ่านค่าของเซนเซอร์แต่ละตัวดังรูปที่ 4.26
  3. นำเซนเซอร์ทุกเซนเซอร์ไปวางไว้บริเวณที่โล่งแจ้ง
  4. สั่งการให้อีเอสพีแปดสองหกหวนค่าเซนเซอร์ทุกเซนเซอร์ไปเก็บไว้บนฐานข้อมูล
  5. สังเกตการแสดงค่าของเซนเซอร์บนฐานข้อมูล

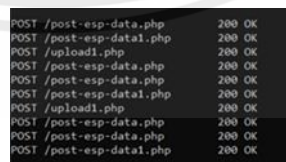


```

001.ino
96 }
97 if(now.minute() != 0) first = true;//rain
98
99 if (Wifi.status() == WL_CONNECTED){
100   if(now.minute() == 0 && first == true){
101     hourlyRain = dailyRain - dailyRain_till_LastHour; // rain
102     dailyRain_till_LastHour = dailyRain; // rain
103
104     Serial.print(": Total Rain for the day = ");
105     Serial.print(dailyRain,8);
106     Serial.print("outvoltage = ");
107     Serial.print(outvoltage);
108     Serial.println("V");
109     int level = 6*outvoltage//ml
110     Serial.print("wind speed is ");
111     Serial.print(level);
112     Serial.println(" level now");
113     Serial.print("Signal: "); Serial.println(analogSignal);
114     Serial.print("Volt: "); Serial.println(voltage);
115     Serial.print("UV-Index: "); Serial.println(uvIndex);
116
117     HTTPClient http;
118     http.begin(serverName);
119     http.addHeader("Content-Type", "application/x-www-form-urlencoded");
120     String httpRequestData = "api_key="+apiKeyValue + "&uvIndex="+uvIndex+"&windspeed="+level+"&AmbientLight="+lux
121     + "&Temperature=" + t + "&WaterDetection=" +String(dailyRain,8)+"";
122     Serial.print("httpRequestData: ");
123     Serial.println(httpRequestData);
124     int httpResponseCode = http.POST(httpRequestData);
125     if (httpResponseCode>0) {
126       Serial.print("HTTP Response code: ");

```

รูปที่ 4.22 โค้ดที่ใช้สำหรับการทดสอบเซนเซอร์รวมกันทุกตัวบนฐานข้อมูล



```

POST /post-esp-data.php 200 OK
POST /post-esp-data1.php 200 OK
POST /upload1.php 200 OK
POST /post-esp-data.php 200 OK
POST /post-esp-data1.php 200 OK
POST /upload1.php 200 OK
POST /post-esp-data.php 200 OK
POST /post-esp-data1.php 200 OK
POST /post-esp-data.php 200 OK

```

รูปที่ 4.23 แสดงการส่งข้อมูลเซนเซอร์รวมกันทุกตัวไปยังฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.12 ตารางแสดงผลการทดสอบการนำค่าเซนเซอร์ทุกเซนเซอร์บนฐานข้อมูล

วันที่	ช่วงเวลา (นาฬิกา)	การแสดงผลค่าเซนเซอร์ทุกเซนเซอร์บนฐานข้อมูล														
		ค่าความเข้มของรังสีอัลตราไวโอเล็ต			ค่าความเข้มของแสง			ค่าความเร็วลม			ปริมาณน้ำฝน			อุณหภูมิ		
		แสดง	ไม่แสดง	ค่าที่แสดง (ดัชนี)	แสดง	ไม่แสดง	ค่าที่แสดง (klux)	แสดง	ไม่แสดง	ค่าที่แสดง (m/s)	แสดง	ไม่แสดง	ค่าที่แสดง (cm)	แสดง	ไม่แสดง	ค่าที่แสดง(°C)
1/5/2564	0.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		25.50
	4.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		25.60
	8.00 น	✓		3.25	✓		3	✓		0	✓		0	✓		29.10
	12.00 น	✓		7.45	✓		58	✓		1	✓		0	✓		43.10
	16.00 น	✓		4.50	✓		40	✓		1	✓		0	✓		35.10
	20.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		26.70
ค่าเฉลี่ย(รวม)				2.53			16.83			3			0			30.85
2/5/2564	0.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		25.00
	4.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		25.20
	8.00 น	✓		3.50	✓		3	✓		1	✓		0	✓		29.00
	12.00 น	✓		7.70	✓		56	✓		0	✓		0	✓		42.00
	16.00 น	✓		4.00	✓		14	✓		0	✓		0	✓		34.90
	20.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		26.70
ค่าเฉลี่ย(รวม)				2.53			12.16			0.16			0			30.46
3/5/2564	0.00 น	✓		0.00	✓		0	✓		1	✓		0	✓		24.50
	4.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		23.00
	8.00 น	✓		2.80	✓		2	✓		0	✓		0	✓		28.00
	12.00 น	✓		7.50	✓		54	✓		0	✓		0	✓		43.00
	16.00 น	✓		4.30	✓		14	✓		2	✓		0	✓		35.01
	20.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		26.71
ค่าเฉลี่ย(รวม)				2.43			11.66			0.5			0			30.03
4/5/2564	0.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		20.00
	4.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		20.50
	8.00 น	✓		2.68	✓		2	✓		0	✓		0	✓		30.00

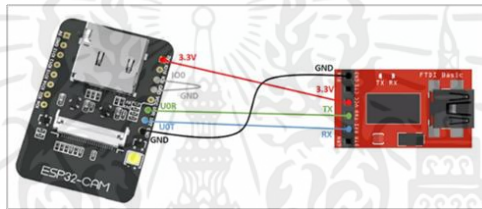
วันที่	ช่วงเวลา (นาฬิกา)	การแสดงผลเซนเซอร์ทุกเซนเซอร์บนฐานข้อมูล														
		ค่าความเข้มของรังสีอัลตราไวโอเล็ต			ค่าความเข้มของแสง			ค่าความเร็วลม			ปริมาณน้ำฝน			อุณหภูมิ		
		แสดง	ไม่แสดง	ค่าที่แสดง (ดัชนี)	แสดง	ไม่แสดง	ค่าที่แสดง (klux)	แสดง	ไม่แสดง	ค่าที่แสดง (m/s)	แสดง	ไม่แสดง	ค่าที่แสดง (cm)	แสดง	ไม่แสดง	ค่าที่แสดง (°C)
4/5/2564	12.00 น	✓		6.90	✓		55	✓		1	✓		0	✓		40.30
	16.00 น	✓		3.54	✓		10	✓		0	✓		0	✓		33.00
	20.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		28.00
ค่าเฉลี่ย(รวม)				2.18			11.66			0.16			0			28.63
5/5/2564	0.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		21.00
	4.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		20.00
	8.00 น	✓		2.48	✓		1	✓		0	✓		0	✓		33.00
	12.00 น	✓		7.76	✓		52	✓		0	✓		0	✓		41.00
	16.00 น	✓		3.62	✓		12	✓		0	✓		0	✓		34.10
	20.00 น	✓		0.00	✓		0	✓		0	✓		0	✓		30.20
ค่าเฉลี่ย(รวม)				2.31			10.83			0			0			29.88

## 4.7 การทดสอบกล่อง ESP-32 CAM

### 4.7.1 การทดสอบการทำงานของกล่อง ESP-32 CAM

#### วัตถุประสงค์ในการทดลอง

1. เพื่อทดสอบความเที่ยงตรงของกล่อง ESP-32 CAM
2. เพื่อทดสอบการส่งข้อมูลเป็นรูปภาพลงบนฐานข้อมูล  
ขั้นตอนการทดลอง
1. ทำการต่อแหล่งจ่ายไฟดีซีให้กับตัวกล่อง ESP-32 CAM
2. เชื่อมต่อกล่อง ESP-32 CAM กับฐานข้อมูล
3. นำกล่อง ESP-32 CAM ไปวางไว้บริเวณที่โล่งแจ้ง



รูปที่ 4.24 การต่ออุปกรณ์งานESP32-CAM เข้ากับ FT232RLUSB เพื่อทำการทดสอบ

```

101
102 void loop() {
103   unsigned long currentMillis = millis();
104   if (currentMillis - previousMillis > timerInterval) {
105     sendPhoto();
106     previousMillis = currentMillis;
107   }
108 }
109
110 String sendPhoto() {
111   String getall;
112   String getbody;
113
114   camera_fb_t * fb = NULL;
115   fb = esp_camera_fb_get();
116   if(!fb) {
117     Serial.println("Camera capture failed");
118     delay(1000);
119     ESP.restart();
120   }
121
122   Serial.println("Connecting to Server: " + serverName);
123
124   if (!client.connect(serverName.c_str(), serverPort)) {
125     Serial.println("Connection successful");
126     String head = "--RandomMerTutorials\r\nContent-Disposition: form-data; name='imagefile'; filename='esp32-cam.jpg'";
127     String tail = "\r\n--RandomMerTutorials-\r\n";
128
129     uint16_t imagelen = fb->len;
130     uint16_t extralen = head.length() + tail.length();
131     uint16_t totalen = imagelen + extralen;
132

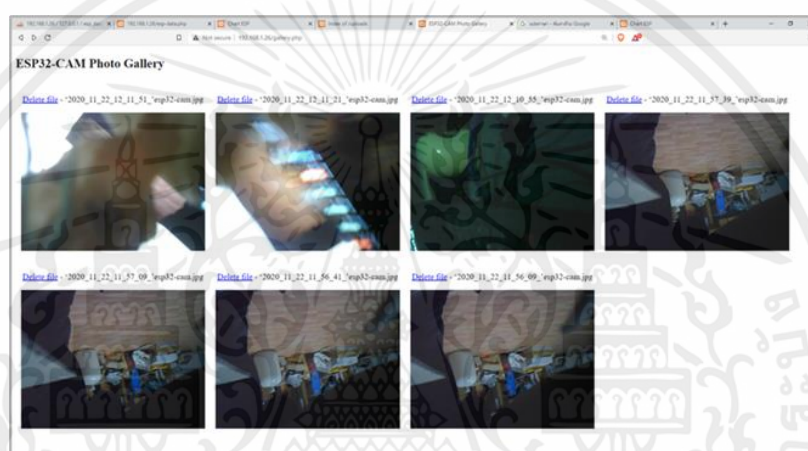
```

รูปที่ 4.25 โค้ดที่ใช้สำหรับการทดสอบ ESP32-CAM บนฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.13 ตารางแสดงผลการทดสอบการทำงานของกล้อง ESP-32 CAM

จำนวน (ครั้ง)	เวลา	การแสดงผลภาพที่กล้อง ESP-32 CAM ทำการถ่าย	
		แสดง	ไม่แสดง
1	0.00 น	✓	
2	4.00 น	✓	
3	8.00 น	✓	
4	12.00 น	✓	
5	16.00 น	✓	



รูปที่ 4.25 รูปภาพที่กล้อง ESP-32 CAM ทำการถ่ายได้นำมาเก็บไว้บนฐานข้อมูล

#### 4.8 การทดสอบเว็บแอปพลิเคชันจตุรายนสภาพอากาศ

##### 4.8.1 การทดสอบการทำงานของเว็บแอปพลิเคชันจตุรายนสภาพอากาศ

###### วัตถุประสงค์ในการทดลอง

1. เพื่อทดสอบความเที่ยงตรงของเว็บแอปพลิเคชันจตุรายนสภาพอากาศ
2. เพื่อทดสอบการแสดงผลข้อมูลของเว็บแอปพลิเคชันจตุรายนสภาพ

อากาศ

###### ขั้นตอนการทดลอง

1. ทดลองรีเฟรชเว็บแอปพลิเคชันจตุรายนสภาพอากาศ
2. เข้าไปดูแต่ละหน้าของเว็บแอปพลิเคชันจตุรายนสภาพอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.14 ตารางแสดงผลการทดสอบของเว็บแอปพลิเคชันจุดรายงานสภาพอากาศ

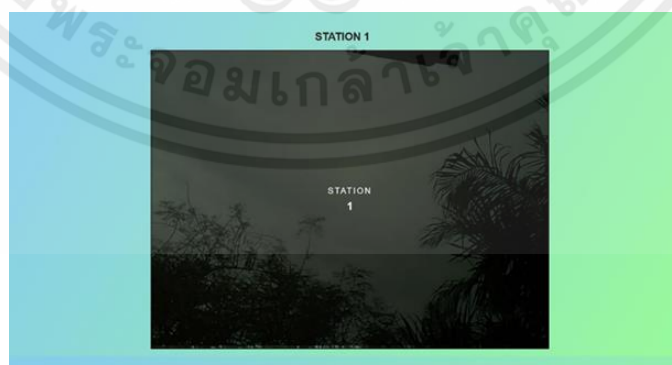
วันที่	เวลา	การแสดงผลข้อมูลหน้าเว็บแอปพลิเคชันจุดรายงานสภาพอากาศ		การแสดงผลรูปภาพที่กล้อง ESP-32 CAM ทำการถ่ายหน้าเว็บแอปพลิเคชันจุดรายงานสภาพอากาศ	
		แสดง	แสดง	แสดง	แสดง
1/5/2564	0.00 น	✓		✓	
	4.00 น	✓		✓	
	8.00 น	✓		✓	
	12.00 น	✓		✓	
	16.00 น	✓		✓	
2/5/2564	0.00 น	✓		✓	
	4.00 น	✓		✓	
	8.00 น	✓		✓	
	12.00 น	✓		✓	
	16.00 น	✓		✓	
3/5/2564	0.00 น	✓		✓	
	4.00 น	✓		✓	
	8.00 น	✓		✓	
	12.00 น	✓		✓	
	16.00 น	✓		✓	
4/5/2564	0.00 น	✓		✓	
	4.00 น	✓		✓	
	8.00 น	✓		✓	
	12.00 น	✓		✓	
	16.00 น	✓		✓	

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วันที่	เวลา	การแสดงผลข้อมูลหน้าเว็บ แอปพลิเคชันจตุรรายงาน สภาพอากาศ		การแสดงผลรูปภาพที่กล้อง ESP-32 CAM ทำการถ่าย หน้าเว็บแอปพลิเคชันจตุร รายงานสภาพอากาศ	
		แสดง	แสดง	แสดง	แสดง
5/5/2564	0.00 น	✓		✓	
	4.00 น	✓		✓	
	8.00 น	✓		✓	
	12.00 น	✓		✓	
	16.00 น	✓		✓	



รูปที่ 4.27 หน้าแสดงข้อมูลหน้าเว็บแอปพลิเคชันจตุรรายงานสภาพอากาศ



รูปที่ 4.28 หน้าแสดงรูปภาพที่กล้อง ESP-32 CAM ทำการถ่ายภาพหน้าเว็บแอปพลิเคชันตรงจุด  
รายงานสภาพอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.9 การทดสอบโมบายแอปพลิเคชันจตุรกายงานสภาพอากาศ

### 4.9.1 การทดสอบการทำงานของเว็บแอปพลิเคชันจตุรกายงานสภาพอากาศ

#### วัตถุประสงค์ในการทดลอง

1. เพื่อทดสอบความเที่ยงตรงของโมบายแอปพลิเคชันจตุรกายงานสภาพอากาศ
2. เพื่อทดสอบการแสดงผลข้อมูลของโมบายแอปพลิเคชันจตุรกายงานสภาพอากาศ

#### ขั้นตอนการทดลอง

1. ทดลองรีเฟรชโมบายแอปพลิเคชันจตุรกายงานสภาพอากาศ
2. เข้าไปดูแต่ละหน้าของโมบายแอปพลิเคชันจตุรกายงานสภาพอากาศ

ตารางที่ 4.15 ตารางแสดงผลการทดสอบของโมบายแอปพลิเคชันจตุรกายงานสภาพอากาศ

วันที่	เวลา	การแสดงผลข้อมูลหน้าโมบายแอปพลิเคชันจตุรกายงานสภาพอากาศ		การแสดงผลรูปภาพที่กล้อง ESP-32 CAM ทำการถ่ายภาพหน้าเว็บแอปพลิเคชันจตุรกายงานสภาพอากาศ	
		แสดง	ไม่แสดง	แสดง	ไม่แสดง
1/5/2564	0.00 น	✓		✓	
	4.00 น	✓		✓	
	8.00 น	✓		✓	
	12.00 น	✓		✓	
	16.00 น	✓		✓	
2/5/2564	0.00 น	✓		✓	
	4.00 น	✓		✓	
	8.00 น	✓		✓	
	12.00 น	✓		✓	
	16.00 น	✓		✓	
3/5/2564	0.00 น	✓		✓	
	4.00 น	✓		✓	
	8.00 น	✓		✓	
	12.00 น	✓		✓	
	16.00 น	✓		✓	

วันที่	เวลา	การแสดงผลข้อมูลหน้าโมบายแอปพลิเคชันจุดรายงานสภาพอากาศ		การแสดงผลรูปภาพที่กล้อง ESP-32 CAM ทำการถ่ายภาพหน้าเว็บแอปพลิเคชันจุดรายงานสภาพอากาศ	
		แสดง	ไม่แสดง	แสดง	ไม่แสดง
4/5/2564	0.00 น	✓		✓	
	4.00 น	✓		✓	
	8.00 น	✓		✓	
	12.00 น	✓		✓	
	16.00 น	✓		✓	
5/5/2564	0.00 น	✓		✓	
	4.00 น	✓		✓	
	8.00 น	✓		✓	
	12.00 น	✓		✓	
	16.00 น	✓		✓	



รูปที่ 4.29 หน้าแสดงผลข้อมูลหน้าโมบายแอปพลิเคชันจุดรายงานสภาพอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.30 หน้าแสดงข้อมูลหน้าโมบายแอปพลิเคชันจุดรายงานสภาพอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### สรุปผลการทดลอง

#### 5.1 สรุปผลการทดลอง

##### 5.1.1 สรุปผลการทดลองส่งค่าของเซ็นเซอร์

จากการทดลองการส่งค่าของเซ็นเซอร์ พบว่าเซ็นเซอร์สามารถส่งค่าได้ถูกต้อง

##### 5.1.2 สรุปผลการทดลองส่งค่าของเซ็นเซอร์บันทึกลงดาต้าเบส

จากการทดลองการส่งค่าของเซ็นเซอร์บันทึกลงดาต้าเบส พบว่าค่าที่ส่งมาจากเซ็นเซอร์สามารถบันทึกลงดาต้าเบสได้ถูกต้อง

##### 5.1.3 สรุปผลการแสดงค่าเซ็นเซอร์ออกหน้าเว็บ

จากการทดลองแสดงค่าเซ็นเซอร์ออกหน้าเว็บ พบว่าสามารถแสดงค่าเซ็นเซอร์บนหน้าเว็บได้ถูกต้อง

##### 5.1.4 สรุปผลการแสดงค่าเซ็นเซอร์ออกโมบายแอปพลิเคชัน

จากการทดลองแสดงค่าเซ็นเซอร์ออกหน้าโมบายแอปพลิเคชัน พบว่าสามารถแสดงค่าเซ็นเซอร์บนหน้าโมบายแอปพลิเคชัน ได้ถูกต้อง

##### 5.1.5 สรุปผลการการถ่ายภาพของกล้องESP32 CAM

จากการทดลองการถ่ายภาพของกล้องESP32-CAMพบว่าสามารถถ่ายภาพได้ตรงตามเวลาที่กำหนดและบันทึกลงดาต้าเบส

#### 5.2 ปัญหาและอุปสรรคในการทำงาน

- 1) โค้ดการทำงานที่เป็นไลบรารีที่ใช้กับเซ็นเซอร์ไม่ตรงตามต้องการต้องทำการแก้ไขให้เข้ากับตัวไมโครคอนโทรลเลอร์ที่ใช้
- 2) ความแรงของสัญญาณอินเตอร์เน็ตไม่เสถียรมีผลกระทบต่อค่าของเซ็นเซอร์ต่างๆ
- 3) เมื่อปล่อยเซ็นเซอร์ทำงานหรือส่งค่าในเวลานานๆเซ็นเซอร์อาจจะมีการส่งค่าที่ผิดปกติ
- 4) เนื่องจากกลุ่มของข้าพเจ้าได้มีการทดสอบในสถานที่จริงจึงมีอุปสรรคในการเลือกสถานที่

#### 5.3 แนวทางแก้ไขปัญหาและอุปสรรค

- 1) เลือกใช้เป็นไลบรารีที่ใช้กับเซ็นเซอร์ต่างๆให้ตรงกับยี่ห้อคอนโทรลเลอร์ที่ใช้
- 2) ทดสอบความแรงของสัญญาณอินเตอร์เน็ตเพื่อหาสัญญาณอินเตอร์เน็ตที่มีความเสถียร
- 3) ค่อยเช็คเซ็นเซอร์ตลอดและมีเซนเซอร์สำรองไว้กรณีเซ็นเซอร์ชำรุด
- 4) เลือกหาสถานที่ที่อำนวยความสะดวกในการใช้ไฟและสัญญาณอินเตอร์เน็ตที่มีความเสถียร

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 5.4 ข้อเสนอแนะแล้วทางการพัฒนา

- 1) หาไลบรารีที่สามารถทำงานได้กับอุปกรณ์
- 2) หาสถานที่ตั้งที่มีสัญญาณอินเทอร์เน็ตที่เสถียร
- 3) หาเซ็นเซอร์ที่มีความคงทน
- 4) จุดรายงานสภาพอากาศนี้สามารถนำมาประยุกต์ใช้กับกังหันต่างๆได้เช่นการเกษตรสามารถใช้ในการวัดค่าลมในการช่วยกสนตัดสินใจในการพินยาให้กับพืชผักต่างๆ



เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม

- [1] [myarduino.net](http://myarduino.net) “NodeMCUESP8266เริ่มต้นติดตั้ง NodeMCUESP8266 ลงบน Arduino IDE”  
<https://www.myarduino.net/สอนใช้งาน-nodemcu-esp8266-เริ่มต้นติดตั้ง>  
เข้าถึงครั้งล่าสุด 25 มิถุนายน 2564
- [2] 99thai.com “การเชื่อมต่อ NodeMCU ESP8266 เข้ากับเครือข่าย WiFi”  
<http://99thai.com> เข้าถึงครั้งล่าสุด 25 มิถุนายน 2564
- [3] อีรพงษ์ ตาทอง “เรียนรู้ไมโครคอนโทรลเลอร์ด้วย Raspberry Pi 3”  
<http://kruteerapong.com/u14Knowledge.php> เข้าถึงครั้งล่าสุด 25 มิถุนายน 2564
- [4] myarduino.net “การเชื่อมต่อ ESP8266 กับเซนเซอร์วัดความเข้มแสง BH1750”  
<https://www.myarduino.net> เข้าถึงครั้งล่าสุด 25 มิถุนายน 2564
- [5] myarduino.net “เซ็นเซอร์วัดความเข้มของแสง UV Sensor Module S12SD”  
[www.myarduino.net/เซ็นเซอร์วัดความเข้มของแสง-uv-sensor-module-s12sd](http://www.myarduino.net/เซ็นเซอร์วัดความเข้มของแสง-uv-sensor-module-s12sd)  
เข้าถึงครั้งล่าสุด 25 มิถุนายน 2564
- [6] esp32learning.com ‘ESP32 and GUVA-S12SD UV Sensor example’  
<http://www.esp32learning.com/esp32-and-guva-s12sd.php> เข้าถึงครั้งล่าสุด 25 มิถุนายน 2564
- [7] myarduino.net “เซ็นเซอร์วัดลม “  
<https://www.myarduino.net/product/wind-speed> เข้าถึงครั้งล่าสุด 25 มิถุนายน 2564
- [8] projects.raspberrypi.org “Buildyourownweatherstation”  
<https://projects.raspberrypi.org/build-your-own-weather-station> เข้าถึงครั้งล่าสุด 25 มิถุนายน 2564
- [9] instructables “ Arduino Rain Gauge Calibration”  
<https://www.instructables.com/Arduino-Rain-Gauge-Calibration/> เข้าถึงครั้งล่าสุด 25 มิถุนายน 2564
- [10] arduitrronics.com “การใช้งาน DHT11 Humidity and Temperature Sensor”  
<https://www.arduitronics.com/articl/การใช้งาน-dht11-sensor> เข้าถึงครั้งล่าสุด 25 มิถุนายน 2564
- [11] myarduino.net “โมดูล ESP32-Cam พร้อมกล้อง OV2640”

- <https://www.myarduino.net/โมดูล-esp32-cam-พร้อมกล้อง-ov2640> เข้าถึงครั้งล่าสุด 25 มิถุนายน 2564
- [12] myarduino.net “สอนใช้งาน Arduino ESP32-CAM ทำกล้องวงจรปิด Wifi”  
<https://www.myarduino.net/esp32-cam-ทำกล้องวงจรปิด-wifi> เข้าถึงครั้งล่าสุด 25 มิถุนายน 2564
- [13] enjoyday.net “สอนทำเว็บตั้งแต่พื้นฐาน สอน HTML”  
[http://www.enjoyday.net/html/html\\_chapter04.html](http://www.enjoyday.net/html/html_chapter04.html) เข้าถึงครั้งล่าสุด 25 มิถุนายน 2564
- {14} bu.ac.th “ภาษาเขียนเว็บ”  
<https://www.bu.ac.th/pdf/aw32.pdf> เข้าถึงครั้งล่าสุด 25 มิถุนายน 2564
- [15] blog.sogoodweb. “CSS คืออะไร มีประโยชน์ อย่างไร”  
<https://blog.sogoodweb.com/Article/Detail/79237/CSS> เข้าถึงครั้งล่าสุด 25 มิถุนายน 2564
- [16] marcuscode.co ”แนะนำภาษา JavaScript”  
<http://marcuscode.com/lang/javascript/introducing-to-javascript>
- [17] webdodee ” PHP คืออะไร สามารถทำอะไรได้บ้าง”  
<https://www.webdodee.com/what-is-php/> เข้าถึงครั้งล่าสุด 25 มิถุนายน 2564

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

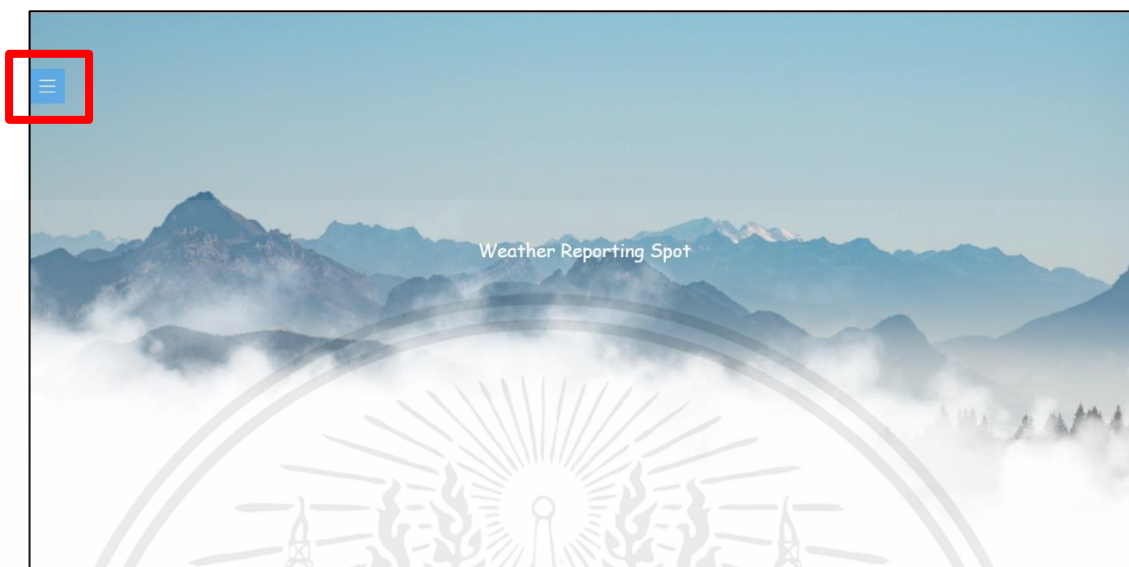


ภาคผนวก ก


คู่มือการใช้งานเว็บไซต์แอปพลิเคชันจตุรกายงานสภาพอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## คู่มือการใช้งานเว็บแอปพลิเคชันจตุรรายงานสภาพอากาศ



รูปที่ ก.1 หน้าเว็บแอปพลิเคชันจตุรรายงานสภาพอากาศ

จากรูปที่ ก 1 เมื่อผู้ใช้งานเข้ามาใช้งานเว็บแอปพลิเคชันจตุรรายงานสภาพอากาศจะพบกับหน้าแรกของเว็บแอปพลิเคชันจตุรรายงานสภาพอากาศเมื่อคลิกปุ่ม  จะสามารถเลือกสถานที่ที่ผู้ใช้ต้องการจะทำการเข้าไปดูการรายงานสภาพอากาศและรูปถ่ายของแต่ละสถานที่ได้



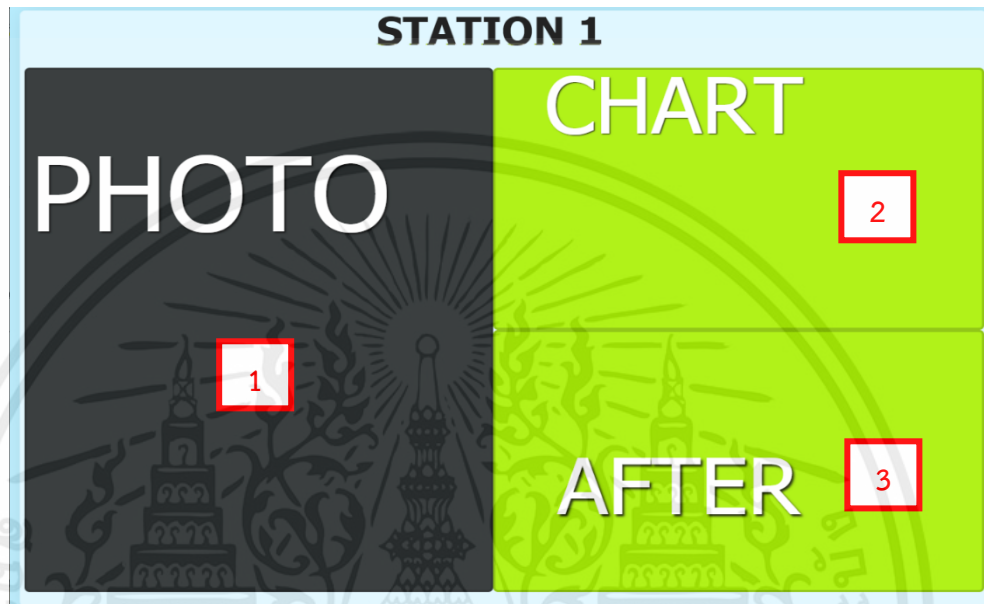
รูปที่ ก.2 หน้าเลือกสถานที่ของเว็บแอปพลิเคชันจตุรรายงานสภาพอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

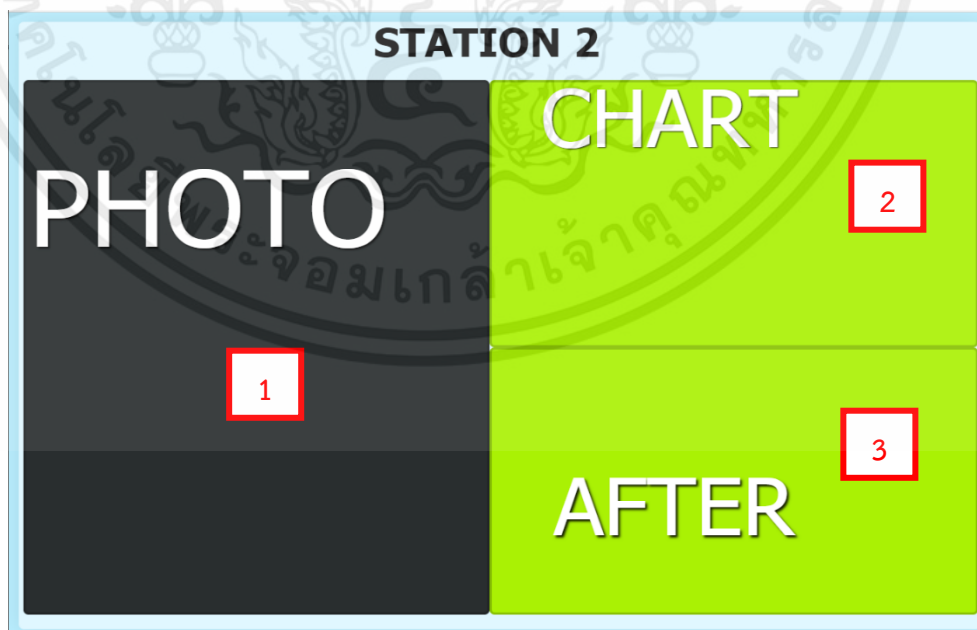
จากรูปที่ ก 2 เป็นหน้าเว็บแอปพลิเคชันจุดรายงานสภาพอากาศที่ให้ผู้ใช้งานได้เลือกสถานที่  
ได้ซึ่งมี 2 สถานที่ที่ผู้ใช้งานสามารถทำการคลิกที่คำว่า Station1 หรือ Station2 ได้

หมายเลข 1 สถานที่ 1 ทดลองตั้งที่หน้าหอพักนักศึกษาของผู้จัดทำ

หมายเลข 2 สถานที่ 2 ทดลองตั้งที่หอพักอาจารย์



รูปที่ ก.3 หน้าแรกของสถานี 1



รูปที่ ก.4 หน้าแรกของสถานี 2

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ ก 3 เป็นหน้าเว็บแอปพลิเคชันจตุรายนสภาพอากาศที่ใช้คลิกที่คำว่า Station1 ตอนที่ให้ผู้ใช้เลือกดูได้ว่าจะดูที่คำว่า Station1 หรือ Station2

หมายเลข 1 ฟังก์ชันเข้ามาดูภาพถ่ายของสถานที่ 1

หมายเลข 2 ฟังก์ชันเข้ามาดูรายงานสภาพอากาศ

หมายเลข 3 ฟังก์ชันดูรายงานสภาพอากาศย้อนหลัง 7 วัน

จากรูปที่ ก 4 เป็นหน้าเว็บแอปพลิเคชันจตุรายนสภาพอากาศที่ใช้คลิกที่คำว่า Station2 ตอนที่ให้ผู้ใช้เลือกดูได้ว่าจะดูที่คำว่า Station1 หรือ Station2

หมายเลข 1 ฟังก์ชันเข้ามาดูภาพถ่ายของสถานที่ 2

หมายเลข 2 ฟังก์ชันเข้ามาดูรายงานสภาพอากาศ

หมายเลข 3 ฟังก์ชันดูรายงานสภาพอากาศย้อนหลัง 7 วัน

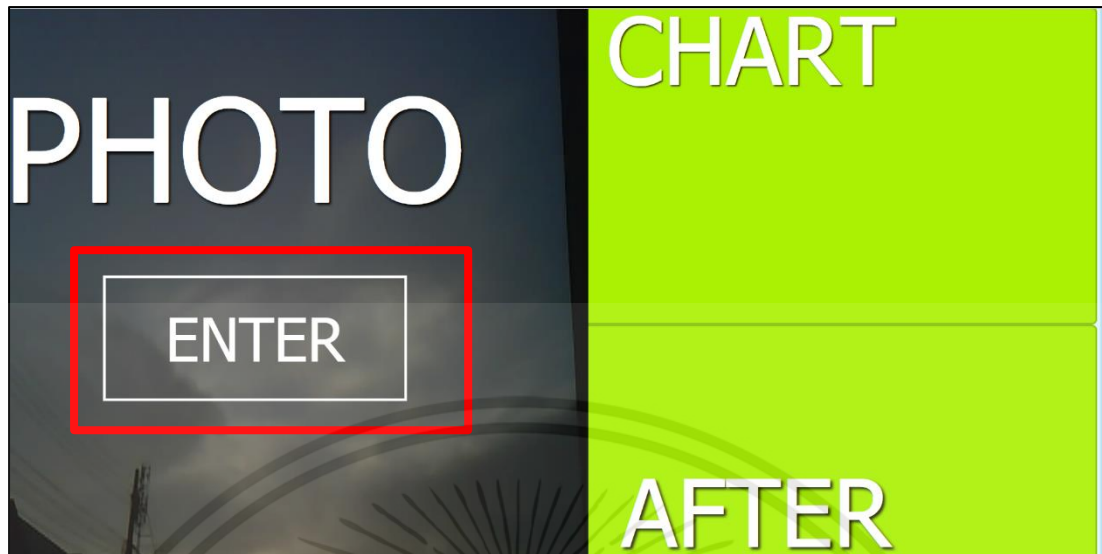
#### ฟังก์ชันเข้ามาดูภาพถ่าย

หากเอาเลื่อนลูกศรมาบริเวณหมายเลข 1 ตามรูปที่ ก 3 กับ รูปที่ ก 4 จะขึ้นคำว่า ENTER ตามรูปที่ ก 5 จะเป็นของสถานที่1 และรูปที่ ก 6 เป็นของสถานที่2 และคลิกที่คำว่า ENTER เพื่อเข้ามาดูภาพถ่าย



รูปที่ ก.5 ฟังก์ชันดูภาพถ่ายล่าสุดของสถานที่1

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.6 ฟังก์ชันดูภาพถ่ายล่าสุดของสถานที่2



รูปที่ ก.7 ดูภาพถ่ายล่าสุดของสถานที่1

จากรูปที่ ก 7 เป็นหน้าเว็บแอปพลิเคชันจตุรรายงานสภาพอากาศหลังจากที่คลิกคำว่า ENTER  
รูปที่ ก 5 แสดงภาพถ่ายล่าสุดที่สถานี1ทำการถ่ายภาพโดยจะถ่ายทุก1ชั่วโมง

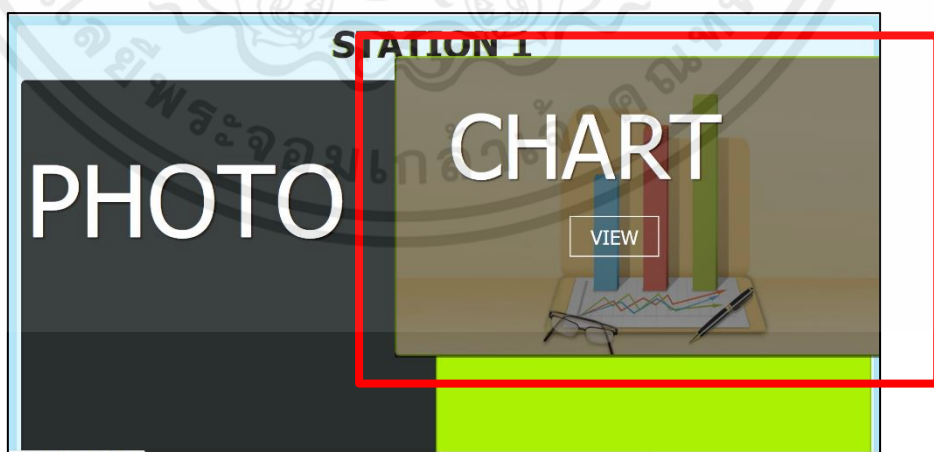
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.8 รูปภาพถ่ายล่าสุดของสถานีที่2

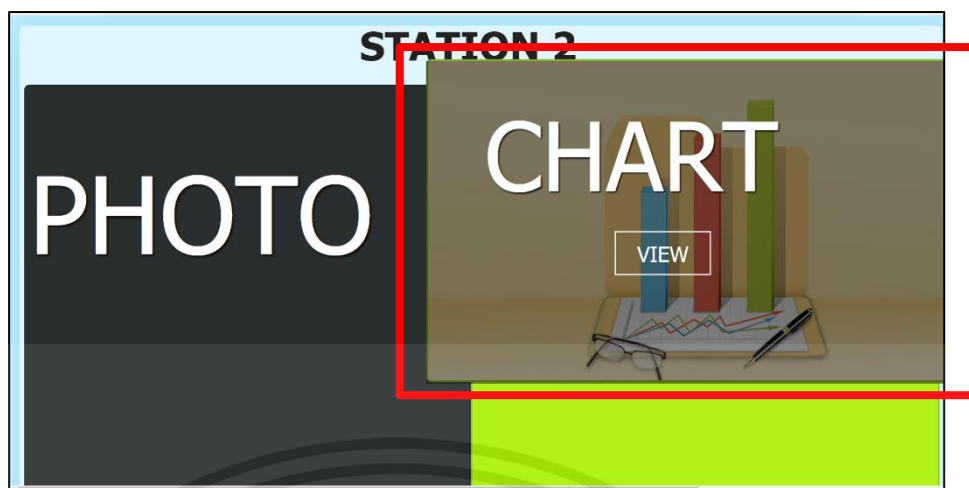
จากรูปที่ ก 8 เป็นหน้าเว็บแอปพลิเคชันจตุรายนสภาพอากาศหลังจากที่คลิกคำว่า ENTER รูปที่ ก 6 แสดงภาพถ่ายล่าสุดที่สถานีที่2ทำการถ่ายภาพโดยจะถ่ายทุก1ชั่วโมง ฟังก์ชันเข้ามาดูรายงานสภาพอากาศ

หากเอาเลื่อนลูกศรมาบริเวณหมายเลข 2 ตามรูปที่ ก 3 กับ รูปที่ ก 4 จะขึ้นคำว่า ENTER ตามรูปที่ ก 9 จะเป็นของสถานีที่1 และรูปที่ ก 10 เป็นของสถานีที่2 และคลิกที่คำว่า ENTER เพื่อดูรายงานสภาพอากาศ



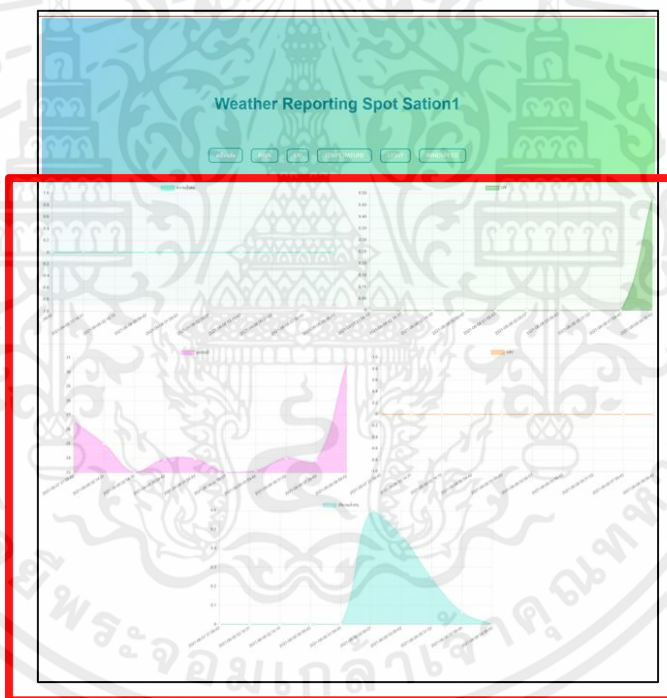
รูปที่ ก.9 ฟังก์ชันดูรายงานสภาพอากาศของสถานีที่1

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ส

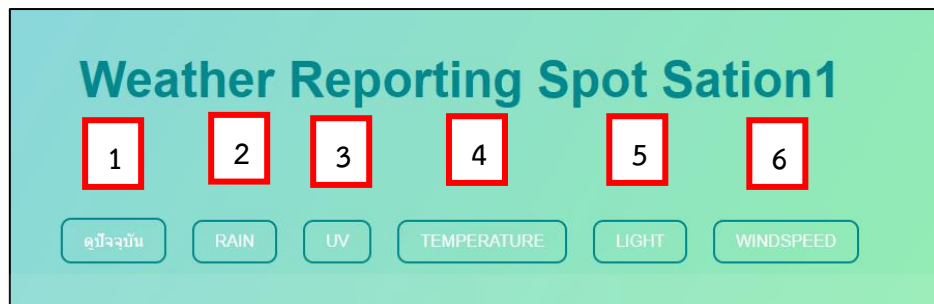
รูปที่ ก.10 ฟังก์ชันดูรายงานสภาพอากาศของสถานีที่2



รูปที่ ก.11 หน้าแรกเว็บดูรายงานสภาพอากาศสถานีที่1

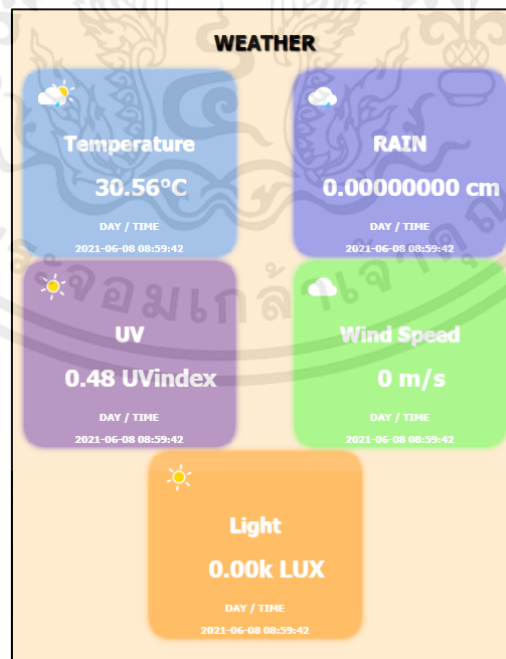
จากรูปที่ ก 9 แสดงรายสภาพอากาศ5ประเภทที่สถานีที่1ทำการเก็บค่าของ ความเร็วลม ความเข้มยูวี อุณหภูมิ ความเข้มแสง ปริมาณน้ำฝน ทุก1ชั่วโมงเป็นแบบกราฟสามารถดูย้อนหลังได้ 10ชั่วโมงและทางด้านขวามือจะเป็นค่าล่าสุดที่สถานีที่1ส่งมา

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



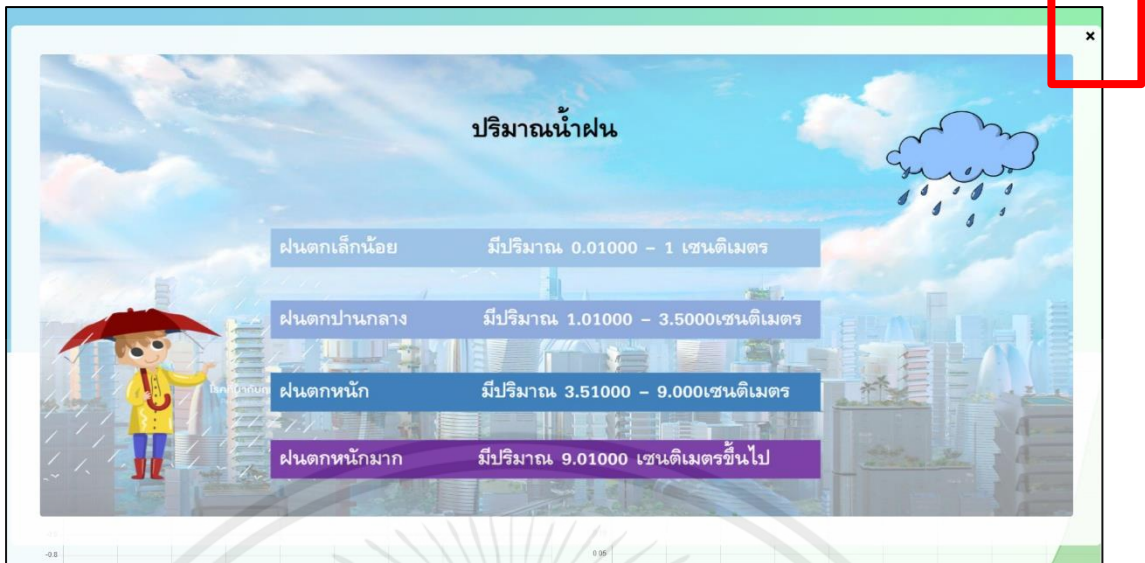
รูปที่ ก.12 ฟังก์ชันของหน้ารายงานสภาพอากาศ

จากรูปที่ ก 10 มีทั้งหมด6ฟังก์ชันของหน้ารายงานสภาพอากาศสามารถกดปุ่มเพื่อดูได้  
 หมายเลข 1 แสดงค่าปัจจุบันตามรูปที่ ก 13  
 หมายเลข 2 แสดงป๊อปอัพค่าการวัดมาตรฐานของปริมาณน้ำฝนตามรูปที่ ก 14  
 หมายเลข 3 แสดงป๊อปอัพค่าการวัดมาตรฐานของความเข้มยูวีตามรูปที่ ก 15  
 หมายเลข 4 แสดงป๊อปอัพค่าการวัดมาตรฐานของอุณหภูมิตามรูปที่ ก 16  
 หมายเลข 5 แสดงป๊อปอัพค่าการวัดมาตรฐานของความเข้มแสงตามรูปที่ ก 17  
 หมายเลข 6 แสดงป๊อปอัพค่าการวัดมาตรฐานของความเร็วลมตามรูปที่ ก 18



รูปที่ ก.13 หน้าเว็บค่าล่าสุดที่วัดได้สถานีที่1

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.14 ป็ออปัค่าการวัดมาตรฐานของปริมาณน้ำฝน



รูปที่ ก.15 ป็ออปัค่าการวัดมาตรฐานของความเข้มยูวี

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.16 ป้อนอัปเดตค่าการวัดมาตรฐานของอุณหภูมิ

เงื่อนไข	ไฟส่องสว่าง (lux)
แสงแดด	107,527
กลางวันเต็มที่	10,752
วันที่ฟ้าครึ้ม	1,075
วันที่มืดมาก	107
สนธยา (Twilight) (ช่วงเวลาก่อนดวงอาทิตย์ขึ้นหรือหลังจกดวงอาทิตย์ตก)	10.8
พระจันทร์เต็มดวง	0.108
คืนที่มืดครึ้ม	0.0001

รูปที่ ก.17 ป้อนอัปเดตค่าการวัดมาตรฐานของความเข้มแสง

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



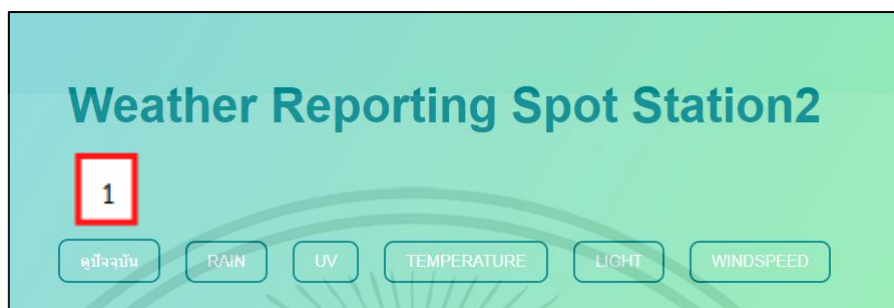
รูปที่ ก.18 ป๊อปอัพค่าการวัดมาตรฐานของความเร็วลม



รูปที่ ก.19 หน้าแรกเว็บดูรายงานสภาพอากาศสถานีที่2

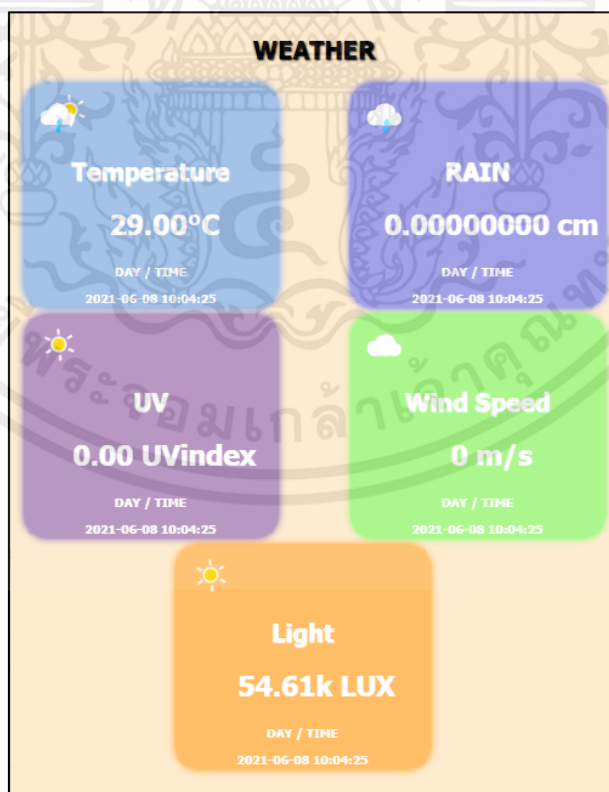
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ ก 19 แสดงรายสภาพอากาศ5ประเภทที่สถานีที่1ทำการเก็บค่าของ ความเร็วลม ความเข้มยูวี อุณหภูมิ ความเข้มแสง ประมาณน้ำฝน ทุก1ชั่วโมงเป็นแบบกราฟสามารถดูย้อนหลังได้ 10ชั่วโมงและทางด้านขวามือจะเป็นค่าล่าสุดที่สถานีที่2ส่งมา



รูปที่ ก.20 ฟังก์ชันของหน้ารายงานสภาพอากาศ

จากรูปที่ ก 20 มีทั้งหมด6ฟังก์ชันของหน้ารายงานสภาพอากาศสามารถกดปุ่มเพื่อดูได้ หมายเลข 1 แสดงค่าปัจจุบันตามรูปที่ ก 21

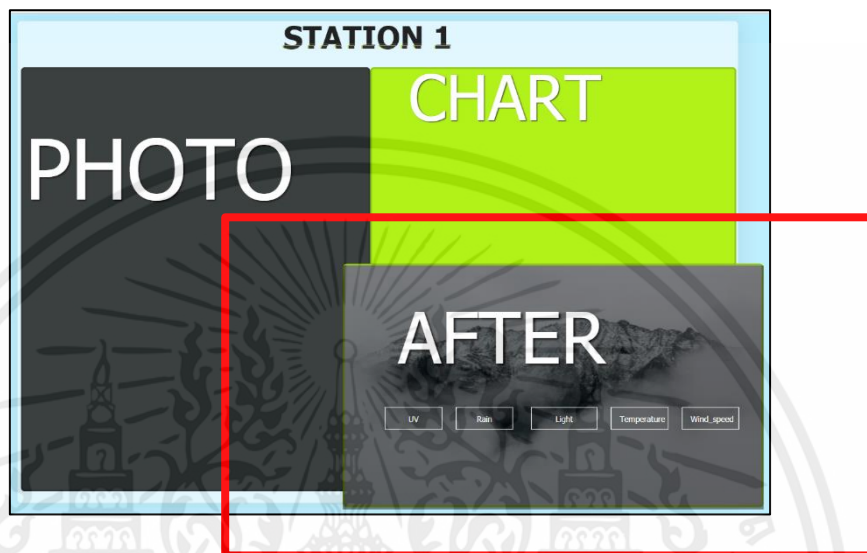


รูปที่ ก.21 หน้าเว็บค่าล่าสุดที่วัดได้สถานีที่2

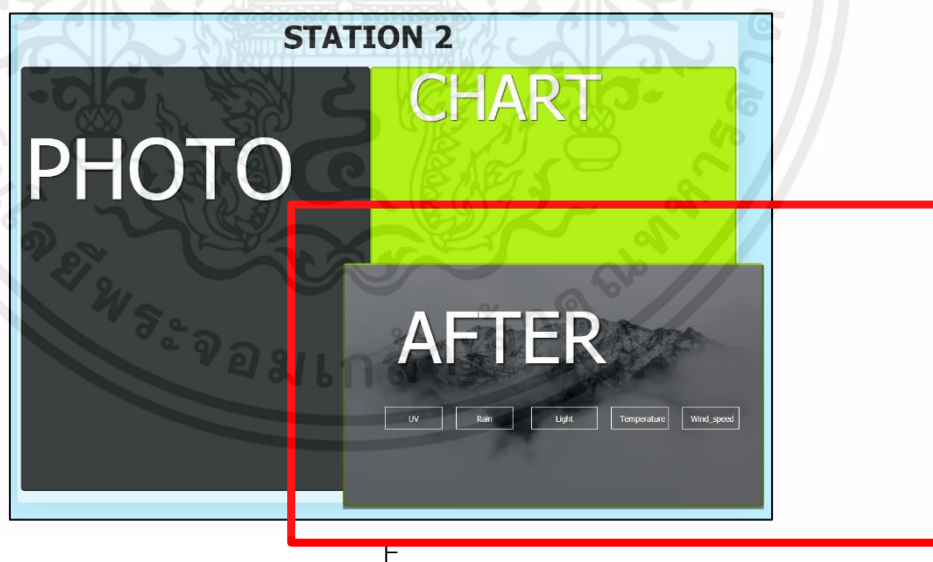
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ฟังก์ชันดูรายงานสภาพอากาศย้อนหลัง 7 วัน

หากเอาเลื่อนลูกศรมาบริเวณหมายเลข 3 ตามรูปที่ ก 3 กับ รูปที่ ก 4 จะขึ้นคำว่า UV ,Rain Light ,Temperature ,Wind\_speed ตามรูปที่ ก 22 จะเป็นของสถานีที่1 และรูปที่ ก 23 เป็นของสถานีที่2 เป็นการรายงานสภาพอากาศย้อนหลัง7วันเป็นกราฟ

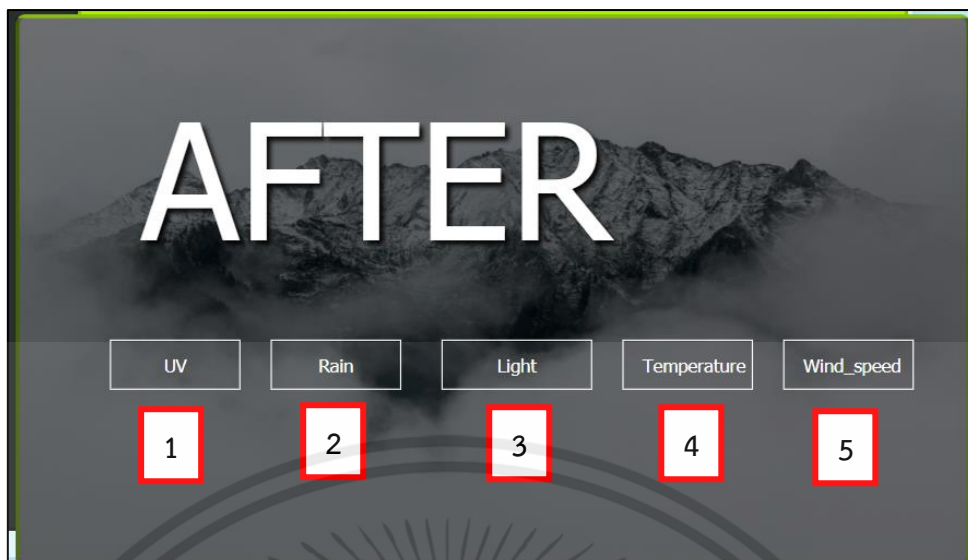


รูปที่ ก.22 ฟังก์ชันดูรายงานสภาพอากาศย้อนหลัง 7 วันของสถานีที่1



รูปที่ ก.23 ฟังก์ชันดูรายงานสภาพอากาศย้อนหลัง 7 วันของสถานีที่2

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.23 ฟังชั่นของรายงานสภาพอากาศย้อนหลัง 7 วัน

จากรูปที่ ก 23 เป็นการดูรายงานสภาพอากาศ7วันเป็นกราฟดูรายงานสภาพอากาศของความเข้มข้นวี ปริมาณน้ำฝน ความเข้มแสง อุณหภูมิ ความเร็วลม 5ประเภท

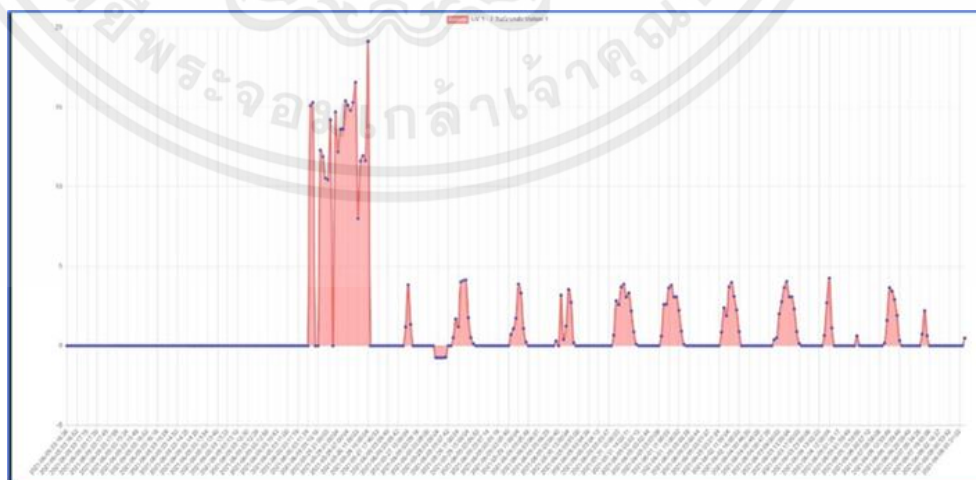
หมายเลข 1 ดูรายงานย้อนหลังของความเข้มข้นวีตามรูปที่ ก 24

หมายเลข 2 ดูรายงานย้อนหลังของปริมาณน้ำฝนตามรูปที่ ก 25

หมายเลข 3 ดูรายงานย้อนหลังของความเข้มแสงตามรูปที่ ก 26

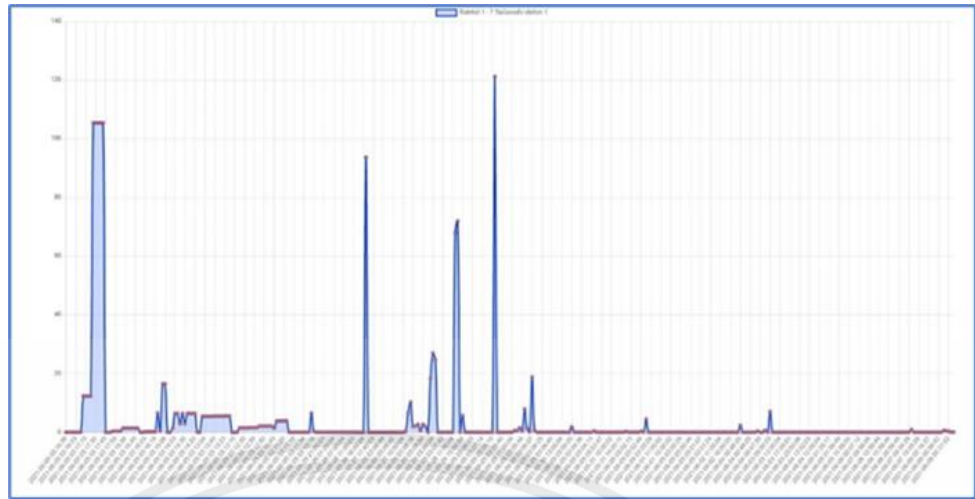
หมายเลข 4 ดูรายงานย้อนหลังของอุณหภูมิตามรูปที่ ก 27

หมายเลข 5 ดูรายงานย้อนหลังของความเร็วลมตามรูปที่ ก 28

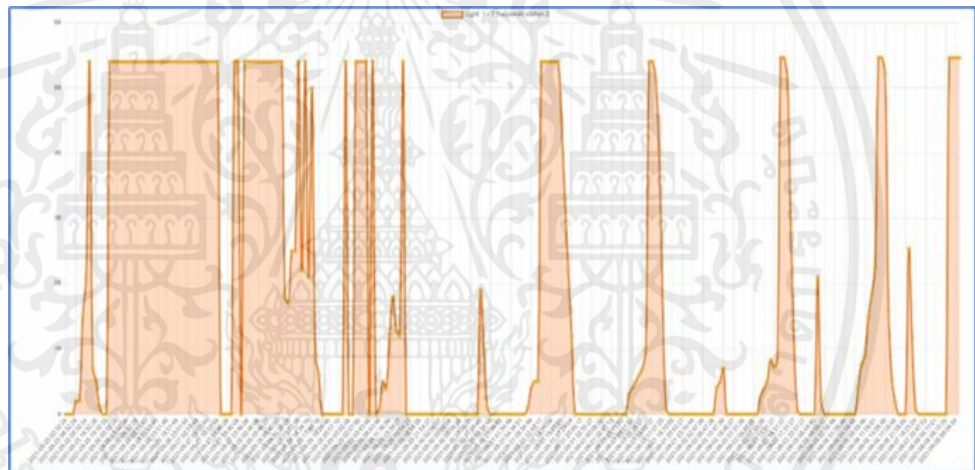


รูปที่ ก.24 รายงานสภาพอากาศย้อนหลัง7วันของความเข้มข้นวี

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

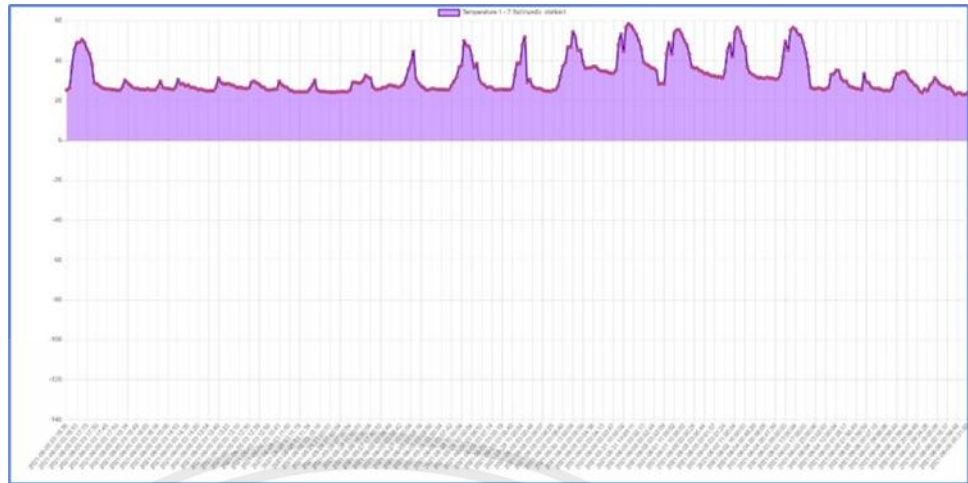


รูปที่ ก.25 รายงานสภาพอากาศย้อนหลัง7วันของปริมาณน้ำฝน

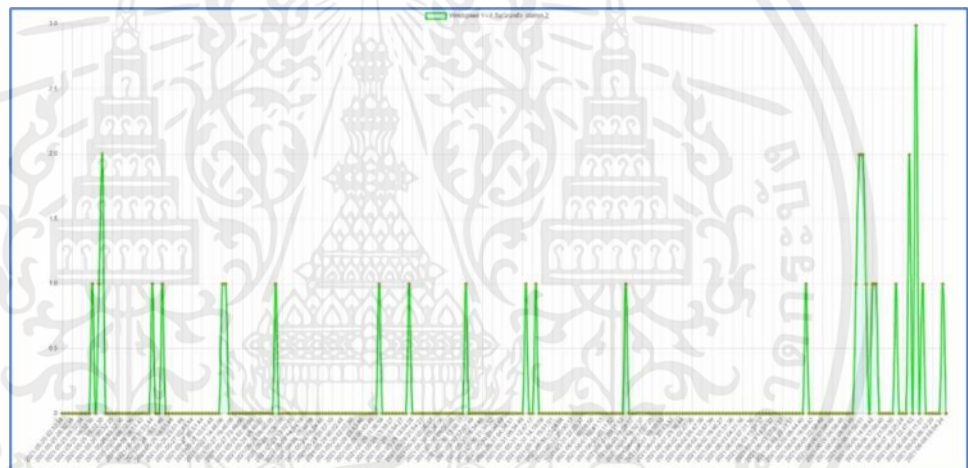


รูปที่ ก.26 รายงานสภาพอากาศย้อนหลัง7วันของความเข้มแสง

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.27 รายงานสภาพอากาศย้อนหลัง 7 วันของอุณหภูมิ



รูปที่ ก.28 รายงานสภาพอากาศย้อนหลัง 7 วันของความเร็วลม

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ประวัติผู้เขียน



ชื่อ-นามสกุล นายคณิศร์ ทองก้านบัว  
 วัน เดือน ปีเกิด วันที่ 25 กันยายน 2541  
 ภูมิลำเนา จังหวัดเพชรบุรี  
 ที่อยู่ 284/23 ต.ชะอำ อ.ชะอำ  
 จ.เพชรบุรี 76120

E-mail [kanis\\_pia@hotmail.co.th](mailto:kanis_pia@hotmail.co.th)

### ประวัติการศึกษา

- สำเร็จการศึกษาระดับประกาศนียบัตรวิชาชีพ (อิลีททรอนิกส์) วิทยาลัยการอาชีพวังไกลกังวล
- สำเร็จการศึกษาระดับปริญญาตรีวิศวกรรมศาสตรบัณฑิต วศ.บ. วิศวกรรมสารสนเทศ ปีการศึกษา 2563 จากสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง วิทยาเขตชุมพรเขตอุดมศักดิ์ จังหวัดชุมพร

### ผลงานกิจกรรม

- ประธานชมรมหุ่นยนต์ชมรมหุ่นยนต์ลาดกระบังชุมพร
- แข่งขันรายการ ABURobocon2017, ABURobocon2018
- WAR OF STEEL 3 (Battle Robot 2018)
- WAR OF STEEL 4 (Battle Robot 2019)
- Thailand Innovation 4.0 เพื่อชุมพร เครื่องส์ทุเรียน สำหรับพลาสติกชีวภาพ รางวัลชมเชย
- CPF Hackathon 2020 on July 7-23, 2020
- LINE CHATBOT
- วิทยากรของค่าย PEPC สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง วิทยาเขตชุมพร (ข้าพเจ้าเป็นผู้บรรยายให้ให้ความรู้เด็กเกี่ยวกับการสร้างหุ่นยนต์ติดตามสาย)
- สมาชิกชมรมอาสาสมัครค่าย สโมสรค่ายลาดกระบังชุมพร (สร้างสนามเด็กเล่นให้โรงเรียนห่างไกลความเจริญ)

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ประวัติผู้เขียน



ชื่อ-นามสกุล นายคงศักดิ์ ทองก้านบัว  
 วัน เดือน ปีเกิด วันที่ 25 กันยายน 2541  
 ภูมิลำเนา จังหวัดเพชรบุรี  
 ที่อยู่ 284/23 ต.ชะอำ อ.ชะอำ  
 จ.เพชรบุรี 76120

E-mail [death01\\_por@hotmail.com](mailto:death01_por@hotmail.com)

### ประวัติการศึกษา

- สำเร็จการศึกษาระดับประกาศนียบัตรวิชาชีพ (อิลีททรอนิกส์) วิทยาลัยการอาชีพวังไกลกังวล
- สำเร็จการศึกษาระดับปริญญาตรีวิศวกรรมศาสตรบัณฑิต วศ.บ. วิศวกรรมสารสนเทศ ปีการศึกษา 2563 จากสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง วิทยาเขตชุมพรเขตอุดมศักดิ์ จังหวัดชุมพร

### ผลงานกิจกรรม

- สมาชิกชมรมหุ่นยนต์ชมรมหุ่นยนต์ลาดกระบังชุมพร
- แข่งขันรายการ ABURobocon2017, ABURobocon2018
- WAR OF STEEL 3 (Battle Robot 2018)
- WAR OF STEEL 4 (Battle Robot 2019)
- Thailand Innovation 4.0 เพื่อชุมพร เครื่องส์ทุเรียน สำหรับพลาสติกชีวภาพ รางวัลชมเชย
- CPF Hackathon 2020 on July 7-23, 2020
- LINE CHATBOT
- วิทยากรของค่าย PEPC สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง วิทยาเขตชุมพร (ข้าพเจ้าเป็นผู้บรรยายให้ให้ความรู้เด็กเกี่ยวกับการสร้างหุ่นยนต์ติดตามสาย)
- สมาชิกชมรมอาสาสมัครค่าย สโมสรค่ายลาดกระบังชุมพร (สร้างสนามเด็กเล่นให้โรงเรียนห่างไกลความเจริญ)

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้